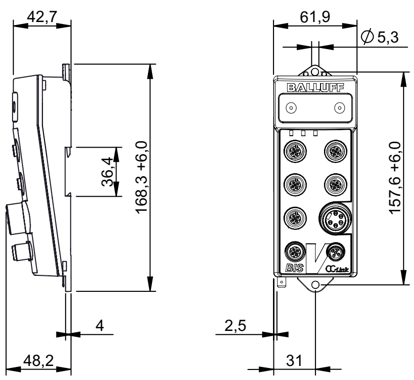
**BALLUFF**

# BIS V-6111 CC-Link

## Technical Description, User's Guide





**english**

www.balluff.com

# 1 User Instructions

## 1.1 About this Manual

This manual describes the processor unit for BIS V-6111 Identification Systems and startup instructions for immediate operation.

## 1.2 Typographical Conventions

The following conventions are used in this manual:

**Enumerations**

Enumerations are shown as a list with an en-dash.
– Entry 1.
– Entry 2.

**Actions**

Action instructions are indicated by a preceding triangle. The result of an action is indicated by an arrow.
► Action instruction 1.
⇒ Action result.
► Action instruction 2.

**Syntax**

**Numbers:**
– Decimal numbers are shown without additional indicators (e.g. 123),
– Hexadecimal numbers are shown with the additional indicator $_{hex}$ (e.g. $00_{hex}$).

**Parameters:**
Parameters are shown in italics (e.g. *CRC_16*).

**Directory paths:**
References to paths where data is stored or to be saved are shown in small caps (e.g. PROJECT:\DATA TYPES\USER DEFINED).

**Control characters:**
Control characters for sending are set in angle brackets (e.g. <ACK>).

**ASCII code:**
Characters transmitted in ASCII code are set in apostrophes (e.g. 'L').

## 1.3 Symbols

⚠ **Caution!**
This symbol indicates a safety instruction that must be followed without exception.

ℹ **Note, tip**
This symbol indicates general notes.

## 1.4 Abbreviations

| | |
|---|---|
| BIS | Balluff Identification System |
| CC-Link | Control and Communication Link |
| CLPA | CC-Link Partner Association |
| CP | Code Present |
| CRC | Cyclic Redundancy Check |
| CSP | CC-Link System Profile |
| DID | Device ID |
| EEPROM | Electrical Erasable and Programmable ROM |
| EMC | Electromagnetic compatibility |
| FCC | Federal Communications Commission |
| FE | Function ground |
| LF CR | Line Feed with Carriage Return |
| MAC | Media Access Control |
| n.c. | not connected |
| PC | Personal Computer |
| PLC | Programmable Logic Controller |
| Tag | Data carriers |
| UID | Unique Identifier |
| VID | Vendor ID |

# 2 Safety

## 2.1 Intended Use

The BIS V-6111 processor unit is a component of the BIS V Identification System. Within the identification system it is used for linking to a host computer (PLC, PC). It may be used only for this purpose in an industrial environment corresponding to Class A of the EMC Law.
This description applies to processor units of the following series:
– BIS V-6111-073-C003
– BIS V-6111-073-C103

## 2.2 General Safety Notes

**Installation and startup**
Installation and startup are to be performed by trained technical personnel only. Any damage resulting from unauthorized manipulation or improper use voids warranty and liability claims against the manufacturer.
When connecting the processor unit to an external controller, observe proper selection and polarity of the connection as well as the power supply (see "Processor Unit Installation" on page 9).
The processor unit may only be used with an approved power supply (see "Technical Data" on page 12).

**Conformity**

| | |
|---|---|
| CE | This product was developed and manufactured in accordance with the applicable European directives.<br>CE conformity has been verified. |

All approvals and certifications are no longer valid if:
– Components are used that are not part of the BIS V Identification System,
– Components are used that have not been explicitly approved by Balluff.

**Operation and testing**
The operator is responsible for ensuring that local safety regulations are observed.
In the event of defects and non-correctable faults in the identification system, take the system out of service and secure it from unauthorized use.

## 2.3 Meaning of Warning Notes

⚠ **Caution!**
The pictogram used with the word "Caution" warns of a situation that could harm someone's health or damage equipment. Failure to observe these warning notes may result in injury or damage to equipment.
► Always observe the described measures for preventing this danger.

# 3 Basic Knowledge

## 3.1 Function Principle of Identification Systems

The BIS V Identification System is classified as a non-contacting system with read and write function, which not only allows it to convey information programmed permanently in the data carrier, but also to collect and pass on current information.

The main components of the BIS V Identification System are:
- Processor unit
- Read/write heads,
- Data carriers.

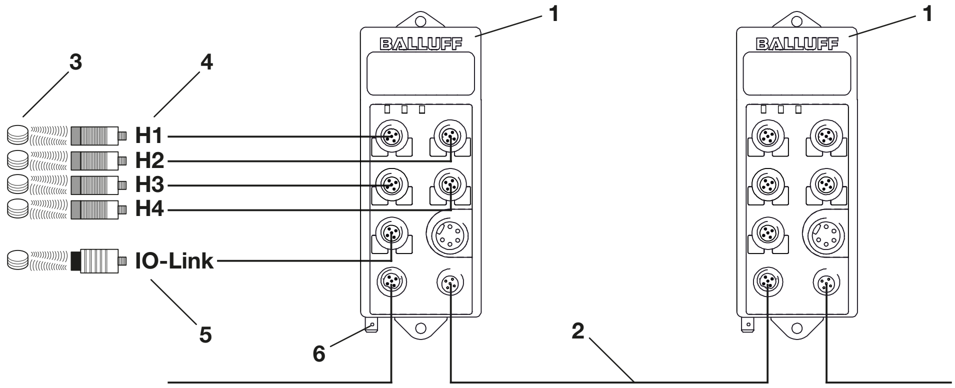



Figure 1: System overview

| | | | |
|---|---|---|---|
| **1** | BIS V | **4** | Read/write heads H1…H4 |
| **2** | CC-Link interface | **5** | Service/IO-Link |
| **3** | Data carriers (max. 1 per R/W head) | **6** | Function ground |

The main areas of application are:
- In production for controlling material flow (e.g. for model-specific processes, conveying systems that transport workpieces, acquisition of safety-relevant data)
- In warehousing for monitoring material movement
- Transporting and conveying

## 3.2 Product Description

Processor Unit BIS V-6111:
- Metal housing
- Round connector terminations
- Four read/write heads can be connected
- 1 × IO-Link module or standard input port can be connected
- Power for the system components provided by the processor unit
- Power for the data carrier provided by the read/write heads via carrier signal
- USB port
- CC-Link In and CC-Link Out
- Display with keys for startup and configuration
- Control displays

# 3 Basic Knowledge

## 3.3 Control Function

The processor unit is the link between data carrier and controlling system. It manages two-way data transfer between data carrier and R/W head and provides buffer storage.
The processor unit uses the read/write head to write data from the controlling system to the data carrier or reads the data from the carrier and makes it available to the controlling system.
Controlling systems may be the following:
- A control computer (e.g. industrial PC).
- A PLC.

## 3.4 Data Integrity

In order to increase data integrity, data transfer between the data carrier and processor unit and the storage device must be monitored using a check procedure.
A CRC_16 data check can be enabled for this via parameter configuration.
With the CRC_16 data check, a check code that allows the validity to be checked at any time is written to the data carrier.

**A CRC_16 data check provides the following advantages:**
- Data integrity even during the non-active phase (data carrier outside the R/W head).
- Shorter read time – page is read once.

## 3.5 Read/write Heads H1…H4

For BIS V-6111-073-C003, read/write heads of the BIS VM-3… and BIS VL-3… series can be connected to terminals H1…H4.
BIS V-6111-073-C103 also supports read/write heads of the BIS C-3… series (adapter required).
Connecting the read/write heads only using shielded cables is recommended for compliance with the EMC Directives.

The maximum cable length for read/write heads of the BIS VM-3… and BIS VL-3… series is 50 m. Depending on the system design, the cable length for the BIS C-3… series is set to 1 m, 5 m or 10 m plus adapter.

## 3.6 CC-Link

Open bus system for process and field communication. Automation devices, such as PLCs, PCs, operating and observation devices, sensors or actuators, can communicate using this bus system.

# 3 Basic Knowledge

## 3.7 IO-Link

IO-Link is defined as a standardized point-to-point connection between sensors/actuators and the I/O module. An IO-Link sensor/actuator can send additional communication data (e.g. diagnostics signals) in addition to the binary process signals over the IO-Link interface.

Compatibility with standard I/O:
- IO-Link sensors/actuators can be connected to existing I/O modules.
- Sensors/actuators which are not IO-Link-capable can be connected to an IO-Link module.
- Standard sensor/actuator cables can be used.

Key technical data:
- Serial point-to-point connection.
- Communication as add-on to standard I/O.
- Standard I/O connection technology, unshielded, 20 m cable length.
- Communication using 24 V pulse modulation, standard UART protocol.

## 3.8 IO-Link Communication Mode

Process data (cyclical):
The size of the IO-Link input and output buffer is always the same and can be adjusted from 2 bytes to 32 bytes in 2-byte increments. If 0 bytes is selected as the size, the IO-Link port is disabled.

Deterministic time behavior:
- Typically 2 ms cycle time for 16 bits of process data and 38.4 kbaud transmission rate.

Service data (diagnostics, parameters):
- Parallel and non-reactive to process data

**Standard I/O mode**

- Startup parameters can be configured using communication, then
- Binary switching signal

## 3.9 USB Port

The device can be connected to a PC's USB interface using the "Service/IO-Link" jack and then behaves like a USB stick (see 4.3). This allows access to the internal memory, where the manual, the CSP+ file and a communications driver for service functions are saved, as well as a program for project configuration. In addition, the BIS V has to be connected to an external voltage source. The communication driver can be installed as needed, but is not required for the USB interface and BIS V to function.

**Note**
Visit www.balluff.com for more information on available software and accessories.

# 4 Installation

## 4.1 Processor Unit Scope of Delivery

Included in the scope of delivery:
- BIS V-6111
- 5 × closure cap
- Safety Notes

**Note**
Visit www.balluff.com for more information on available software and accessories.

## 4.2 Processor Unit Installation

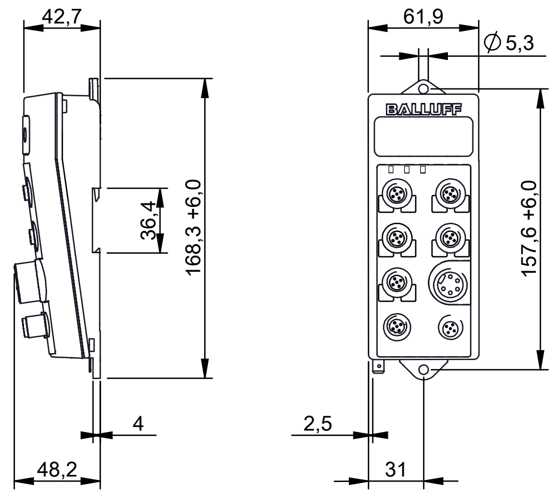


Figure 2: Mechanical connection (dimensions in mm)

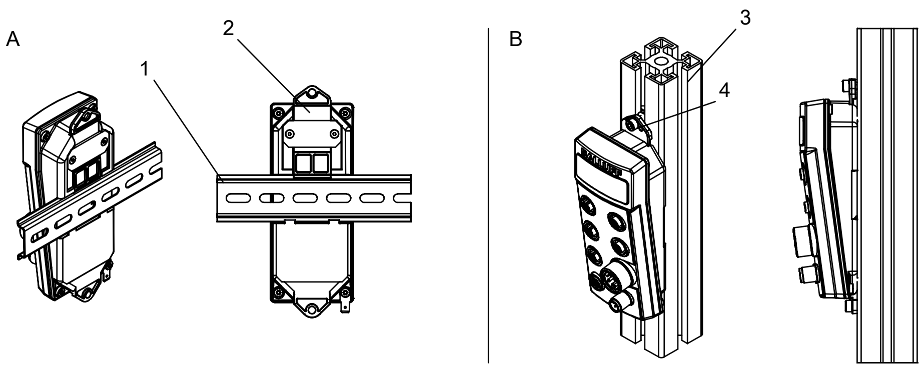


Figure 3: Installation examples (A: attachment to DIN rails, B: attachment to T-slotted framing)

**1** DIN rail
**2** Fastening
**3** T-slotted framing
**4** Holder for screw mounting

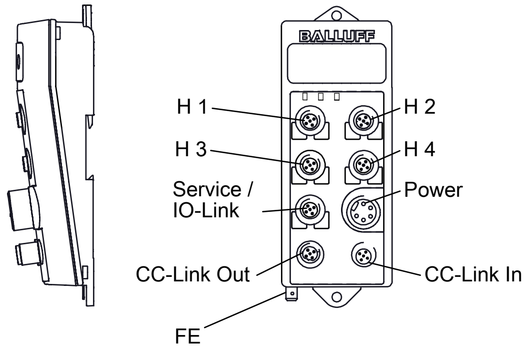
- ► Select a suitable installation position.
- ► Secure the processor unit using two M5 screws (strength category 8.8, lightly oiled, tightening torque M = 5.5 Nm).

# 4 Installation

## 4.3 Electrical Connection

> **Note**
> Make the ground connection either directly or using an RC combination to ground.

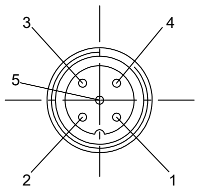
**Connections**




| | |
|---|---|
| H1…H4 | Read/Write Heads |
| Service/IO-Link | USB function Service/IO-Link (master function) |
| CC-Link In | CC-Link input |
| CC-Link Out | CC-Link output |
| FE | Function ground |

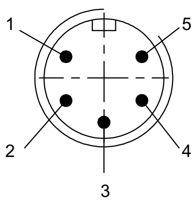
Figure 4: Electrical connection

**H1…H4**
5-pin M12 socket, A-coded



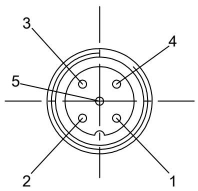

| PIN | Function |
|---|---|
| 1 | +24 V DC |
| 2 | A |
| 3 | 0 V |
| 4 | B |
| 5 | n.c. |

**Power**
5-pin 7/8" plug




| PIN | Function |
|---|---|
| 1 | 0 V |
| 2 | 0 V |
| 3 | FE |
| 4 | +24 V DC |
| 5 | n.c. |

**IO-Link / Service**
5-pin M12 socket, A-coded




IO-Link

| PIN | Function |
|---|---|
| 1 | VP (+24 V DC) |
| 2 | n.c. |
| 3 | 0 V |
| 4 | Q/C (IO-Link) or digital input |
| 5 | n.c. |

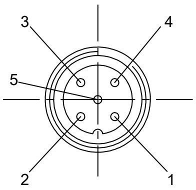
Service/USB

| PIN | Function |
|---|---|
| 1 | n.c. |
| 2 | USB– |
| 3 | 0 V |
| 4 | n.c. |
| 5 | USB+ |

> **NOTICE**
> **Damage to the USB interface**
> **Standard USB cables can damage the USB interface on the PC.**
> ► Use the cable BCC0CR2 for the use of the Balluff Software BIS Cockpit or UHF Manager on the Port IO-Link / Service.

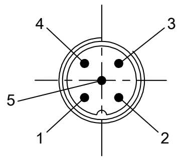
# 4 Installation

**CC-Link Out**
5-pin M12 socket, A-coded




| PIN | Function |
|---|---|
| 1 | SLD |
| 2 | DB |
| 3 | DG |
| 4 | DA |
| 5 | n.c. |

**CC-Link In**
5-pin M12 plug, A-coded




| PIN | Function |
|---|---|
| 1 | SLD |
| 2 | DB |
| 3 | DG |
| 4 | DA |
| 5 | n.c. |

## 5 Technical Data

**Dimensions**

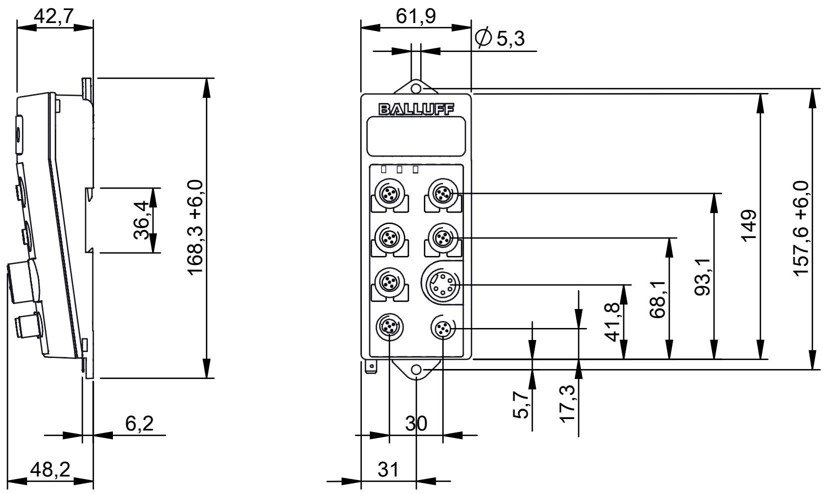



Figure 5: Dimensions in mm

**Mechanical data**

| | |
|---|---|
| Housing material | Zinc die-cast housing |
| H1…H4 | $V_S$ 24 V DC - 5-pin M12 female, A-coded |
| Service/IO-Link (master function) | 5-pin M12 socket, A-coded |
| Power | 5-pin 7/8" plug |
| CC-Link Out | 5-pin M12 socket, A-coded |
| CC-Link In | 5-pin M12 plug, A-coded |
| Degree of protection | IP 65 (with connectors) |
| Weight | 800 g |

**Electrical data**

| | |
|---|---|
| Supply voltage $V_S$ | 24 V DC ±20 % LPS Class 2 |
| Residual ripple | ≤10% |
| Current consumption | ≤ 2 A |
| Application interfaces | CC-Link, IO-Link |

**Application interfaces**

**IO-Link port M12, A-coded, female**

| | |
|---|---|
| Pin 1 | +24 V DC, 1 A |
| Pin 2 | USB+ |
| Pin 3 | 0 V / GND |
| Pin 4 | IO-Link / input max. 500 mA |
| Pin 5 | USB– |

## 5 Technical Data

**Operating conditions**

| | |
|---|---|
| Ambient temperature | 0 °C…+60 °C |
| Storage temperature | 0 °C…+60 °C |
| EMC (BIS V-6111-073-C003) | |
| – IEC EN 61000-6-2 | |
| – IEC EN 61000-4-2/4/5/6 | – Severity level 2A/3B/2B/3A |
| – IEC EN 61000-4-3 | |
| 80 MHz – 1000 MHz | – Severity level 3A |
| 1400 MHz – 2000 MHz | – Severity level 3A |
| 2000 MHz – 2700 MHz | – Severity level 2A |
| – Emission as per EN 55016-2-3 | – EN 61000-6-4 |
| Vibration/shock | EN 60068 Part 2-6/27/29 |

# 6 Startup

## 6.1 CC-Link

The BIS V-6111 processor unit and the controlling system communicate via CC-Link.
The CC-Link system consists of the following components:
– Bus master
– Bus modules/slaves (here, the BIS V-6111 processor unit)

**Important note for implementation using PLCs**
There are controllers where the CC-Link data range is not transmitted synchronously with the update of the input/output map. If more than 2 bytes of data are transmitted, a mechanism must be used that guarantees that the data in the PLC and the data in the BIS V is always the same!

**Synchronous data transmission as a setting on the master**
In this method, the bus master ensures that all of the data necessary for the respective slave is transmitted together. Usually a special software function must be used in the PLC that also controls the access between the PLC and bus master so that all data is always transmitted together.

### Device data

In order to configure the parameters for the bus master correctly based on type, the device master data for the BIS V-6111 processor unit are included in the form of a CSP+ file. The CSP+ file can be found in the processor unit's internal memory and can be retrieved from there via the USB interface.

### Station address

Each device on a CC-Link line section must have a unique station address. Therefore, before being used on the bus for the first time, it must be configured individually using the display or using the CC-Link tool.

### Process data

BIS V-6111 implements a CC-Link Remote Device Station. Depending on the project configuration, the protocol used is compatible with CC-Link 1.0/1.1 or CC-Link 2.0.

CC-Link divides the cyclical process data buffer into words (16-bit, WW for output data and WR for input data) and into points (1 bit, referred to in the CC-Link specification as RY for output data and RX for input data). The number of points and words depends on how many station addresses the device occupies. Between 1 and 4 station addresses are allowed. In addition, CC-Link 2.0 or higher allows you to use more words and points by using the extended cycles function. In doing so, the process data size can be two, four or eight times as large as when using CC-Link 1.0 or 1.1.

Only words are used for the process data of the read/write heads. The IO-Link port likewise uses words to transmit the IO-Link data in IO-Link operation. However, points are used for transmitting acyclical IO-Link data and for status data.

As a word is assigned to a head or to the IO-Link port either entirely or not at all, the size of the process data buffer is always an even number in bytes. Zero is used as the size for a head in order to disable the corresponding head. The size zero for IO-Link switches off the IO-Link port.

The size of the process data buffer for each head and for the IO-Link port is identical for input and output data.

# 6 Startup

**Baud rate**

BIS V-6111 supports all 5 CC-Link baud rates. They must be set identically on the master and slave.

**Profile**

BIS V-6111 allows you to use profiles for simplified project configuration. Profiles 0 to 8 are prepared configurations of the process data buffer. Profile 9 enables the device to be given any desired project configuration.

| Profile | Process data configuration | | Assigned stations | Extended cycles |
|---|---|---|---|---|
| 0 | Head 1 to Head 4:<br>IO-Link port:<br>CC-Link 1.1 compatible! | 4 words each<br>Disabled | 4 | - |
| 1 | Head 1<br>Head 2 to Head 4:<br>IO-Link port:<br>CC-Link 1.1 compatible! | 16 words<br>Disabled<br>Disabled | 4 | - |
| 2 | Head 1 and Head 2:<br>Head 3 and Head 4:<br>IO-Link port: | 16 words each<br>Disabled.<br>Disabled | 4 | 2 |
| 3 | Head 1 to Head 3:<br>Head 4:<br>IO-Link port: | 16 words each<br>Disabled<br>Disabled | 3 | 4 |
| 4 | Head 1 to Head 4:<br>IO-Link port: | 16 words each<br>Disabled | 4 | 4 |
| 5 | Head 1<br>Head 2 to Head 4:<br>IO-Link port: | 16 words<br>Disabled<br>8 words | 4 | 2 |
| 6 | Head 1 and Head 2:<br>Head 3 and Head 4:<br>IO-Link port: | 16 words each<br>Disabled<br>8 words | 3 | 4 |
| 7 | Head 1 to Head 3:<br>Head 4:<br>IO-Link port: | 16 words each<br>Disabled<br>8 words | 4 | 4 |
| 8 | Head 1 to Head 4:<br>IO-Link port: | 13 words each<br>8 words | 4 | 4 |
| 9 | Use the settings configured via the service inter-face. | | | |

# 6 Startup

**Project configuration settings**

A complete project configuration of BIS V-6111 includes these settings:
- Station address
- Baud rate
- Profile
- Assigned station addresses
- Extended cycles
- Process data size for head 1
- Process data size for head 2
- Process data size for head 3
- Process data size for head 4
- Process data size of the IO-Link port

Where profiles 0 to 8 define the following parameters implicitly.

The project configuration can take place using the display. Then, only the parameters of station address, baud rate and profile are available. Detailed configurations can take place via the service interface using the CC-Link tool

## 6.2 Project Configuration on the Device

The display can be used to configure the station address, the baud rate and the profile. After the configuration, the device carries out a reset to apply the values.

If a profile from 0 to 8 is selected, the parameters for assigned station addresses, extended cycles and the process data sizes are defined automatically. For Profile 9, the values are used that were configured during the last project configuration using the CC-Link tool.

## 6.3 Project Configuration Using the CC-Link Tool

The project configuration can be carried out using the CC-Link tool. The tool is already in the device's internal memory. If the BIS V-6111 is connected to a PC, the internal memory is detected as a drive, like a USB stick. To use the CC-Link tool, a driver must be installed on the PC that allows the service interface of the BIS V-6111 to be addressed as a virtual COM port. The driver is available as a 32-bit version in the Driver/x86 directory and as a 64-bit version in the Driver/x64 directory. It functions with Windows XP, Vista, 7 and 8. Administrator rights are required for installation.

If the driver is installed, the PC has an additional virtual COM port as soon as the BIS V-6111 is connected to it via USB. The CC-Link tool is in the Tools directory. Ideally, it should be copied to the PC's hard drive. No installation is required. However, the CC-Link tool requires Microsoft.net Framework 4. You may have to install it from the Microsoft website.

You can now configure the BIS V-6111 conveniently and flexibly using the CC-Link tool. It is easy to select the baud rate and station address parameters. For the process data buffer sizes, you can either choose a profile from 0 to 8 or select Profile 9, which enables the process data buffer size to be adjusted separately for each head and for the IO-Link port. The box underneath then shows all possible combinations that support the desired process data buffer. Here, select the desired combination, select the COM port to which the BIS V is connected and send the project configuration to the device using the "Send Parameters to BIS V-6111" button. BIS V-6111 takes over the values and carries out a reset.

# 6 Startup

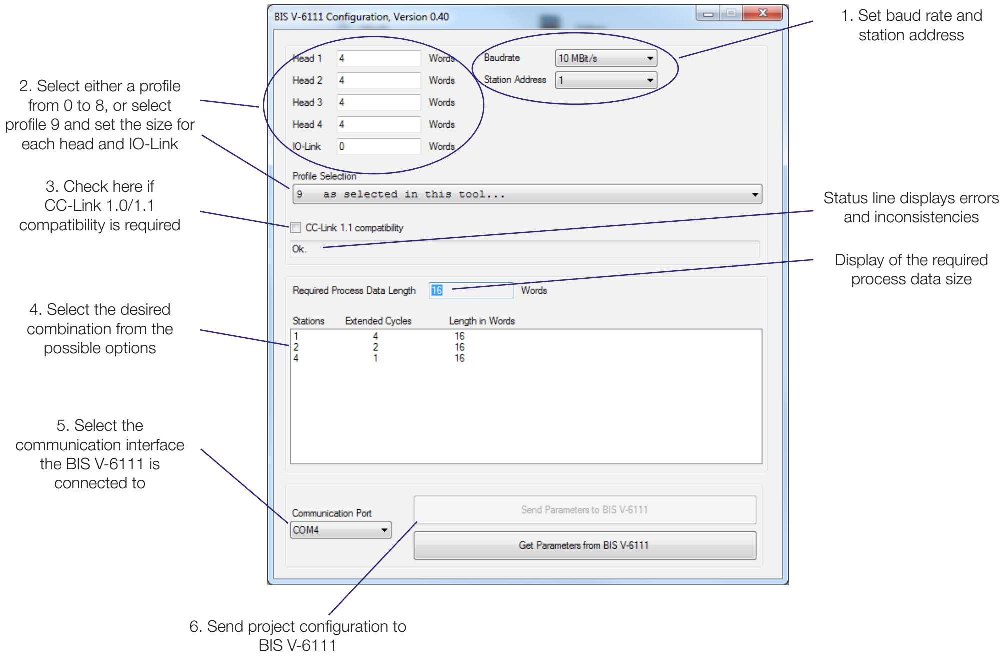



Figure 6: Project configuration

The CC-Link tool can also read back the values configured on the device. To do so, select the COM port and read out the data using the "Get Parameters from BIS V-6111" button.

# 6 Startup

## 6.4 Configuration

For operating BIS V, there are also additional parameters transmitted as a message in the CC-Link process data (see Section 7). The individual parameters are described here.

**User Parameters**

| Command | Byte | Bit | Meaning | |
|---|---|---|---|---|
| Set device parameters | 1 | 0–2 | Keyboard/LCD: read only | 0/1: Disable/enable |
| | | 3–7 | Device LEDs off | 0/1: Disable/enable |
| Set head parameters | 1 | 0 | CRC check | 0/1: Disable/enable |
| | | 1 | Dynamic mode | 0/1: Disable/enable |
| | | 2 | Type serial number | 0/1: Disable/enable |
| | | 3 | Slow tag detection | 0/1: Disable/enable |
| | | 4 | Low antenna power | 0/1: Disable/enable |
| | | 5 | Head LEDs off | 0/1: Disable/enable |
| | 2 | 0–7 | Tag type | 0: All data carriers are detected |
| | | | | 10: Mifare |
| | | | | 11: ISO 15693 |
| | | | | 20: EM4x02 |
| | | | | 21: Hitag1 |
| | | | | 22: Hitag2 |
| | | | | 30: BIS C 32 byte |
| | | | | 31: BIS C 64 byte |
| | 3 | 0–2 | UID compare count | 1–7: Number of repetitions |

**Description of individual parameters**

**Keyboard/LCD: read only**
If this function is enabled, then the CC-Link station address, profile and baud rate can no longer be changed via the display.

**Device LEDs off**
If this function is enabled, then the read/write head LEDs on the BIS V-6111 processor unit are shut off after 30 min. The parameters for this function are configured in the header module.

# 6 Startup

**CRC check**

The CRC check is a procedure for determining a check value for data in order to be able to recognize transmission errors. If the CRC check is activated, an error message is sent when a CRC error is detected.

**Checksum**
M and L system:
The checksum is written to the data carrier as 2 bytes of information. 2 bytes per block are lost. This leaves 14 bytes per block available. The usable number of bytes can be found in the following table.

C system:
The checksum is written to the data carrier as 2 bytes of information per page.
2 bytes per page are lost, i.e. the page size is 30 bytes or 62 bytes depending on the data carrier type.

The number of usable bytes thus decreases when using the checksum.

| Balluff data carrier type | Memory capacity | Usable bytes for CRC_16 |
|---|---|---|
| BIS M-1_ _-**01** | 752 bytes | 658 bytes |
| BIS M-1_ _-**02** | 2000 bytes | 1750 bytes |
| BIS M-1_ _-**03** | 112 bytes | 98 bytes |
| BIS M-1_ _-**04** | 256 bytes | 224 bytes |
| BIS M-1_ _-**05** | 224 bytes | 196 bytes |
| BIS M-1_ _-**06** | 288 bytes | 252 bytes |
| BIS M-1_ _-**07** | 992 bytes | 868 bytes |
| BIS M-1_ _-**08** | 160 bytes | 140 bytes |
| BIS M-1_ _-**09** | 32 bytes | 28 bytes |
| BIS M-1_ _-**10** | 736 bytes | 644 bytes |
| BIS M-1_ _-**11** | 8192 bytes | 7168 bytes |
| BIS M-1_ _-**13** | 32786 bytes | 28672 bytes |
| BIS M-1_ _-**14** | 65536 bytes | 57344 bytes |
| BIS M-1_ _-**15** | 131072 bytes | 114688 bytes |
| BIS M-1_ _-**20** | 8192 bytes | 7168 bytes |
| BIS L-1_ _-**01** | 192 bytes | 168 bytes |
| BIS L-2_ _-**03** | 5 bytes (read-only) | — |
| BIS L-1_ _-**05** | 192 bytes | 168 bytes |
| BIS C-1_ _-**04** | 511 bytes | 450 bytes |
| BIS C-1_ _-**05** | 1023 bytes | 930 bytes |
| BIS C-1_ _-**11** | 2047 bytes | 1922 bytes |
| BIS C-1_ _-**32** | 8192 bytes | 7936 bytes |

**Initializing**

To use the CRC check, the data carrier has to be initialized. Initialization of the data carrier is accomplished in the output buffer using the command designator $12_{hex}$. If the data carrier does not contain the correct CRC, the processor unit sets an error message in the input buffer (see Example 7 on page 50).
Data carriers as shipped from the factory can be written immediately with a checksum, since all data is set to 0.

# 6 Startup

**Dynamic mode**

As soon as the *Dynamic mode* function is enabled, the processor unit accepts the read/write job from the controlling system and stores it, regardless of whether a data carrier is in the active zone of the R/W head or not. If a data carrier enters the active range of the R/W head, the stored job is run.

> **Note**
> To achieve the read times specified on  on page 42 in dynamic operation, the "Tag type" parameter has to be set to "BIS C 32 byte" or "BIS C 64 byte" on the respective head.

**Type serial number**

If this function is enabled, the type of the read/write head as well as the data carrier type and serial number (UID = Unique Identifier) for the data carrier are output when CP occurs.
The length of the serial number can vary depending on the type of data carrier. To be able to determine the length, the data is preceded by a length field.

> **Note**
> BIS C data carriers do not have a UID value.

| Data format | 1 byte | 1 byte | 1 byte | 4 bytes or 8 bytes |
|---|---|---|---|---|
| **Meaning** | Length (number of bytes including length) | Read/write head type | Data carrier type | UID (Mifare or Hitag 1) or UID (ISO 15693) |

| BIS VM-3_ _-001-S4 | BIS VL-3_ _-001-S4 | BIS C-3… |
|---|---|---|
| 03 | 02 | 01 |

**Slow tag detection**

For this option, the antenna on the read/write head is switched on for data carrier detection only every 200 ms. The parameters for this function are configured in the respective read/write head module.

**Low antenna power**

Transmitting power is reduced when using this parameter. The parameters for this function are configured in the respective read/write head module and is reserved for future read/write heads.

**Head LEDs off**

This parameter switches off the LEDs on the respective read/write head. The parameters for this function are configured in the respective read/write head module.

# 6 Startup

**Tag type**

The following data carriers are available for the BIS V-6111 processor unit.

**Mifare data carriers (for read/write heads BIS VM):**

| Balluff data carrier type | Manufacturer | Description | Memory capacity | Memory type |
|---|---|---|---|---|
| BIS M-1_ _-**01** | NXP | Mifare Classic | 752 bytes | EEPROM |
| BIS M-1_ _-**10** | NXP | Mifare Classic | 736 bytes | EEPROM |

**ISO 15693 data carriers (for read/write heads BIS VM):**

| Balluff data carrier type | Manufacturer | Description | Memory capacity | Memory type |
|---|---|---|---|---|
| BIS M-1_ _-**02** | Fujitsu | MB89R118 | 2000 bytes | FRAM |
| BIS M-1_ _-**03** | NXP | SL2ICS20 | 112 bytes | EEPROM |
| BIS M-1_ _-**04*** | Texas Instruments | TAG-IT Plus | 256 bytes | EEPROM |
| BIS M-1_ _-**05*** | Infineon | SRF55V02P | 224 bytes | EEPROM |
| BIS M-1_ _-**06*** | EM | EM4135 | 288 bytes | EEPROM |
| BIS M-1_ _-**07** | Infineon | SRF55V10P | 992 bytes | EEPROM |
| BIS M-1_ _-**08*** | NXP | SL2ICS530 | 160 bytes | EEPROM |
| BIS M-1_ _-**09*** | NXP | SL2ICS500 | 32 bytes | EEPROM |
| BIS M-1_ _-**11** | Balluff | BIS M-1 | 8192 bytes | FRAM |
| BIS M-1_ _-**13** | Balluff | BIS M-1 | 32768 bytes | FRAM |
| BIS M-1_ _-**14** | Balluff | BIS M-1 | 65536 bytes | FRAM |
| BIS M-1_ _-**15** | Balluff | BIS M-1 | 161072 bytes | FRAM |
| BIS M-1_ _-**20** | Fujitsu | MB89R112 | 8192 bytes | FRAM |

* On request

**For read/write heads BIS VL:**

| Balluff data carrier type | Manufacturer | Description | Memory capacity | Memory type |
|---|---|---|---|---|
| BIS L-1_ _-**01** | NXP | Hitag1 | 192 bytes | EEPROM |
| BIS L-2_ _-**03** | EM | EM4x02 | 5 bytes (read-only) | – |
| BIS L-1_ _-**05** | NXP | HitagS | 192 bytes | EEPROM |

**For read/write heads BIS C (with adapter):**

| Balluff data carrier type | Manufacturer | Memory capacity | Memory type | Memory organization |
|---|---|---|---|---|
| BIS C-1_ _-**04** | Balluff | 511 bytes | EEPROM | 32-byte blocks |
| BIS C-1_ _-**05** | Balluff | 1023 bytes | EEPROM | 32-byte blocks |
| BIS C-1_ _-**11** | Balluff | 2047 bytes | EEPROM | 64-byte blocks |
| BIS C-1_ _-**32** | Balluff | 8192 bytes | FRAM | 64-byte blocks |

**Note**
The data carriers contain additional memory ranges for configuration and protected data. These ranges cannot be processed using the BIS V-6111 processor unit.

# 6 Startup

**UID Compare Count**

This parameter specifies how often the 5-byte ID of a BIS L-2_ _-03 data carrier is read in and compared before the data carrier is shown as detected. This value is set to "2" by default. For highly dynamic applications, this value can be set to "1".

## 6.5 RX/RY Point Buffer

IO-Link communication is based on the BAP-05028-J specification. The differences result from the fact that BIS V-6111 has only one IO-Link port.

CC-Link points are used to configure the IO-Link port and the bit header for the read/write head.

**1 read/write head without I/O-Link**

| Slave -> Master | | Master -> Slave | |
|---|---|---|---|
| RXm0…RXm7 | Bit header 1 | RYm0…RYm7 | Bit header 1 |

m = station address from the master

**2 read/write heads without I/O-Link**

| Slave -> Master | | Master -> Slave | |
|---|---|---|---|
| RXm0…RXm7 | Bit header 1 | RYm0…RYm7 | Bit header 1 |
| RXm8…RXmF | Bit header 2 | RYm8…RYmF | Bit header 2 |

m = station address from the master

**3 read/write heads without I/O-Link**

| Slave -> Master | | Master -> Slave | |
|---|---|---|---|
| RXm0…RXm7 | Bit header 1 | RYm0…RYm7 | Bit header 1 |
| RXm8…RXmF | Bit header 2 | RYm8…RYmF | Bit header 2 |
| RX(m+1)0…RX(m+1)7 | Bit header 3 | RY(m+1)0…RY(m+1)7 | Bit header 3 |

m = station address from the master

**4 read/write heads without I/O-Link**

| Slave -> Master | | Master -> Slave | |
|---|---|---|---|
| RXm0…RXm7 | Bit header 1 | RYm0…RYm7 | Bit header 1 |
| RXm8…RXmF | Bit header 2 | RYm8…RYmF | Bit header 2 |
| RX(m+1)0…RX(m+1)7 | Bit header 3 | RY(m+1)0…RY(m+1)7 | Bit header 3 |
| RX(m+1)8…RX(m+1)F | Bit header 4 | RY(m+1)8…RY(m+1)F | Bit header 4 |

m = station address from the master

# 6 Startup

**RX/RY Point Buffer (continued)**

**2 read/write heads with I/O-Link**

| Slave -> Master | | Master -> Slave | |
|---|---|---|---|
| RXm0 | Input pin 4 | RYm0 | Output pin 4 |
| RXm1…RXmF | Not used | RYm1…RYmF | Not used |
| RX(m+1)0 | Diagnostic pin 4 | RY(m+1)0 | Port direction |
| RX(m+1)1…RX(m+1)F | Not used | RY(m+1)1…RY(m+1)F | Not used |
| RX(m+2)0 | Diagnostic pin 1 | | |
| RX(m+2)1…RX(m+2)F | Not used | RY(m+2)1…RY(m+2)F | Not used |
| RX(m+3)0 | IO-Link port valid | RY(m+3)0 | IO-Link port enable |
| RX(m+3)1…RX(m+3)7 | Not used | RY(m+3)1…RY(m+3)F | Not used |
| RX(m+3)8 | IO-Link event flag | RY(m+3)8 | IO-Link event clear |
| RX(m+3)9…RX(m+3)F | Not used | RY(m+3)9…RY(m+3)F | Not used |
| RX(m+4)0…RX(m+4)7 | Bit header 1 | RY(m+4)0…RY(m+4)7 | Bit header 1 |
| RX(m+4)8…RX(m+4)F | Bit header 2 | RY(m+4)8…RY(m+4)F | Bit header 2 |

m = station address from the master

**4 read/write heads with I/O-Link**

| Slave -> Master | | Master -> Slave | |
|---|---|---|---|
| RXm0 | Input pin 4 | RYm0 | Output pin 4 |
| RXm1…RXmF | Not used | RYm1…RYmF | Not used |
| RX(m+1)0 | Diagnostic pin 4 | RY(m+1)0 | Port direction |
| RX(m+1)1…RX(m+1)F | Not used | RY(m+1)1…RY(m+1)F | Not used |
| RX(m+2)0 | Diagnostic pin 1 | | |
| RX(m+2)1…RX(m+2)F | Not used | RY(m+2)1…RY(m+2)F | Not used |
| RX(m+3)0 | IO-Link port valid | RY(m+3)0 | IO-Link port enable |
| RX(m+3)1…RX(m+3)7 | Not used | RY(m+3)1…RY(m+3)F | Not used |
| RX(m+3)8 | IO-Link event flag | RY(m+3)8 | IO-Link event clear |
| RX(m+3)9…RX(m+3)F | Not used | RY(m+3)9…RY(m+3)F | Not used |
| RX(m+4)0…RX(m+4)7 | Bit header 1 | RY(m+4)0…RY(m+4)7 | Bit header 1 |
| RX(m+4)8…RX(m+4)F | Bit header 2 | RY(m+4)8…RY(m+4)F | Bit header 2 |
| RX(m+5)0…RX(m+5)7 | Bit header 3 | RY(m+5)0…RY(m+5)7 | Bit header 3 |
| RX(m+5)8…RX(m+5)F | Bit header 4 | RY(m+5)8…RY(m+5)F | Bit header 4 |

m = station address from the master

## 6 Startup

**RX/RY Point Buffer (continued)**

Buffer size depending on the assigned stations and the extended cyclic setting.

| | Extended cycles | | | |
|---|---|---|---|---|
| | **Single** | **Double** | **Quadruple** | **Eightfold** |
| Assigns 1 station | 32 points<br>4 words | 32 points<br>8 words | 64 points<br>16 words | 128 points<br>32 words<br>Profile(0/1) |
| Assigns 2 stations | 64 points<br>8 words | 96 points<br>16 words | 192 points<br>32 words | 384 points<br>64 words |
| Assigns 3 stations | 96 points<br>12 words | 160 points<br>24 words | 320 points<br>48 words<br>Profile(3/6) | 640 points<br>96 words |
| Assigns 4 stations | 128 points<br>16 words<br>Profile(0/1) | 224 points<br>32 words<br>Profile(2/5) | 448 points<br>64 words<br>Profile(4/7/8) | 896 points<br>128 words |

For the meaning of the individual points:

**Input pin 4**
If Pin 4 is configured as the input, this bit shows the status of the input.

**Diagnostic pin 4**
Pin 4 error. Short-circuit or overcurrent.

**Port direction**
If Pin 4 is operated in SIO mode, this bit defines whether it is the input (0).

**Diagnostic pin 1**
Pin 1 (supply voltage) error. Short-circuit or overcurrent.

**IO-Link port valid**
This bit is 1 if IO-Link communication with an IO-Link device is in progress and the device has not set the PDinvalid flag.

**IO-Link port enabled**
Controls whether the port works in IO-Link mode (1) or in SIO mode (0).

**IO-Link event flag**
The connected IO-Link device signals an event. When all pending events have been retrieved using the message transfer function, the bit is deleted automatically.

**IO-Link event clear**
If this bit is set to 1, all pending events in the IO-Link device are deleted. As long as the bit remains at 1, new events are deleted immediately also.

**Bit header for read/write heads**
Control and status bits for heads 1…4.

The last 16 points are reserved for the system area. The implementation is in accordance with CC-Link specifications BAP-05026-J and BAP-05028-J.

## 6 Startup

**RX/RY Point Buffer (continued)**

| Slave -> Master | | Master -> Slave | |
|---|---|---|---|
| RX(m+n)0 | Not used | RY(m+n)0 | Not used |
| RX(m+n)1 | | RY(m+n)1 | |
| RX(m+n)2 | | RY(m+n)2 | |
| RX(m+n)3 | | RY(m+n)3 | |
| RX(m+n)4 | Message transmission size | RY(m+n)4 | |
| RX(m+n)5 | | RY(m+n)5 | |
| RX(m+n)6 | | RY(m+n)6 | |
| RX(m+n)7 | Not used | RY(m+n)7 | |
| RX(m+n)8 | Initial data processing request | RY(m+n)8 | Initial data processing |
| RX(m+n)9 | Initial data setting completion flag | RY(m+n)9 | Initial data setting request flag |
| RX(m+n)A | Error state flag | RY(m+n)A | Error reset request flag |
| RX(m+n)B | Remote state flag | RY(m+n)B | Not used |
| RX(m+n)C | Message transmission | RY(m+n)C | Message transmission request flag |
| RX(m+n)D | Message handshake flag | RY(m+n)D | Message handshake flag |
| RX(m+n)E | Not used | RY(m+n)E | Not used |
| RX(m+n)F | | RY(m+n)F | |

m = Station address
n = The value depends on the number of assigned station addresses and the "Extended Cycles".

| Assigned station addresses (occupied stations) | Extended cycles | n |
|---|---|---|
| 2 | No | $3_{hex}$ |
| 2 | 2 | $5_{hex}$ |
| 2 | 4 | $B_{hex}$ |
| 2 | 8 | $17_{hex}$ |
| 3 | No | $5_{hex}$ |
| 3 | 2 | $9_{hex}$ |
| 3 | 4 | $13_{hex}$ |
| 3 | 8 | $27_{hex}$ |
| 4 | No | $7_{hex}$ |
| 4 | 2 | $D_{hex}$ |
| 4 | 4 | $1B_{hex}$ |
| 4 | 8 | $37_{hex}$ |

### 6.6 IO-Link MTA

The Message transmission size is always 4 words (8 bytes). To transmit messages to the IO-Link device, the Message Transfer Area in the cyclical process data buffer is used. The protocol used makes it possible to transmit messages that are larger than the Message Transfer Area.

At the start of the transmission, the master sets RY(m+n)C, the Message transmission request flag. BIS V-6111 acknowledges this by setting RX(m+n)C Message Transmission. The master can now copy the first 8 bytes of the message into the Message Transmission Area and set the RY(m+n)D Message handshake flag. BIS V-6111 reads out the Message Transmission Area, writes its response into it and sets the RX(m+n)D Message handshake flag. If another data block is to be transmitted, the master can now copy it to the Message Transmission Area and invert the Message handshake flag. BIS V-6111 acknowledges it by inverting its Message handshake flag. When the message transmission is complete, the master sets RY(m+n)C to 0, which BIS V-6111 acknowledges by also setting RX(m+n)C to 0.

**Message Structure**

| Read request | | Write request | |
|---|---|---|---|
| Block number | L | Block number | L |
| Subcommand type | H | Subcommand type | H |
| Division number | L | Division number | L |
| Data size | H | Data size | H |
| Request data | | Request data | |
| Sum check | | Sum check | |

| Read response | | Write response | |
|---|---|---|---|
| Block number | L | Block number | L |
| Subcommand type | H | Subcommand type | H |
| Return status | L | Return status | L |
| | H | | H |
| Division number | L | Division number | L |
| Data size | H | Data size | H |
| Response data | | Response data | |
| Sum check | | Sum check | |

# 6 Startup

**IO-Link MTA (continued)**

The following is a description of the messages that can be transmitted in the Message Transfer Area.

BIS V-6111 processes Identification data directly:

| Byte No. | Item | Identification data | Process data size |
|---|---|---|---|
| Byte 0 | Channel | 0x10 | 0x20 |
| Byte 1 | Length | 0-64 | 1 |
| Byte 2 | Control/Status | Read | Read |
| Byte 3 | Index | Index | 0 |
| Byte 4 | Message data | Request/Response Data (length = 0-64 bytes) | IO-Link port 0 |
| Byte 5 | | | Unused |
| Byte 6 | | | |
| Byte 7 | | | |
| Byte 8 | | | |
| Byte 9 | | | |
| Byte 10 | | | |
| Byte 11 | | | |
| Byte 12 | | | |
| … | | | |
| Byte 252 | | | |

The index defines which information is read.

| Index | Description |
|---|---|
| 0x00 … 0x0A | Unused |
| 0x10 | Vendor name |
| 0x11 | Vendor text |
| 0x12 | Product name |
| 0x13 | Product code |
| 0x14 | Product text |
| 0x15 | Serial Number |
| 0x16 | Hardware revision |
| 0x17 | Firmware revision |
| 0x18 … 0xFF | Unused |

# 6 Startup

**IO-Link MTA (continued)**

The following messages affect the connected IO-Link device:

| Byte No. | Item | Identification data | | | |
|---|---|---|---|---|---|
| | | ISDU | Event data | Validation data | Data storage |
| Byte 0 | Channel | 0x30 | 0x31 | 0x32 | 0x33 |
| Byte 1 | Length | 0-232 | 4 | 24 | 2 |
| Byte 2 | Control/ Status | Read/write | Read/- | Read/write | Read/write |
| Byte 3 | Index | Port Number | Port Number | Port Number | Port Number |
| Byte 4 | Message data | Index_L | Event Qualifier | Validation Type | Configuration |
| Byte 5 | | Index_H | Reserved (Fixed to 0) | Reserved (Fixed to 0) | Reserved (Fixed to 0) |
| Byte 6 | | Subindex | EventCode_L | Vendor ID_1 | Unused |
| Byte 7 | | Reserved (Fixed to 0) | EventCode_H | Vendor ID_2 | |
| Byte 8 | | Request/ Response Data (length = 0-232 bytes) | Unused | Device ID_1 | |
| Byte 9 | | | | Device ID_2 | |
| Byte 10 | | | | Device ID_3 | |
| Byte 11 | | | | Reserved (Fixed to 0) | |
| Byte 12 | | | | Serial number 1 | |
| … | | | | … | |
| Byte 27 | | | | Serial number 16 | |
| … | | | | Unused | |
| Byte 252 | | | | | |

# 6 Startup

**IO-Link MTA (continued)**

The following table describes the individual fields of the messages:

| Item | Description |
|---|---|
| Channel | Data selects the access area |
| Length | The length of the message data |
| Control/Status | For request message:<br>0x02 = write<br>0x03 = read<br>For response message:<br>0x00 = OK<br>0xF0 = error |
| Index/port no. | Index = address of the gateway identification data (for details refer to the next table)<br>Port Number = IO-Link port number |
| Identification data | Identification data of the gateway, for details refer to the following "Description of the gateway identification data index" |
| Process data size | The process data size setting for each IO-Link port can be read and written. The value shows the mapped process data size for each IO-Link port in words (0 - 16).<br>The sum of all IO-Link port's data size plus the message transmission area must not exceed the maximum available word area, which is determined by the number of occupied stations and extended cyclic setting. |
| ISDU | Reading/writing IO-Link parameter data |
| Event data | The event data (event qualifier and event code) of a pending event indicated by the IO-Link port event flag can be read. After reading the event data the IO-Link port event flag changes to 0. |
| Validation data | IO-Link device validation. The validation type is defined as follows:<br>0x00 = validation deactivated<br>0x01 = validation of IO-Link Vendor ID and IO-Link Device ID<br>0x02 = validation of IO-Link Vendor ID, IO-Link Device ID and serial number<br>Depending on the configuration of the IO-Link device validation the connected device's information is verified and the result indicated by the port valid bit. |
| Data storage | For configuration of the data storage function of the IO-Link master.<br>The configuration byte is defined as follows:<br>0x81 = data storage upload enable (IO-Link device -> IO-Link master)<br>0x82 = data storage download enable (IO-Link master -> IO-Link device)<br>0x83 = data storage upload/download enable |

# 7 Device Function

## 7.1 Function Principle of the BIS V-6111

Two buffers are needed to exchange data and commands between the processor unit and the controlling system (input buffer and output buffer). The buffer contents are exchanged using cyclical polling. The buffer content depends on the cycle in which it is written (e.g. control commands at the beginning of a job).
When writing to the buffer, the transmitted data from the preceding cycle is overwritten. Unwritten words are not deleted and retain their data content.

**Example:**
Total buffer size 44 words (4 × 8 words: heads H1…H4, 8 words: IO-Link), 4 words Message Transfer Area

| |
|---|
| H1<br>0x00…0x07 |
| H2<br>0x08…0x0F |
| H3<br>0x10…0x17 |
| H4<br>0x18…0x1F |
| IO-Link<br>0x20…0x28 |
| MTA<br>0x2C…0x2F |

The buffer size of the total buffer results from the sum of all buffers.

The process data buffer is divided into multiple ranges:
– Range 1…4 = read/write heads 1…4 (H1…H4)
– Range 5 = IO-Link
– Message Transfer Area

The size of these areas can be configured.

**IO-Link**

IO-Link data is transmitted unchanged to the IO-Link slaves via the IO-Link master. IO-Link buffer: 0…32 bytes (max.)

## 7.2 Process Data Buffer

### Output buffer

**Bit header**
If the IO-Link function is disabled, the bit header from head 1 has the CC-Link point address RYm0…RYm7. If the IO-Link is enabled, the point address RY(m+4)0…RY(m+4)7 and the number of points has to be greater than or equal to 96.

| Bit No. | Bit header |
|---|---|
| 0 | AV |
| 1 | – |
| 2 | GR |
| 3 | – |
| 4 | – |
| 5 | KA |
| 6 | TI |
| 7 | – |

**Point address of bit header 1–4**

| | Head 1 | Head 2 | Head 3 | Head 4 |
|---|---|---|---|---|
| **IO-Link disable** | RYm0…RYm7 | RYm8…RYmF | RY(m+1)0… RY(m+1)7 | RY(m+1)8… RY(m+1)F |
| **IO-Link enable** | RY(m+4)0… RY(m+4)7 | RY(m+4)8… RY(m+4)F | RY(m+5)0… RY(m+5)7 | RY(m+5)8… RY(m+5)F |

**Assignment and explanation**

| Subaddress | Bit name | Meaning | Function description |
|---|---|---|---|
| RYm0 | AV | Job | A job is present. |
| RYm2 | GR | Basic state | Cancels the current job for this R/W head and puts the channel into a basic state. The R/W head can then be used again once GR = 0 and the controller has acknowledged this with BB = 1. The CP and MT of the input buffer (see page 37) are 0. |
| RYm5 | KA | Head shutoff | Shuts off the R/W head's antenna. Tag detection no longer takes place. The CP and MT of the input buffer (see page 37) are 0. |
| RYm6 | TI | Toggle bit in | Controller is ready to receive additional data (read job). |

**Command structure**

**Command designator $0000_{hex}$: No command present**

| Subaddress | Meaning | Function description |
|---|---|---|
| $0000_{hex}$ | Command designator | $0000_{hex}$: No command present. |
| … | None | No meaning |

**Command designator $0001_{hex}$: Read from data carrier**

| Subaddress | Meaning | Function description |
|---|---|---|
| $0000_{hex}$ | Command designator | $0001_{hex}$: Read from data carrier. |
| $0001_{hex}$ | Start address | Start address for reading. |
| $0002_{hex}$ | Number of words | Number of words to be read starting from the start address. |

**Command designator $0002_{hex}$: Write to data carrier**

| Subaddress | Meaning | Function description |
|---|---|---|
| $0000_{hex}$ | Command designator | $0002_{hex}$: Write to data carrier. |
| $0001_{hex}$ | Start address | Start address to be written from. |
| $0002_{hex}$ | Number of words | Number of words to be written starting from the start address. |

Data is accepted from the processor unit only after the command has been accepted by the processor unit and acknowledged.

| | | |
|---|---|---|
| $0000_{hex}$ | Data | Transmission of the data that is to be written to the data carrier. |
| $0001_{hex}$ | Data | Transmission of the data that is to be written to the data carrier. |
| … | Data | Transmission of the data that is to be written to the data carrier. |

**Command designator $0003_{hex}$: Display output**

The display output function allows a text to be output on the display. The text remains in place until a button is pressed and held down. A maximum of 24 characters can be output. The line break is inserted automatically after 8 characters. Space characters are allowed. A zero discontinues the output.

| Subaddress | Meaning | Function description |
|---|---|---|
| $0000_{hex}$ | Data | $0003_{hex}$: Display output |
| $0001_{hex}$ | Command designator | Characters for display output. |
| … | Data | Characters for display output. |

**Command designator $0007_{hex}$: Store the start address for the "Auto Read" function**

| Subaddress | Meaning | Function description |
|---|---|---|
| $0000_{hex}$ | Command designator | $0007_{hex}$: Store the start address for the "Auto Read" function in EEPROM. |
| $0001_{hex}$ | Start address | Address for the "Auto Read" function starting from which the data carrier is read. The value is stored in the EEPROM. |

**Command designator $0009_{hex}$: Type and serial number**

| Subaddress | Meaning | Function description |
|---|---|---|
| $0000_{hex}$ | Command designator | $0009_{hex}$: Read the read/write head type, data carrier type and UID (unique identifier) of a data carrier in the field (for data format, see page 20). |

**Command designator $0011_{hex}$: Copy data between data carriers**

| Subaddress | Meaning | Function description |
|---|---|---|
| $0000_{hex}$ | Command designator | $0011_{hex}$: Copy data between data carriers |
| $0001_{hex}$ | Source start address | Copy the start address of the source data carrier for the function from which copying is to start. |
| $0002_{hex}$ | Target start address | Copy the start address of the target data carrier for the function from which copying is to start. |
| $0003_{hex}$ | Number of words | Number of words to be copied starting from the source start address. |
| $0004_{hex}$ | Target R/W head number | Number of the read/write head that the target data carrier is in front of. |

**Command designator $0012_{hex}$ : Initialize CRC_16 data check**

| Subaddress | Meaning | Function description |
|---|---|---|
| $0000_{hex}$ | Command designator | $0012_{hex}$: Initialize CRC_16 data check |
| $0001_{hex}$ | Start address | Start address from which the CRC_16 data check is to be carried out. |
| $0002_{hex}$ | Number of words | Number of words for which a CRC_16 data check is to be carried out from the start address (low word). |

# 7 Device Function

**Command designator 0032$_{hex}$: Write constant value to data carrier**

| Subaddress | Meaning | Function description |
|---|---|---|
| 0000$_{hex}$ | Command designator | 0032$_{hex}$: Write constant value to data carrier. |
| 0001$_{hex}$ | Start address | Start address to be written from. |
| 0002$_{hex}$ | Number of words | Number of words to be written starting from the start address. |

Data is accepted from the processor unit only after the command has been accepted by the processor unit and acknowledged.

| | | |
|---|---|---|
| 0000$_{hex}$ | Data | Value that is to be written to the data carrier. |
| 0001$_{hex}$ | None | No meaning |

**Command designator 0051$_{hex}$ : Set device parameters**

| Subaddress | Meaning | Function description |
|---|---|---|
| 0000$_{hex}$ | Command designator | 0051$_{hex}$: Set device parameters |
| 0001$_{hex}$ | Device parameter | Bit 0: Display / Keyboard read-only, if 1<br>Bit 1: Energy saving function of LEDs active, if 1 |

**Command designator 0052$_{hex}$ : Set head parameters**

| Subaddress | Meaning | Function description |
|---|---|---|
| 0000$_{hex}$ | Command designator | 0052$_{hex}$: Set head parameters |
| 0001$_{hex}$ | Head parameter | If the corresponding bit is set to 1, the function is active:<br>Bit 0: CRC<br>Bit 1: Dynamic mode<br>Bit 2: Output type and serial numbers if Tag-Present<br>Bit 3: Energy saving function – slow data carrier detection<br>Bit 4: Energy saving function – reduced transmitting power<br>Bit 5: Energy saving function – shut off LEDs |
| 0002$_{hex}$ | Data carrier type | 1: All data carriers are detected<br>10: Mifare<br>11: ISO 15693<br>20: EM4x02<br>21: HItag1<br>22: HItagS<br>30: BIS C 32 Byte<br>31: BIS C 64 Byte |
| 0003$_{hex}$ | Repeat counter | Bits 0…2 Repeat counter for L-heads. |

# 7 Device Function

**Command designator 0081$_{hex}$ : Read data carrier with 24-bit address assignment**

| Subaddress | Meaning | Function description |
|---|---|---|
| 0000$_{hex}$ | Command designator | 0081$_{hex}$: Read data carrier |
| 0001$_{hex}$ | Start address (low word) | Start address for reading. |
| 0002$_{hex}$ | Start address (high word) | Start address for reading. |
| 0003$_{hex}$ | Number of words (low word) | Number of words to be read starting from the start address. |
| 0004$_{hex}$ | Number of words (high word) | Number of words to be read starting from the start address. |

**Command designator 0082$_{hex}$ : Write data carrier with 24-bit address assignment**

| Subaddress | Meaning | Function description |
|---|---|---|
| 0000$_{hex}$ | Command designator | 0082$_{hex}$: Write to data carriers |
| 0001$_{hex}$ | Start address (low word) | Start address to be written from |
| 0002$_{hex}$ | Start address (high word) | Start address to be written from |
| 0003$_{hex}$ | Number of words (low word) | Number of words to be written starting from the start address. |
| 0004$_{hex}$ | Number of words (high word) | Number of words to be written starting from the start address. |

**Command designator 0087$_{hex}$ : Saving the start address for the Auto Read function with 24-bit address assignment**

| Subaddress | Meaning | Function description |
|---|---|---|
| 0000$_{hex}$ | Command designator | 0087$_{hex}$: Store the start address for the "Auto Read" function in EEPROM. |
| 0001$_{hex}$ | Start address (low word) | Address for the "Auto Read" function starting from which the data carrier is read. The value is stored in the EEPROM. |
| 0002$_{hex}$ | Start address (high word) | Address for the "Auto Read" function starting from which the data carrier is read. The value is stored in the EEPROM. |

**Command designator 0091$_{hex}$ : Copy data between data carriers with 24-bit address assignment**

| Subaddress | Meaning | Function description |
|---|---|---|
| 0000$_{hex}$ | Command designator | 0091$_{hex}$: Copy data |
| 0001$_{hex}$ | Source start address (low word) | Copy the start address of the source data carrier for the function from which copying is to start. |
| 0002$_{hex}$ | Source start address (high word) | Copy the start address of the source data carrier for the function from which copying is to start. |
| 0003$_{hex}$ | Target start address (low word) | Copy the start address of the target data carrier for the function from which copying is to start. |
| 0004$_{hex}$ | Target start address (high word) | Copy the start address of the target data carrier for the function from which copying is to start. |
| 0005$_{hex}$ | Number of words (low word) | Number of words to be copied starting from the source start address. |
| 0006$_{hex}$ | Number of words (high word) | Number of words to be copied starting from the source start address. |
| 0007$_{hex}$ | Target R/W head number | Number of the read/write head that the target data carrier is in front of. |

**Command designator 0092$_{hex}$ : Initialize CRC_16 data check with 24-bit address assignment**

| Subaddress | Meaning | Function description |
|---|---|---|
| 0000$_{hex}$ | Command designator | 0092$_{hex}$: Initialize CRC_16 data check. |
| 0001$_{hex}$ | Start address (low word) | Start address from which the CRC_16 data check is to be carried out. |
| 0002$_{hex}$ | Start address (high word) | Start address from which the CRC_16 data check is to be carried out. |
| 0003$_{hex}$ | Number of words (low word) | Number of words for which a CRC_16 data check is to be carried out from the start address (low word). |
| 0004$_{hex}$ | Number of words (high word) | Number of words for which a CRC_16 data check is to be carried out from the start address (high word). |

**Command designator 00B2$_{hex}$ : Write constant value to data carrier with 24-bit address assignment**

| Subaddress | Meaning | Function description |
|---|---|---|
| 0000$_{hex}$ | Command designator | 00B2$_{hex}$: Write to data carrier. |
| 0001$_{hex}$ | Start address (low word) | Start address to be written from. |
| 0002$_{hex}$ | Start address (high word) | Start address to be written from. |
| 0003$_{hex}$ | Number of words (low word) | Number of words to be written. |
| 0004$_{hex}$ | Number of words (high word) | Number of words to be written. |

# 7 Device Function

**Input buffer**

**Bit header**

If the IO-Link function is disabled, the bit header from head 1 has the CC-Link point address RXm0…RXm7. If the IO-Link is enabled, the point address RX(m+4)0…RX(m+4)7 and the number of points has to be greater than or equal to 96.

| Bit No. | Header |
|---|---|
| 0 | CP |
| 1 | AA |
| 2 | AE |
| 3 | AF |
| 4 | MT |
| 5 | TO |
| 6 | HF |
| 7 | BB |

**Point address of bit header 1–4**

| | Head 1 | Head 2 | Head 3 | Head 4 |
|---|---|---|---|---|
| **IO-Link disable** | RXm0…RXm7 | RXm8…RXmF | RX(m+1)0… RX(m+1)7 | RX(m+1)8… RX(m+1)F |
| **IO-Link enable** | RX(m+4)0… RX(m+4)7 | RX(m+4)8… RX(m+4)F | RX(m+5)0… RX(m+5)7 | RX(m+5)8… RX(m+5)F |

**Assignment and explanation**

| Subaddress | Bit name | Meaning | Function description |
|---|---|---|---|
| RXm0 | CP | Code Present | A data carrier has been detected. |
| RXm1 | AA | Job start | A job was detected and started. |
| RXm2 | AE | Job end | A job was completed without errors. |
| RXm3 | AF | Job error | A job was processed incorrectly or was canceled. |
| RXm4 | MT | Multiple Tag | More than 1 data carrier is in the R/W head's field. |
| RXm5 | TO | Toggle bit out | Read: Additional data is being provided by the identification system.<br>Write: Identification system can accept additional data. |
| RXm6 | HF | Head error | Cable break to the R/W head. |
| RXm7 | BB | Ready | After powering up or after a reset via the GR bit (of the output buffer, see page 31) the BB bit indicates that the corresponding channel is ready. |

# 7 Device Function

The structure of the process data buffer is identical for all commands.

| Subaddress | Meaning | Function description |
|---|---|---|
| $0000_{hex}$ | Data | Transmission of data that was read from the data carrier. |
| $0001_{hex}$ | Data | Transmission of data that was read from the data carrier. |
| $0002_{hex}$ | Data | Transmission of data that was read from the data carrier. |
| … | Data | Transmission of data that was read from the data carrier. |

In case of error:

| Subaddress | Meaning | Function description |
|---|---|---|
| $0000_{hex}$ | Error | Error code |

### Status codes

**Note**
Status codes are only valid in connection with the AF bit!

| Status code | Function description |
|---|---|
| $0000_{hex}$ | Everything OK |
| $0001_{hex}$ | Job cannot be run because there is no data carrier in range of the read/write head. |
| $0002_{hex}$ | Cannot read the data carrier. |
| $0003_{hex}$ | Data carrier was removed from the R/W head's range during reading. |
| $0004_{hex}$ | Cannot write to the data carrier. |
| $0005_{hex}$ | Data carrier was removed from the R/W head's range during writing. |
| $0007_{hex}$ | No or invalid command designator with set AV bit (of the output buffer, see page 31) or the number of bytes is $00_{hex}$. |
| $0009_{hex}$ | R/W head cable break or no R/W head connected. |
| $000D_{hex}$ | Communication to the R/W head disrupted. |
| $000E_{hex}$ | CRC for the read data and CRC for the data carrier do not agree. |
| $000F_{hex}$ | 1st and 2nd bit header are unequal. The 2nd bit header must be used. |
| $0020_{hex}$ | Address assignment of the read/write job is outside the memory range of the data carrier. |
| $0021_{hex}$ | This function is not possible for this data carrier. |

**Description of the Code Present (CP) and Multiple Tag (MT) bits**

| CP | MT | Meaning |
|---|---|---|
| 0 | 0 | No tag in the field |
| 1 | 0 | Exactly one tag in the field. Automatic reading is OK (if configured). |
| 0 | 1 | More than one data carrier is in the field. They cannot be processed. |
| 1 | 1 | Does not occur. |

**Communication**

Communication between the controlling system and processor unit is defined by a sequence protocol. Communication between controlling system and processor unit is implemented using the control bit in the RX/RY-point buffer.

**Basic sequence**

1. The controller sends a command designator to the processor unit in the output buffer with the AV bit (RYm0) set.
   The AV bit tells the processor unit that a job is beginning and the transmitted data is valid.
2. The processor unit accepts the job and confirms the job by setting the AA bit (RXm1) in the input buffer.
3. If additional data needs to be exchanged for the job, readiness for additional data exchange is indicated by inverting the TI (RYm6) and TO (RXm5) toggle bits.
4. The processor unit has correctly executed the job and sets the AE bit (RXm2) in the RX buffer.
5. The controller has accepted all data. The AV bit (RYm0) in the RX buffer is reset.
6. The processor unit resets all the control bits set in the RX buffer during the job (AA bit, AE bit). The processor unit is ready for the next job.

**Read/write times**

**Note**
All specifications are typical values. Deviations are possible depending on the application and combination of R/W head and data carrier.
The specifications apply to static operation; no CRC_16 data checking.
All specified write/read times refer to communication between the data carrier and read/write head.
The times for data communication between the processor unit and the controlling system are not included.

For read/write heads BIS VM

**Mifare**

| Read times Data carrier with 16 bytes per block | |
|---|---|
| Data carrier detection | ≤ 20 ms |
| Read bytes 0 to 15 | ≤ 25 ms |
| for each additional start of a 16-byte block | ≤ 10 ms |

| Write times Data carrier with 16 bytes per block | |
|---|---|
| Data carrier detection | ≤ 20 ms |
| Write bytes 0 to 15 | ≤ 60 ms |
| for each additional start of a 16-byte block | ≤ 30 ms |

# 7 Device Function

**ISO 15693**

| Read times Data carrier with 16 bytes per block | |
|---|---|
| Data carrier detection | ≤ 20 ms |
| Read bytes 0 to 15 | ≤ 25 ms |
| for each additional start of a 16-byte block | ≤ 10 ms |

| Write times Data carrier with 16 bytes per block | | |
|---|---|---|
| | FRAM (BIS M-1_ _-02/20) | EEPROM (BIS M-1_ _-03/07/08) |
| Data carrier detection | ≤ 20 ms | ≤ 20 ms |
| Write bytes 0 to 15 | ≤ 60 ms | ≤ 80 ms |
| for each additional start of a 16-byte block | ≤ 25 ms | ≤ 80 ms |

**High speed***

| Read times Data carrier with 64 bytes per block | |
|---|---|
| Data carrier detection | ≤ 20 ms |
| Read bytes 0 to 63 | ≤ 14 ms |
| for each additional start of a 64-byte block | ≤ 6 ms |

| Write times Data carrier with 64 bytes per block | |
|---|---|
| Data carrier detection | ≤ 20 ms |
| Write bytes 0 to 63 | ≤ 30 ms |
| for each additional start of a 64-byte block | ≤ 15 ms |

*These times apply only to the combination of read/write head BIS VM-3_ _-**4**01-S4 with BIS M-1_ _-**11**/A, BIS M-1_ _-**13**/A, BIS M-1_ _-**14**/A , or BIS M-1_ _-**15**/A data carriers.

For read/write heads BIS VL

**Read times**

| Data carrier with 16 bytes per block | BIS L-1_ _ |
|---|---|
| Data carrier detection | ≤ 110 ms |
| Read bytes 0 to 15 | ≤ 175 ms |
| for each additional start of a 16-byte block | ≤ 40 ms |

**Data carrier BIS L-2_ _**
Data carrier detection + Read data carrier ≤ 140 ms

**Write times**

| Data carrier with 16 bytes per block | BIS L-1_ _ |
|---|---|
| Data carrier detection | ≤ 110 ms |
| Write bytes 0 to 15 | ≤ 285 ms |
| for each additional start of a 16-byte block | ≤ 100 ms |

**Data carrier BIS L-2_ _**
Writing not possible

# 7 Device Function

For read/write heads BIS C

**Read times in static mode**

| Data carrier with 32 bytes per block | |
|---|---|
| **No. of bytes** | **Read time [ms]** |
| 0 to 31 | 110 |
| for each additional start of a 32-byte block | 120 |

| Data carrier with 64 bytes per block | |
|---|---|
| **No. of bytes** | **Read time [ms]** |
| 0 to 63 | 220 |
| for each additional start of a 64-byte block | 230 |

**Write times in static mode**

| Data carrier with 32 bytes per block | |
|---|---|
| **No. of bytes** | **Read time [ms]** |
| 0 to 31 | 110 + n * 10 |
| ≥ 32 bytes | y * 120 + n * 10 |

| Data carrier with 64 bytes per block | |
|---|---|
| **No. of bytes** | **Read time [ms]** |
| 0 to 63 | 220 + n * 10 |
| ≥ 64 bytes | Y * 230 + n * 10 |

n = Number of contiguous bytes to write
y = Number of blocks to process

Example: 17 bytes should be written starting at address 187. Data carrier = 32 bytes per block. Blocks 5 and 6 are processed, since the start address 187 is in block 5 and end address 203 is in block 6.
t = 2 * 120 + 17 * 10 = 410

**Read times within the first block in dynamic mode**

| Data carrier with 32 bytes per block | |
|---|---|
| **No. of bytes** | **Read time [ms]** |
| 0 to 3 | 14 |
| for each additional byte | 3.5 |
| 0 to 31 | 112 |

| Data carrier with 64 bytes per block | |
|---|---|
| **No. of bytes** | **Read time [ms]** |
| 0 to 3 | 14 |
| for each additional byte | 3.5 |
| 0 to 63 | 224 |

m = Highest address to read
Formula: t = (m + 1) * 3.5 ms

Example: Read 11 bytes starting at address 9. This means that the largest address to be read is 19. This yields 70 ms.

**Note**
Dynamic operation with BIS C: The times indicated apply after the data carrier has been detected. Otherwise 45 ms must be added for powering up until the data carrier is recognized. To achieve the read times specified in dynamic operation, the Tag Type parameter has to be set to "BIS C 32 Byte" or "BIS C 64 Byte" on the respective head.

## 7.3 Function Indicator

The operating states of the identification system, the Profibus DP interface and the IO-Link master are displayed using LEDs.

**Overview of display elements**

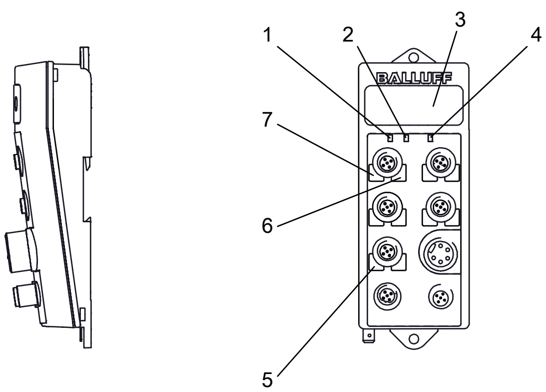



Figure 7: Function indicators

**1** Ready device (RD)
**2** Ready (RD) CC-Link
**3** Display
**4** Error (ERR) CC-Link
**5** Service/IO-Link
**6** COM R/W head
**7** RD R/W head

# 7 Device Function

**Device LEDs**

| Display | Function | | |
|---|---|---|---|
| | **Ready device (RD) (Green)** | **Ready (RD) (Green)** | **Error (ERR) (Red)** |
| Off | Device is not ready for operation | No voltage or no CC-Link | No voltage or no CC-Link |
| LED lights up | Device is ready for operation | Device in cyclical data exchange | Bus error |

**R/W head LEDs**

| Display | Function | |
|---|---|---|
| | **RD R/W head (Green)** | **COM R/W head (Yellow)** |
| Off | Not ready | No data carrier detected |
| LED lights up | Ready | Data carrier detected (CP) |
| LED flashes | Cable break or R/W head not connected | Data carrier is being processed |

**IO-Link port LED**

The IO port is assigned an LED for displaying operating states.

| Display | Function | |
|---|---|---|
| | **IO-Link** | **Input** |
| Off | CC-Link not yet started | Signal = 0 |
| Yellow | – | Signal = 1 |
| Red | Error | SS* |
| Green | IO-Link communication active | – |
| Green, flashing | IO-Link node missing or cable break | – |

* Short-circuit at PIN 1. In this case, the LED lights up in red.

## 7 Device Function

### 7.4 Examples

**1. Reading 30 words at R/W head 1, start address 10**

Once enough data has been read to fill the input buffer of R/W head 1 while running the reading job, the data will be carried over to the input buffer. The AE bit is not set until the processor unit has finished the "Read" operation.
The reply "Job end" (AE bit) is reliably set no later than before the last data has been sent. This time point depends on the requested data amount and the time response of the controller. In the example, the use of italics for "*Set AE bit*" calls your attention to this fact.

**Control**

1. Process output buffer (note sequence):

| | |
|---|---|
| $0000_{hex}$ | Command designator $0001_{hex}$ |
| $0001_{hex}$ | Start address $000A_{hex}$ |
| $0002_{hex}$ | No. of words $001E_{hex}$ |
| RYm0 | Set AV bit |

**Identification system**

2. Process input buffer (note sequence):

| | |
|---|---|
| RXm1 | Set AA bit |
| $0000_{hex}$… $000F_{hex}$ | Enter first 16 words |
| RXm5 | Invert TO bit |
| RXm2 | *Set AE bit* |

3. Process input buffer:

| | |
|---|---|
| $0000_{hex}$… $000F_{hex}$ | Copy first 16 words |
| RYm6 | Invert TI bit |

4. Process input buffer:

| | |
|---|---|
| $0000_{hex}$… $000D_{hex}$ | Enter second 16 words |
| RXm5 | Invert TO bit |
| RXm2 | *Set AE bit* |

5. Process input buffer:

| | |
|---|---|
| $0000_{hex}$… $000D_{hex}$ | Copy second 16 words |
| RYm0 | Reset AV bit |

6.

| | |
|---|---|
| RXm1 | Reset AA bit |
| RXm2 | Reset AE bit |

# 7 Device Function

**2. Reading 30 words at R/W head 1, start address 10, problem during reading**

---

**Note**
If a problem occurs, the AF bit is set with the corresponding status number instead of the AE bit. Setting the AF bit cancels the job and declares it as finished.

---

**Control**

1. Process output buffer (note sequence):

| | |
|---|---|
| $0000_{hex}$ | Command designator $0001_{hex}$ |
| $0001_{hex}$ | Start address $000A_{hex}$ |
| $0002_{hex}$ | Number of words $001E_{hex}$ |
| RYm0 | Reset AV bit |

**Identification system**

2. Process input buffer: (note sequence): **If problem occurs immediately!**

| | |
|---|---|
| RXm1 | Set AA bit |
| $0000_{hex}$ | Enter status number |
| RXm3 | Set AF bit |

3. Process input buffer:

| | |
|---|---|
| $0000_{hex}$… $000F_{hex}$ | Copy status number |
| RYm6 | Reset AV bit |

4.

| | |
|---|---|
| RXm1 | Reset AA bit |
| RXm3 | Reset AF bit |

### 3. Reading 30 words at R/W head 1, start address 10, problem during reading

**Note**
If a problem occurs after transmission of the data has started, the AF bit is provided instead of the AE bit together with a corresponding status number. The AF status message is dominant. Which data is incorrect cannot be specified. Setting the AF bit cancels the job and declares it as finished.

**Control** / **Identification system**

**Control**

1. Process output buffer (note sequence):

| | |
|---|---|
| $0000_{hex}$ | Command designator $0001_{hex}$ |
| $0001_{hex}$ | Start address $000A_{hex}$ |
| $0002_{hex}$ | Number of words $001E_{hex}$ |
| RYm0 | Reset AV bit |

**Identification system**

2. Process input buffer (note sequence):

| | |
|---|---|
| RXm1 | Set AA bit |
| $0000_{hex}$… $000F_{hex}$ | Enter first 16 words |
| RXm5 | Invert TO bit |

**Control**

3. Process input buffer:

| | |
|---|---|
| $0000_{hex}$… $000F_{hex}$ | Copy first 16 words |
| RYm6 | Invert TI bit |

**Identification system**

4. Process input buffer:
**If a problem has occurred!**

| | |
|---|---|
| $0000_{hex}$ | Enter status number |
| RXm3 | Set AF bit |

**Control**

5. Process input buffer:

| | |
|---|---|
| $0000_{hex}$ | Copy status number |
| RYm0 | Reset AV bit |

**Identification system**

6.

| | |
|---|---|
| RXm1 | Reset AA bit |
| RXm3 | Reset AF bit |

**4. Writing 30 words at R/W head 1, start address 20**

**Control** / **Identification system**

1. Process output buffer (note sequence):

| | |
|---|---|
| $0000_{hex}$ | Command designator $0002_{hex}$ |
| $0001_{hex}$ | Start address $000A_{hex}$ |
| $0002_{hex}$ | No. of words $001E_{hex}$ |
| RYm0 | Set AV bit |

2. Process input buffer (note sequence):

| | |
|---|---|
| RXm1 | Set AA bit |
| RXm5 | Invert TO bit |

3. Process output buffer:

| | |
|---|---|
| $0000_{hex}$… $000F_{hex}$ | Enter first 16 words |
| RYm6 | Invert TI bit |

4. Process output buffer:

| | |
|---|---|
| $0000_{hex}$… $000F_{hex}$ | Copy first 16 words |
| RXm5 | Invert TO bit |

5. Process output buffer:

| | |
|---|---|
| $0000_{hex}$… $000D_{hex}$ | Enter last words |
| RYm6 | Invert TI bit |

6. Process output buffer:

| | |
|---|---|
| $0000_{hex}$… $000D_{hex}$ | Copy last words |
| RXm2 | Set AE bit |

7.

| | |
|---|---|
| RYm0 | Reset AV bit |

8.

| | |
|---|---|
| RXm1 | Reset AA bit |
| RXm2 | Reset AE bit |

# 7 Device Function

**5. Copying data from one data carrier to another**

The data from one data carrier at a read/write head (source) is copied to a data carrier in front of another read/write head (target). The data carriers have to be in front of the read/write heads (even if dynamic mode has been configured) and must have the specified address range. The command is processed in the buffer of the source head.

In the example, 17 words starting from address 10 of the data carrier are to be copied to the data carrier in front of read/write head 3 starting from address 35.

**Control**

1. Process output buffer (note sequence):

| | |
|---|---|
| $0000_{hex}$ | Command designator $0011_{hex}$ |
| $0001_{hex}$ | Source start address $000A_{hex}$ |
| $0002_{hex}$ | Target start address $0023_{hex}$ |
| $0003_{hex}$ | Number of words $0011_{hex}$ |
| $0004_{hex}$ | Target head number $0003_{hex}$ |
| RYm0 | Set AV bit |

**Identification system**

2. (note sequence)

| | |
|---|---|
| RXm1 | Set AA bit |
| RXm2 | Set AE bit |

3.

| | |
|---|---|
| RYm0 | Reset AV bit |

4.

| | |
|---|---|
| RXm1 | Reset AA bit |
| RXm2 | Reset AE bit |

**6. Writing a constant value to a data carrier**

A data carrier is to be written with 1000 words (constant value) starting from start address 80.

**Control**

1. Process output buffer (note sequence):

| | |
|---|---|
| $0000_{hex}$ | Command designator $0032_{hex}$ |
| $0001_{hex}$ | Start address $000A_{hex}$ |
| $0002_{hex}$ | No. of words $001E_{hex}$ |
| RYm0 | Set AV bit |

**Identification system**

2. (note sequence):

| | |
|---|---|
| RXm1 | Set AA bit |
| RXm5 | Invert TO bit |

3. Process output buffer:

| | |
|---|---|
| $0000_{hex}$ | Enter constant value |
| RYm6 | Invert TI bit |

4. Process output buffer:

| | |
|---|---|
| $0000_{hex}$ | Copy constant value |
| RXm2 | Set AE bit |

5.

| | |
|---|---|
| RYm0 | Reset AV bit |

6.

| | |
|---|---|
| RXm1 | Reset AA bit |
| RXm2 | Reset AE bit |

**7. Initializing a data carrier for CRC**

The sequence for CRC initialization is similar to a write command. The start address and number of words must correspond to the maximum volume of data used.
In the example the complete memory area of a data carrier (376 words) is used. 329 words on the data carrier are available as data words, since 47 words are required for the CRC.

**Control**

1. Process output buffer (note sequence):

| | |
|---|---|
| $0000_{hex}$ | Command designator $0012_{hex}$ |
| $0001_{hex}$ | Start address $0000_{hex}$ |
| $0002_{hex}$ | Number of words $0292_{hex}$ |
| RYm0 | Set AV bit |

**Identification system**

2. (note sequence):

| | |
|---|---|
| RXm1 | Set AA bit |
| RXm5 | Invert TO bit |

3. Process output buffer:

| | |
|---|---|
| $0000_{hex}$… $000F_{hex}$ | Enter first 16 words |
| RYm6 | Invert TI bit |

4. Process output buffer:

| | |
|---|---|
| $0000_{hex}$… $000F_{hex}$ | Copy first 16 words |
| RXm5 | Invert TO bit |

5. Process output buffer:

| | |
|---|---|
| $0000_{hex}$… $000F_{hex}$ | Enter second 16 words |
| RYm6 | Invert TI bit |

6. Process output buffer:

| | |
|---|---|
| $0000_{hex}$… $000F_{hex}$ | Copy second 16 words |
| RXm5 | Invert TO bit |

42. Process output buffer:

| | |
|---|---|
| $0000_{hex}$… $0002_{hex}$ | Enter last words |
| RYm6 | Invert TI bit |

43. Process output buffer:

| | |
|---|---|
| $0000_{hex}$… $0002_{hex}$ | Copy last words |
| RXm2 | Set AE bit |

44.

| | |
|---|---|
| RYm0 | Reset AV bit |

45.

| | |
|---|---|
| RXm1 | Reset AA bit |
| RXm2 | Reset AE bit |

**8. Creating a basic state for a R/W head or switching off a R/W head**

The read/write heads for the identification system can be put into a basic state independently of each other and the respective read/write head can be shut off.

| Control | | Identification system | |
|---|---|---|---|
| 1. | | 2. Go to basic state. Process input buffer: | |
| RYm2 | Set GR bit | RXm7 | Reset BB bit |
| | | ⇒ R/W head is shut off. | |
| 3. | | 4. | |
| RYm2 | Reset GR bit | RXm7 | Set BB bit |

**9. Switching off a read/write head antenna**

During normal operation, all read/write head antennas are switched on. The antenna of a respective R/W head can be switched off by setting the KA bit.

**Control**

1.

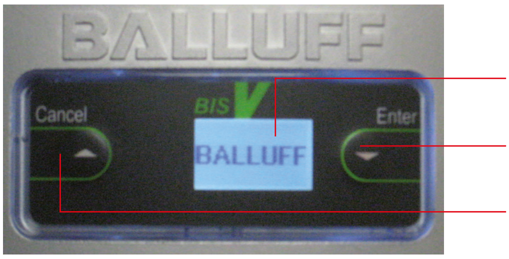
| | |
|---|---|
| RYm5 | Set KA bit |

The R/W head's antenna is switched back on by resetting the KA bit.

# 7 Device Function

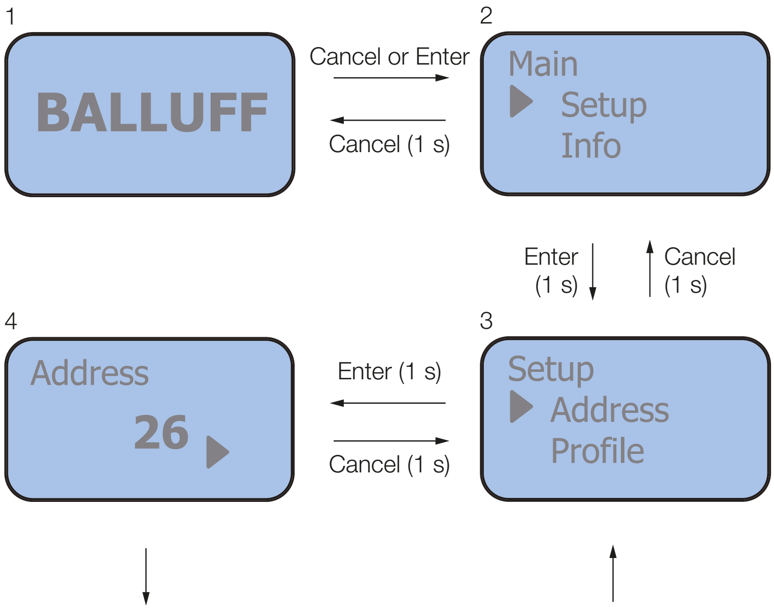
## 7.5 Display

The display provides functions for starting up the BIS V. This can be used to configure the station address, profile and baud rate. Tag data and the version information can be read out for diagnostic purposes. It is controlled using a 2-button controller.
You can navigate within a menu level by holding the Enter/Down and Cancel/Up buttons for a short time. You can switch between menu levels or confirm or cancel an action by pressing the buttons longer.

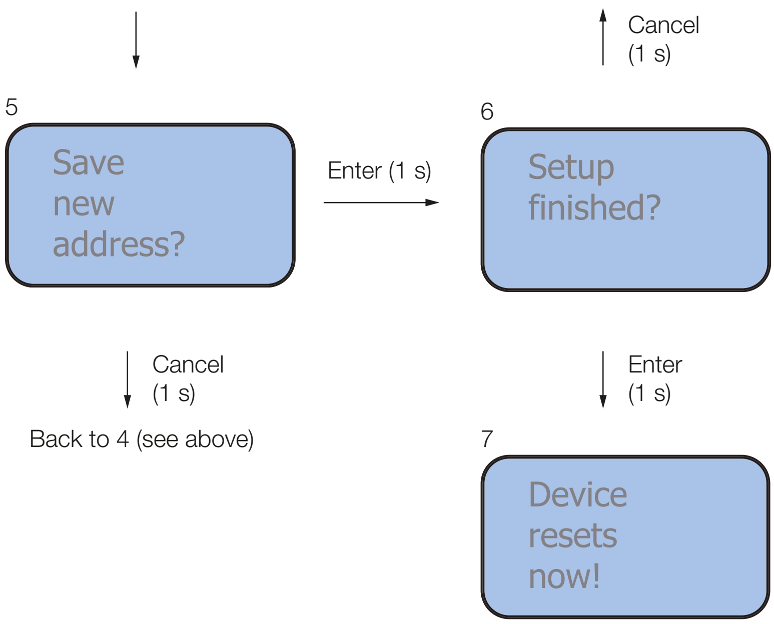


# 7 Device Function

**Setting the station address**



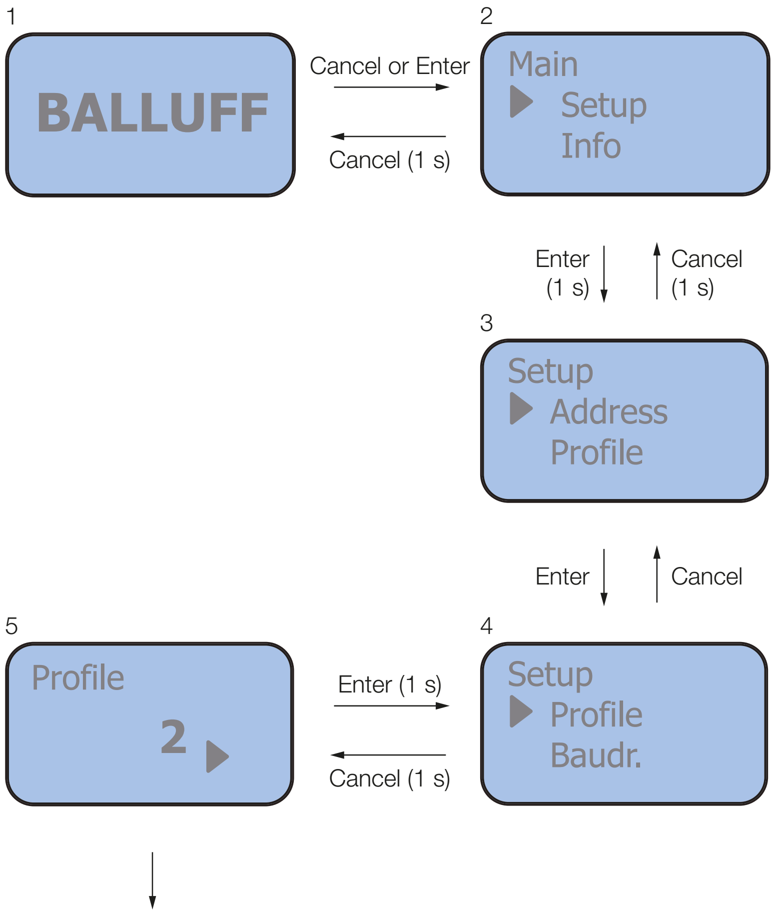

- You switch through the positions (1st, 2nd, 3rd) using Enter/Down.
  ⇒ The current position starts flashing.
- You can cause the number at the currently selected position to go up using Cancel/Up.
- Pressing Cancel (1 s) cancels the configuration.
  ⇒ Back to 4; the currently configured station address is shown.
- Press Enter (1 s) to confirm the configured address.



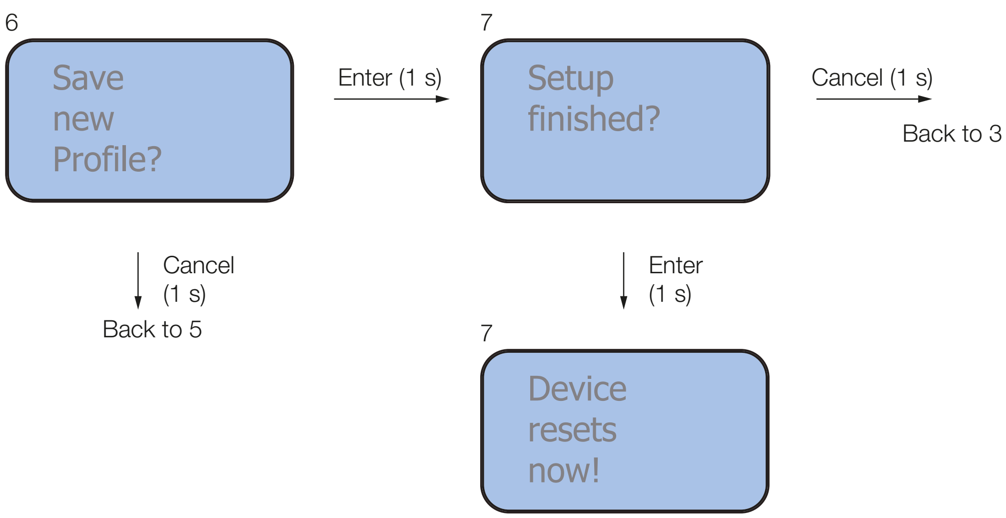

⇒ A device reset is carried out (after restart: Back to 1)

# 7 Device Function

**Setting the profile**

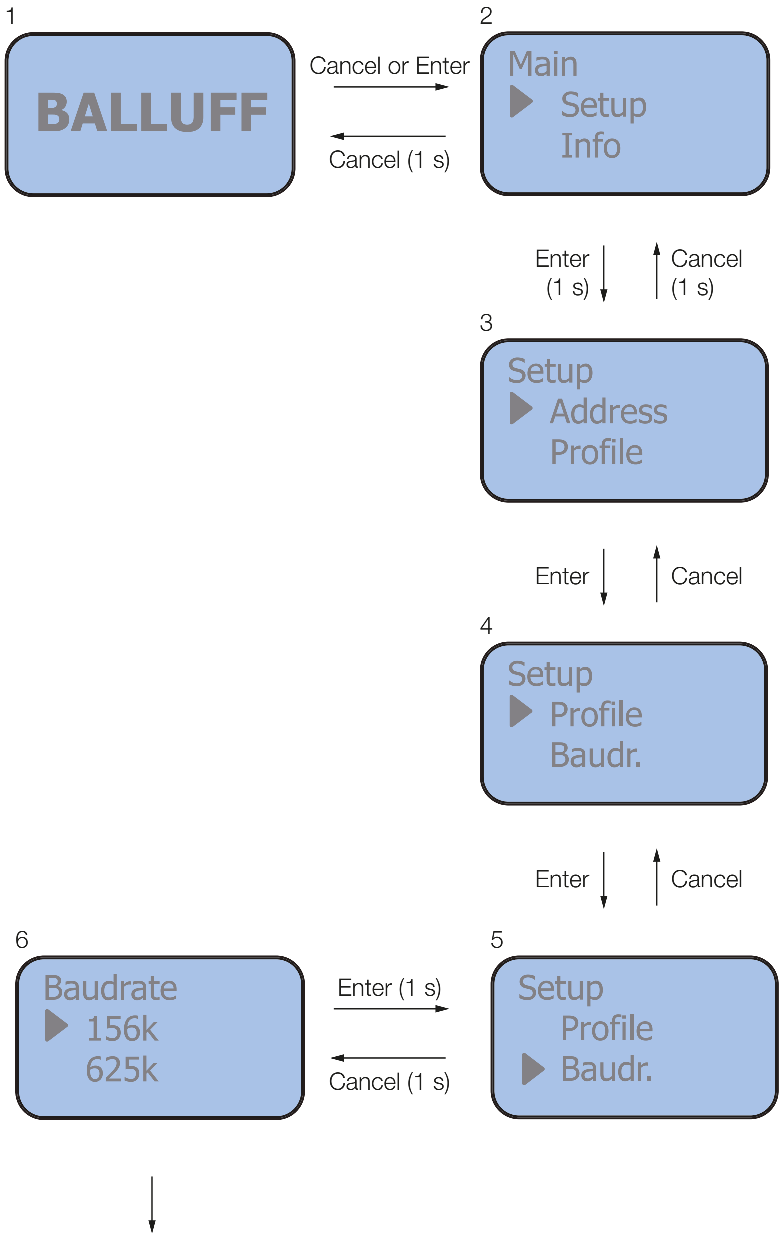



- Pressing Enter/Down causes the number to start flashing.
- Pressing Cancel/Up causes the number to go up.
- Pressing Cancel (1 s) cancels the configuration.
  ⇒ Back to 5
- Press Enter (1 s) to confirm the configured profile.



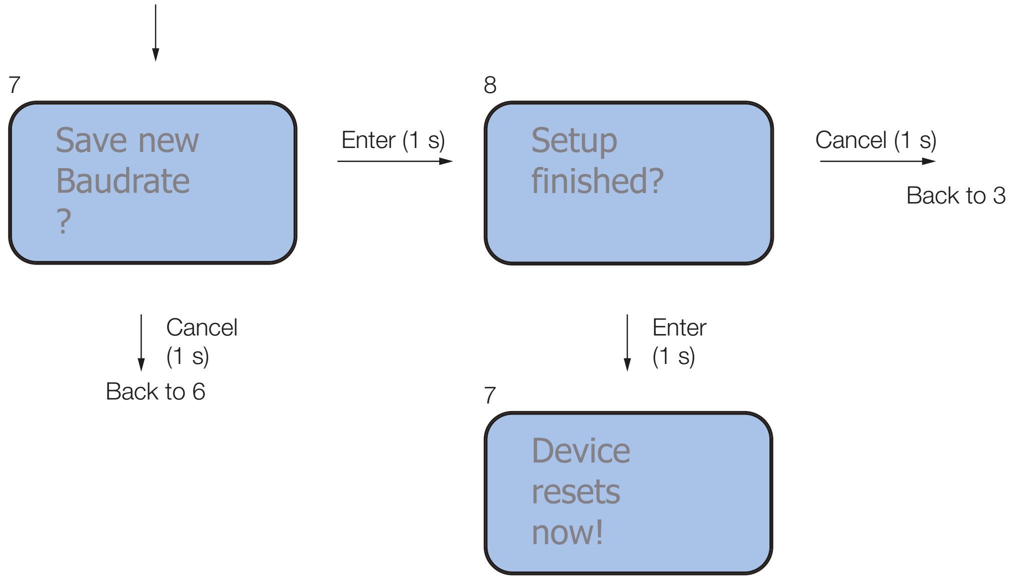
# 7 Device Function

**Setting the baud rate**



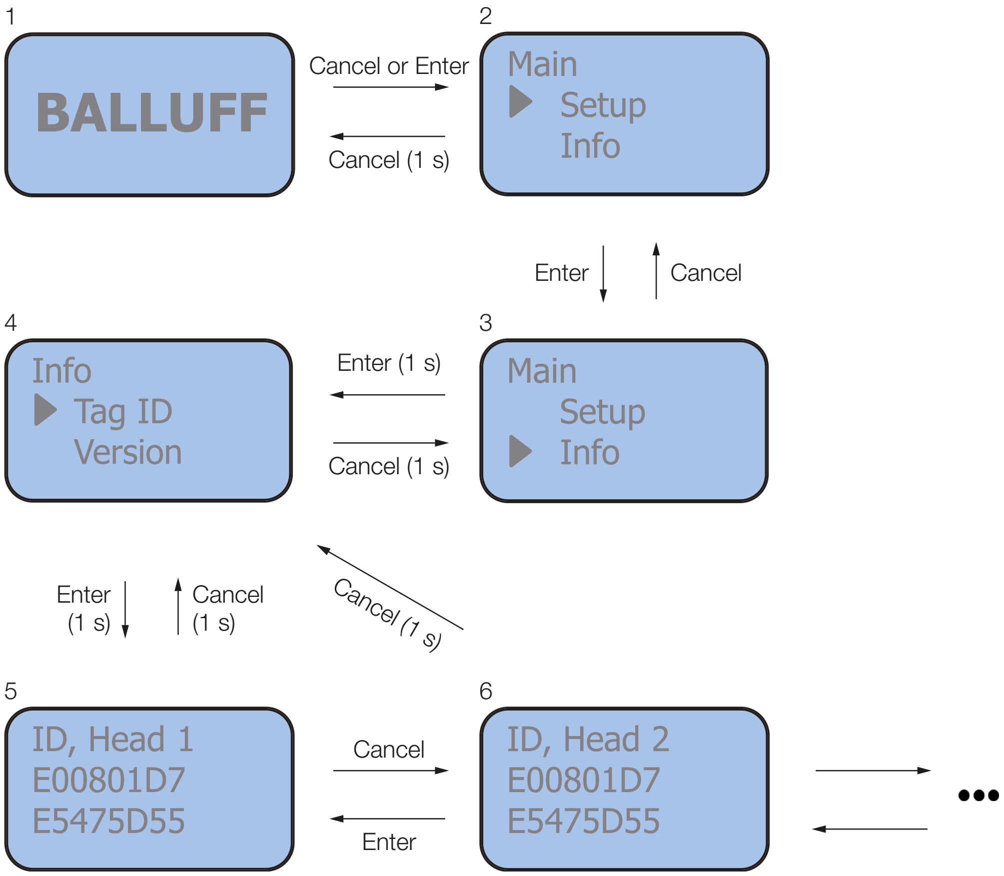

- ► Press Enter/Down to select the desired baud rate
- ► Pressing Cancel (1 s) cancels the configuration.
  ⇒ Back to 4; the current baud rate is shown
- ► Press Enter (1 s) to apply the value

# 7 Device Function



# 7 Device Function

**Displaying tag data**

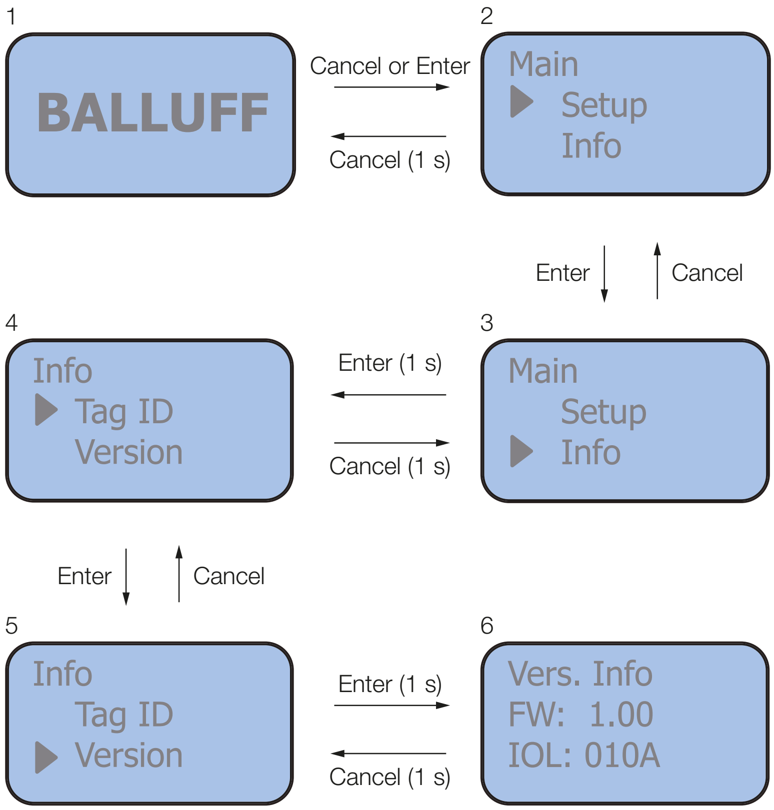



Display of the ID of the data carrier in front of read head 1

Display of the ID of the data carrier in front of read head 2

When selecting Head_IDs 1…4 (5, 6, etc.), you can hold down Cancel for 1 s to return to 4.

# 7 Device Function




Two versions are displayed:
- The firmware version of the device (here 1.00)
- The software version of the IO-Link firmware (010A)

## Appendix

**Type code**

**BIS V – 6 1 11 – 073 – C_03**

Balluff Identification System

Series V (V = variable)

System component

6 = Processor unit

Generation (design/material)

1 = Generation 1, 2011 housing design, metal

Interface

02 = PROFIBUS DP
06 = Ethernet/IP
08 = PROFINET
10 = EtherCAT
11 = CC-Link

Software type

073 = CC-Link

Connection system

C003 = *Power supply:* 5-pin flanged male connector with 7/8" external thread, A-coded
*IO-Link master and USB:* 5-pin flanged female connector with M12 internal thread, A-coded
*CC-Link input:* 5-pin flanged male connector with M12 external thread, A-coded
*CC-Link output:* 5-pin flanged female connector with M12 internal thread, A-coded
*4 heads VL/VM and future systems:* 5-pin flanged female connector with M12 internal thread, A-coded

C103 = As for C003, also supports BIS C read/write heads (adapter required)

**Accessories (optional, not included)**

**Note**
You can find more accessories for the BIS V-6111-… in the Balluff BIS catalog and at www.balluff.com.

## Appendix

### ASCII table

| Decimal | Hex | Control code | ASCII |
|---|---|---|---|
| 0 | 00 | Ctrl @ | NUL |
| 1 | 01 | Ctrl A | SOH |
| 2 | 02 | Ctrl B | STX |
| 3 | 03 | Ctrl C | ETX |
| 4 | 04 | Ctrl D | EOT |
| 5 | 05 | Ctrl E | ENQ |
| 6 | 06 | Ctrl F | ACK |
| 7 | 07 | Ctrl G | BEL |
| 8 | 08 | Ctrl H | BS |
| 9 | 09 | Ctrl I | HT |
| 10 | 0A | Ctrl J | LF |
| 11 | 0B | Ctrl K | VT |
| 12 | 0C | Ctrl L | FF |
| 13 | 0D | Ctrl M | CR |
| 14 | 0E | Ctrl N | SO |
| 15 | 0F | Ctrl O | SI |
| 16 | 10 | Ctrl P | DLE |
| 17 | 11 | Ctrl Q | DC1 |
| 18 | 12 | Ctrl R | DC2 |
| 19 | 13 | Ctrl S | DC3 |
| 20 | 14 | Ctrl T | DC4 |
| 21 | 15 | Ctrl U | NAK |
| 22 | 16 | Ctrl V | SYN |
| 23 | 17 | Ctrl W | ETB |
| 24 | 18 | Ctrl X | CAN |
| 25 | 19 | Ctrl Y | EM |
| 26 | 1A | Ctrl Z | SUB |
| 27 | 1B | Ctrl [ | ESC |
| 28 | 1C | Ctrl \ | FS |
| 29 | 1D | Ctrl ] | GS |
| 30 | 1E | Ctrl ^ | RS |
| 31 | 1F | Ctrl _ | US |
| 32 | 20 | | SP |
| 33 | 21 | | ! |
| 34 | 22 | | „ |
| 35 | 23 | | # |
| 36 | 24 | | $ |
| 37 | 25 | | % |
| 38 | 26 | | & |
| 39 | 27 | | ‘ |
| 40 | 28 | | ( |
| 41 | 29 | | ) |
| 42 | 2A | | * |

| Decimal | Hex | ASCII |
|---|---|---|
| 43 | 2B | + |
| 44 | 2C | , |
| 45 | 2D | - |
| 46 | 2E | . |
| 47 | 2F | / |
| 48 | 30 | 0 |
| 49 | 31 | 1 |
| 50 | 32 | 2 |
| 51 | 33 | 3 |
| 52 | 34 | 4 |
| 53 | 35 | 5 |
| 54 | 36 | 6 |
| 55 | 37 | 7 |
| 56 | 38 | 8 |
| 57 | 39 | 9 |
| 58 | 3A | : |
| 59 | 3B | ; |
| 60 | 3C | < |
| 61 | 3D | = |
| 62 | 3E | > |
| 63 | 3F | ? |
| 64 | 40 | @ |
| 65 | 41 | A |
| 66 | 42 | B |
| 67 | 43 | C |
| 68 | 44 | D |
| 69 | 45 | E |
| 70 | 46 | F |
| 71 | 47 | G |
| 72 | 48 | H |
| 73 | 49 | I |
| 74 | 4A | J |
| 75 | 4B | K |
| 76 | 4C | L |
| 77 | 4D | M |
| 78 | 4E | N |
| 79 | 4F | O |
| 80 | 50 | P |
| 81 | 51 | Q |
| 82 | 52 | R |
| 83 | 53 | S |
| 84 | 54 | T |
| 85 | 55 | U |

| Decimal | Hex | ASCII |
|---|---|---|
| 86 | 56 | V |
| 87 | 57 | W |
| 88 | 58 | X |
| 89 | 59 | Y |
| 90 | 5A | Z |
| 91 | 5B | [ |
| 92 | 5C | \ |
| 93 | 5D | [ |
| 94 | 5E | ^ |
| 95 | 5F | _ |
| 96 | 60 | ` |
| 97 | 61 | a |
| 98 | 62 | b |
| 99 | 63 | c |
| 100 | 64 | d |
| 101 | 65 | e |
| 102 | 66 | f |
| 103 | 67 | g |
| 104 | 68 | h |
| 105 | 69 | i |
| 106 | 6A | j |
| 107 | 6B | k |
| 108 | 6C | l |
| 109 | 6D | m |
| 110 | 6E | n |
| 111 | 6F | o |
| 112 | 70 | p |
| 113 | 71 | q |
| 114 | 72 | r |
| 115 | 73 | s |
| 116 | 74 | t |
| 117 | 75 | u |
| 118 | 76 | v |
| 119 | 77 | w |
| 120 | 78 | x |
| 121 | 79 | y |
| 122 | 7A | z |
| 123 | 7B | { |
| 124 | 7C | \| |
| 125 | 7D | } |
| 126 | 7E | ~ |
| 127 | 7F | DEL |

## Index

www.balluff.com


**Headquarters**

**Germany**
Balluff GmbH
Schurwaldstrasse 9
73765 Neuhausen a.d.F.
Phone + 49 7158 173-0
Fax +49 7158 5010
balluff@balluff.de

**Global Service Center**

**Germany**
Balluff GmbH
Schurwaldstrasse 9
73765 Neuhausen a.d.F.
Phone +49 7158 173-370
Fax +49 7158 173-691
service@balluff.de

**US Service Center**

**USA**
Balluff Inc.
8125 Holton Drive
Florence, KY 41042
Phone (859) 727-2200
Toll-free 1-800-543-8390
Fax (859) 727-4823
technicalsupport@balluff.com

**CN Service Center**

**China**
Balluff (Shanghai) trading Co., ltd.
Room 1006, Pujian Rd. 145.
Shanghai, 200127, P.R. China
Phone +86 (21) 5089 9970
Fax +86 (21) 5089 9975
service@balluff.com.cn


No. 895851-726 EN · 04.127612 · L18; Subject to modification. Replaces K17.