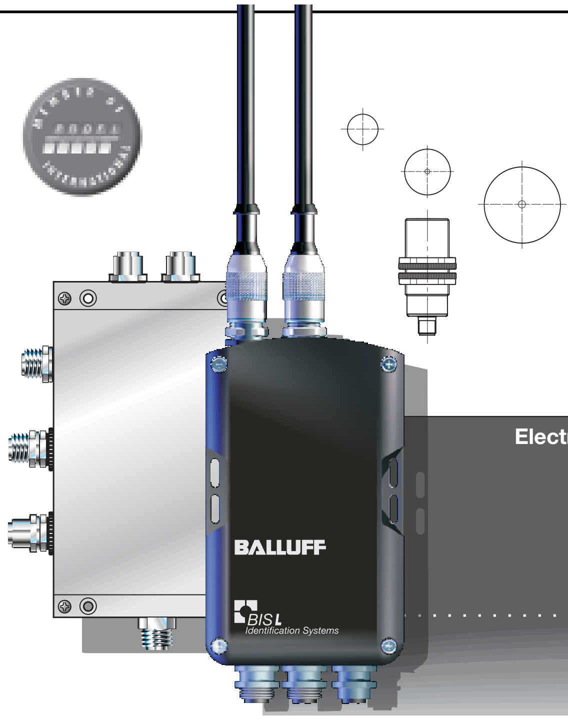
BALLUFF



Manual

Electronic Identification Systems BIS
Processor BIS L-60_2
Profibus DP

Deutsch – bitte wenden!

No. 828 132 D/E • Edition 1007
Subject to modification.
Replaces edition 1002.

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

■ **www.balluff.com**

Contents

Safety Considerations 4
Introduction, BIS L Identification Systems 5/6
BIS L-60_2 Processor, Basic knowledge for application 7/8
BUS interface PROFIBUS-DP 9-11
Function Description: Communication with the processor 12
Input and Output Buffers 13/14
Output Buffer, configuration and explanation 15-18
Input Buffer, configuration and explanation 19-22
Parametering the BIS C-60_2 processor 23/24
Parametering, Parametering Bytes 25-27
Data-carrier models BIS L-10_-01/L 28
Data-carrier models BIS L-20_-03/L 29
Processing data carriers 30-36
Examples for protocol sequence 37-50
Read/Write Times 51
LED Display 52
BIS L-6002 **BIS L-6022**
Mounting the Processor 53 63
Opening the Processor / Interface Information 54 64
Interface Information / Wiring Diagrams 55-57 65-67
Changing the EEPROM 58 68
Technical Data .... 59/60 .... 69/70
Ordering Information .... 61/62 .... 71/72
Symbols / Abbreviations 73
Appendix, ASCII Table 74

Safety Considerations

Approved Operation

Series BIS L-60_2 processors along with the other BIS L system components comprise an identification system and may only be used for this purpose in an industrial environment in conformity with Class A of the EMC Law.

Installation and Operation

Installation and operation should be carried out by trained personnel only. Unauthorized work and improper use will void the warranty and liability.

When installing the processor, follow the chapters containing the wiring diagrams closely. Special care is required when connecting the processor to external controllers, in particular with respect to selection and polarity of the signals and power supply.

Only approved power supplies may be used for powering the processor. See chapter 'Technical Data' for details.

Use and Checking

Prevailing safety regulations must be adhered to when using the identification system. In particular, steps must be taken to ensure that a failure of or defect in the identification system does not result in hazards to persons or equipment.

This includes maintaining the specified ambient conditions and regular testing for functionality of the identification system including all its associated components.

Fault Conditions

Should there ever be indications that the identification system is not working properly, it should be taken out of commission and secured from unauthorized use.

Scope

This manual applies to processors in the series BIS L-6002-019-050-03-ST11 and
BIS L-6022-019-050-03-ST14.

Introduction
BIS L Identification Systems

This manual is designed to assist the user in setting up the control program and installing and starting up the components of the BIS L Identification System, and to assure rapid, trouble-free operation.

Principles

The BIS L Identification Systems belongs in the category of

non-contact systems for reading and writing.

This dual function permits applications for not only transporting information in fixed-programmed data carriers, but also for gathering and passing along up-to-date information as well. The BIS L identification system allows the use of read-only data carriers.

☞ If 2 read/write heads are connected to a BIS L-60_2 processor, both heads can be operated independently of each other. This means for example that you can read a data carrier from one head while writing to another data carrier at the other head.

Applications

Some of the notable areas of application include

– **for controlling material flow in production processes**
 (e.g. in model-specific processes)**,**
 for workpiece conveying in transfer lines,
 in data gathering for quality assurance,
 for gathering safety-related data,

– **in storage systems for monitoring inventory movement;**

– **in transporting and conveying systems.**

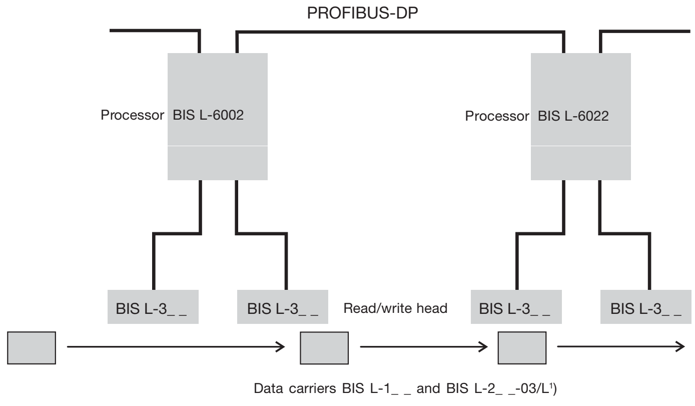
Introduction
BIS L Identification Systems

System Components

The main components of the BIS L Identification Systems are:

– **Processor,**
– **Read/Write Heads and**
– **Data carriers**

Configuration with BIS L-6002 processor



Schematic representation of an Identification System (example)

[1]) Mixed operation of type BIS L-1_ _ and BIS L-2_ _-03/L together is possible

BIS L-60_2 Processor
Basic knowledge for application

Selecting System Components

The **BIS L-6002** processor has a plastic housing.

The **BIS L-6022** processor has a metal housing.

Connection is made through round connectors. Two read/write heads can be cable connected.

Series BIS L-60_2 processors have in addition a digital input. The input has various functions depending on the configuration (see Parametering).

The read/write distances depend on which data carriers are used. Additional information on the read/write heads in series BIS L-3_ _ including all the possible data carrier/read-write head combinations can be found in the manuals for the respective read/write heads.

The system components are electrically supplied by the processor. The data carrier represents a free-standing unit and needs no line-carried power. It receives its energy from the read/write head. The latter constantly sends out a carrier signal which supplies the code head as soon as the required distance between the two is reached. The read/write operation takes place during this phase. Reading and writing may be dynamic or static.

BIS L-60_2 Processor
Basic knowledge for application

Control Function

The processor writes data from the host system to the data carrier or reads data from the data carrier through the read/write head and prepares it for the host system. Host systems may include:

– **a host computer (e.g. industrial PC) or**
– **a programmable logic controller (PLC)**

Data checking with CRC_16

For applications requiring high security against bad data, CRC_16 checking can be used. Here a check code is written to the data carrier which allows the data to be checked for integrity at any time or location.

Advantages to CRC_16: Very high data integrity, even during the non-active phase (data carrier outside the active zone of the r/w head)

Disadvantages to CRC_16: Longer read/write times, some user data space is taken up on the data carrier.

Use of CRC_16 can be parameterized by the user. (see 🗋 25)

BUS interface PROFIBUS-DP

PROFIBUS-DP

Communication between the BIS L-60_2 processor and the host system is via PROFIBUS-DP. The PROFIBUS-DP system consists of the components:

– the bus master and
– the bus modules/slaves (here the BIS L-60_2 processor).

☞

Important hints for use with PLC:

In some control systems the PROFIBUS-DP data area is not synchronously transmitted with the updating of the input/output content. If more than 2 bytes of data are sent, a mechanism must be used which guarantees that the data in the PLC and the data in the BIS L are always identical!

1st alternative: Synchronous data transmission as a setting on the Master
In this method the bus Master ensures that all the data necessary for the respective Slave are always sent contiguously. There is usually a special software function in the PLC which likewise controls access between the PLC and bus Master so that data are always sent contiguously.

2nd alternative: Set 2nd bit header
Data exchange between PLC and BIS is controlled by the so-called bit header. This is always the first byte of the respective read/write head in the data buffer. This bit header exists both in the input range (data from BIS to the PLC) and in the output range (data from the PLC to the BIS). IIf this bit header is also sent as the last byte, a comparison of these two bytes can be used to guarantee the consistency of the transmitted data.

In this method the PLC cycle is unaffected nor is the bus access time changed. All that is required is that a byte in the data buffer be used for the 2nd bit header instead of for user data.

This 2nd alternative is the Balluff recommended setting (factory default).

BUS interface PROFIBUS-DP

Unit's Master Data

For the correct parametering of the bus master as per type, CD ROM, containing the unit’s master data in the form of a GSD file is included with the BIS L-60_2 processor.

Station Address

The Processor BIS L-60_2 is delivered with the station address 126. This has to be set individually before using in a bus system. See information on 🗋 11.

Input/Output Buffer

An input buffer and an output buffer are used for the data exchange with the control system. The size of these buffers has to be configured via the master.

☞ The possible settings are entered in the GSD file (and Type file). A minimum of 4 and a maximum of 128 bytes can be accommodated. However, it must be an even number.

Parametering Bytes
User-Parameter Bytes

Besides, in the case of the BIS L-60_2 processor, there are 6 further bytes (User-Parameter Bytes) which have to be set while parametering. The significance of the 6 bytes for parametering is described starting from 🗋 25.

☞ The preset is stored in the GSD file.

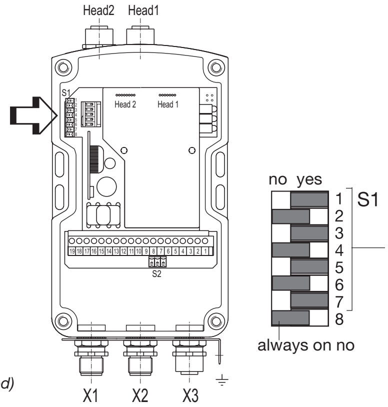
BUS interface PROFIBUS-DP

Station Address setting

The station address under which the unit is accessed on the bus can be assigned through the slide switch S1. Each address shall be assigned only once.

The slide switch S1 is binary coded. The setting of the station address is carried out according to the scheme shown in the table. Switch position: no = left, yes = right.

The address 85 is set in the following figure.



Slide switch S1
(with cover removed)

Station Address	Slide switch S1						
	7	6	5	4	3	2	1
	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	not allowed						
1	no	no	no	no	no	no	yes
2	no	no	no	no	no	yes	no
3	no	no	no	no	no	yes	yes
4	no	no	no	no	yes	no	no
5	no	no	no	no	yes	no	yes
...							
85	yes	no	yes	no	yes	no	yes
...							
123	yes	yes	yes	yes	no	yes	yes
124	yes	yes	yes	yes	yes	no	no
125	yes	yes	yes	yes	yes	no	yes
126	yes	yes	yes	yes	yes	yes	no
127	not allowed						

To open the cover of the processor, see ⎘ 54 for BIS L-6002 or ⎘ 64 for BIS L-6022.

Function Description
Communication with the processor

Basic Procedure

Communication between the host system and the processor takes place using a fixed protocol sequence. Data integrity from the control to the processor and vice-versa is indicated by a control bit. This bit is used to implement a handshake between the control and the processor.

Following is a simplified representation of the sequence of a job sent from the control to the processor:

1. The control sends a command designator to the processor together with the associated command parameters and sets a bit (AV bit). This bit indicates to the processor that the transmitted data are valid and that the job is now beginning.
2. The processor takes the job and sets a bit (AA bit), which indicates this to the control.
3. If an additional exchange of data between the control and the processor is required to carry out the job, each uses a bit (TI bit and TO bit) to indicate that the control / processor is now ready for additional data exchange or has accepted the received data.
4. Once the processor has carried out the job correctly, it sets a bit (AE bit).
5. Once the control has accepted all the important data, it indicates this to the processor by resetting the bit that was set at the beginning (AV bit).
6. The processor now in turn sets all the control bits that were set during the sequence (AA bit, AE bit) and is ready for the next job.

Please see also ⎘⎘ 30...36 and the examples on ⎘⎘ 37...50.

Function Description
Input and Output Buffers

Input and Output Buffers

In order to transmit commands and data between the BIS L-60_2 and the host system, the latter must prepare two fields. These two fields are:

- **the output buffer**
  for the control commands which are sent **to** the BIS Identification System and
  for the data to be written.
- **the input buffer**
  for the data to be read and
  for the designators and error codes which come **from** the BIS Identification System.

The possible setting values are stored in the GSD file.

The buffer size can be selected between 4 and 128 bytes in steps of 2 bytes. This must be given by the master during parametering. The total buffer size is divided into 2 ranges:

Buffer range 1 for Read/Write Head 1; size is specified in paramter byte 6.
Buffer range 2 for Read/Write Head 2; size = total buffer size – buffer size of Read/Write Head 1.
See 🗎 14 for example.

☞

If a buffer size of less than 8 bytes is set for a read/write head, a read/write request can be carried out without specifying the start address and the number of bytes. Automatic reading for Codetag-Present (see 🗎 31) remains active. This permits fast reading of small data quantities without placing an unnecessary load on the bus.

Buffer size – 1 = number of bytes read without double bit header;
Buffer size – 2 = number of bytes read with double bit header.

Please note the basic procedure on 🗐 12 and 30...36 and the examples on pages 🗐 37...50.

Function Description
Input and Output Buffers

Input and Output Buffers (continued)

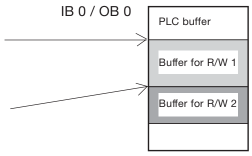
Example: The 82 bytes for the total buffer need to be distributed. An input/output buffer of 46 bytes is assigned to Read/Write Head 1. This results in an input/output buffer of 36 bytes for Read/Write Head 2.

Procedure: The buffer size for Read/Write Head 1 is set to 46 bytes. This means using the parameter byte 6 to enter Hex value 2E (corresponds to 46 decimal), which corresponds to binary 0**010111**0.

PLC Organisation: The buffer range starts at input byte IB 32 and output byte OB 32.

Result:

Read/Write Head 1:	Subaddress 00	IB 32 and OB 32
(R/W 1)	Input buffer	IB 32 to IB 77
	Output buffer	OB 32 to OB 77
Read/Write Head 2:	Subaddress 00	IB 78 and OB 78
(R/W 2)	Input buffer	IB 78 to IB 113
	Output buffer	OB 78 to OB 113



☞

Note that these buffers can be in two different sequences depending on the type of control.

The following description is based on sequence 1!

Sequence 1	Sequence 2
Subaddress 00	Subaddress 01
01	00
02	03
03	02
04	05
05	04
06	07
07	06

Please note the basic procedure on 🗐 12 and 30...36 and the examples on pages 🗐 37...50.

Function Description
Output buffer, configuration and explanation

Configuration of the Output Buffer for One (1) Read/Write Head

The last two bytes can be parameterized as the 2nd bit header (default).

Bit No. / Subaddress	7	6	5	4	3	2	1	0	
00Hex = Bit Header		TI	KA			GR		AV	Bit Name
01Hex	Command Designator					or	Data		
02Hex	Start Address (Low Byte) or Program No.					or	Data		
03Hex	Start Address (High Byte)					or	Data		
04Hex	No. of Bytes (Low Byte)					or	Data		
05Hex	No. of Bytes (High Byte)					or	Data		
06Hex	Data								
...	Data								
Last Byte	2nd Bit Header (as above)					or	Data		

Description of Output Buffer

Sub-address	Bit Name	Meaning	Function Description
00Hex Bit Header	TI	Toggle-Bit In	Shows during a read action that the controller is ready for additional data.
	KA	Head function	Turn read/write head on/off as needed. Active = 0 Read/write head is on. Inactive = 1 Read/write head is off.
	GR	Ground state	Causes the BIS system to go to the ground state for the respective read/write head. Any pending command is cancelled.
	AV	Command	Signals the identification system that a command for the respective read/write head is present.

Please note the basic procedure on ⎘ 12 and 30...36 and the examples on pages ⎘ 37...50.

(continued next ⎘)

Function Description
Output buffer, configuration and explanation

Description of Output Buffer (continued)

Sub-address	Meaning	Function Description
01Hex	Command designator	
	00Hex	No command present
	01Hex	Read data carrier
	02Hex	Write to data carrier
	06HEX	Store program in the EEPROM for the Mixed Data Access function
	07HEX	Store the start address for the Auto-Read function in the EEPROM
	12HEX	Initialize the CRC16 data check
	21HEX	Read for Mixed Data Access function (corresponding to the program stored in the EEPROM)
	22HEX	Write for Mixed Data Access function (corresponding to the program stored in the EEPROM)
or:	Data	for writing to the data carrier
or:	Program data	for writing to the EEPROM.

(continued next ⎘)

Please note the basic procedure on ⎘ 12 and 30...36 and the examples on pages ⎘ 37...50.

Function Description
Output buffer, configuration and explanation

Description of Output Buffer
(continued)

Sub-address	Meaning	Function Description
02$_{Hex}$	Start address (Low Byte)	Address at which reading from or writing to the data carrier begins. (The Low Byte includes the address range from 0 to 255).
or:	Start address (Low Byte)	Address for the Auto-Read function, starting at which the code tag is to be read. The value is stored in the EEPROM. (The Low Byte covers the address range from 0 to 255).
or:	Program No.	Number of the program to be stored in the EEPROM in conjunction with command ID 06$_{Hex}$ for Mixed Data Access function (values between 01$_{Hex}$ and 0A$_{Hex}$ are allowed!).
or:	Program No.	Number of the program stored in the EEPROM for read or write operations in conjunction with command ID 22$_{Hex}$ or 22$_{Hex}$ for the Mixed Data Access function.
or:	Data	for writing to the data carrier
or:	Program data	for writing to the EEPROM.
03$_{Hex}$	Start address (High Byte)	Address for reading from or writing to the data carrier . (The High Byte includes the address range from 256 to 1999).
or:	Start address (High Byte)	Address for the Auto-Read function, starting at which the code tag is to be read. The value is stored in the EEPROM. (The High Byte includes the address range from 256 to 1999).
or:	Data	for writing to the data carrier
or:	Program data	for writing to the EEPROM.

(continued next 🕮)

Please note the basic procedure on 🕮 12 and 30...36 and the examples on pages 🕮 37...50.

Function Description
Output buffer, configuration and explanation

Description of Output Buffer
(continued)

Sub-address	Meaning	Function Description
04$_{Hex}$	No. of bytes (Low Byte)	Number of bytes to read or write beginning with the start address (the Low Byte includes from 1 to 255 bytes).
or:	Data	for writing to the data carrier
or:	Program data	for writing to the EEPROM.
05$_{Hex}$	No. of bytes (High Byte)	Number of bytes to read or write beginning with the start address (the High Byte includes the address range from 256 to 1999).
or:	Data	for writing to the data carrier
or:	Program data	for writing to the EEPROM.
06$_{Hex}$	Data	for writing to the data carrier
or:	Program data	for writing to the EEPROM.
...	Data	for writing to the data carrier
or:	Program data	for writing to the EEPROM.
Last byte		
	2nd Bit header	The data are valid if the 1st and 2nd bit header are identical.
or:	Data	for writing to the Data carrier
or:	Program data	for writing to the EEPROM.

Please note the basic procedure on 🕮 12 and 30...36 and the examples on pages 🕮 37...50.

Function Description
Input buffer, configuration and explanation

Configuration of the input buffer for one (1) read/write head

The last byte can be arranged as a 2nd bit header through parametering (default).

Bit No. / Subaddress	7	6	5	4	3	2	1	0	
00_{Hex} = Bit Header	BB	HF	TO	IN	AF	AE	AA	CP	Bit Name
01_{Hex}	Error Code or Data								
02_{Hex}	Data								
03_{Hex}	Data								
04_{Hex}	Data								
05_{Hex}	Data								
06_{Hex}	Data								
...	Data								
Last byte	2nd Bit Header (as above) or Data								

Description of Input Buffer

Sub-address	Bit Name	Meaning	Function Description
00_{Hex}	BB	Ready	The BIS Identification System is in the Ready state.
Bit Header	HF	Head Error	Cable break from read/write head or no read/write head connected.
	TO	Toggle-Bit Out	for read: BIS has new/additional data ready. for write: BIS is ready to accept new/additional data.

Please note the basic procedure on 📖 12 and 30...36 and the examples on pages 📖 37...50.

(continued on next 📄)

Function Description
Input buffer, configuration and explanation

Description of Input Buffer (continued)

Sub-address	Bit Name	Meaning	Function Description
00_{Hex}	(continued)		
Bit Header	IN	Input	If the parameter "Input IN" is 1, this bit indicates the state of the Input.
	AF	Command Error	The command was incorrectly processed or aborted.
	AE	Command end	The command was finished without error.
	AA	Command start	The command was recognized and started.
	CP	Codetag Present	Data carrier present within the active zone of the read/write head.

In addition to the CP bit, the output signal **CT present** is available. This allows you to process the presence of a data carrier directly as a hardware signal.

Sub-address	Meaning	Function Description
01_{Hex}	Error code	Error number is entered if command was incorrectly processed or aborted. Only valid with AF bit!
	00_{Hex}	No error.
	01_{Hex}	Reading or writing not possible because no data carrier is present in the active zone of a read/write head.
	02_{Hex}	Read error.
	03_{Hex}	Data carrier was removed from the active zone of the read/write head while it was being read.
	04_{Hex}	Write error.

(continued on next 📄)

Please note the basic procedure on 📖 12 and 30...36 and the examples on pages 📖 37...50.

Function Description
Input buffer, configuration and explanation

Description of Input Buffer (continued)

Sub-address	Meaning		Function Description
01$_{\text{Hex}}$	Error code		(continued)
	05$_{\text{Hex}}$		Data carrier was removed from the active zone of the read/write head while it was being written.
	07$_{\text{Hex}}$		AV bit is set but the command designator is missing or invalid.
		or:	Number of bytes is 00$_{\text{Hex}}$.
	09$_{\text{Hex}}$		Cable break to select read/write head, or head not connected.
	0C$_{\text{Hex}}$		The EEPROM cannot be read/programmed.
	0D$_{\text{Hex}}$		Communication with the read/write head.
	0E$_{\text{Hex}}$		The CRC of the read data does not coincide with the CRC of the data carrier.
	0F$_{\text{Hex}}$		Contents of the 1st and 2nd bit header (1st and last bytes) of the output buffers are not identical (2nd bit header must be served).
	20$_{\text{Hex}}$		Adressing of the read/write job is outside the memory range of the data carrier.
	21$_{\text{Hex}}$		Function invoked which is not possible for the data carrier currently in front of the read/write head.
or:	Data		Data which was read from the data carrier.

(continued next 📄)

Please note the basic procedure on 📄 12 and 30...36 and the examples on pages 📄 37...50.

Function Description
Input buffer, configuration and explanation

Description of Input Buffer (continued)

Sub-address	Meaning	Function Description
02$_{\text{Hex}}$	Data	Data which was read from the data carrier.
...	Data	Data which was read from the data carrier.
Last byte		
	2nd Bit header	The data are valid if the 1st and 2nd bit headers are in agreement.
or:	Data	Data which was read from the data carrier.

Please note the basic procedure on 📄 12 and 30...36 and the examples on pages 📄 37...50.

Function Description
Parametering the BIS L-60_2 processor

Parameters, Overview

There are 6 user parameter bytes stored on the Profibus master that can be used to activate and deactivate various functions. Setting is done directly by linking a device to the Profibus master. The parameter default settings are stored in the GSD file.

- **CRC_16** data check**:**
  If this function is activated, the correctness of the read or written data is ensured by a CRC_16 data check (see 📖 8).
- **Dynamic operation on read/write head 1 or 2:**
  If dynamic operation is parametered, a read/write job can be sent even though there is no Data carrier in the active zone of the head. As soon as a Data carrier passes by the head, the command is immediately carried out.
- **"Auto-Read" for read/write head 1 or 2:**
  If this function is activated, the processor reads out the first (max. 31) bytes from the Data carrier starting at a defined start address as soon as the tag enters the active zone of the read/write head. The start address must first have been stored in the processor's EEPROM with the command ID 07_{Hex}.
- **2nd bit header at end of in- and output buffer:**
  The 2nd bit header (factory setting) prevents data from being accepted by the bus as long as it is not fully updated.
- **Display state of the digital input in the bit header of the input buffer:**
  If this function is activated, the IN-bit displays the state of the digital input of the processor: IN = 0 → digital input low; IN = 1 → digital input high

Please note the basic procedure on 📖 12 and 30...36 and the examples on pages 📖 37...50.

Function Description
Parametering the BIS L-60_2 processor

Parameters, Overview (continued)

- **Reset BIS L-60_2 processor through the digital input:**
  If this function is activated, the processor is reset when the digital input is set to high.
- **Selecting the data carrier type for processing:**
  Depending on the selection, either all or only specified data carriers may be processed.
- **Output data carrier model and serial number:**
  If this function is activated, at CT Present the data carrier model and serial number (UID = unique ID) are output.
  At data carrier model BIS L-1_ _-01 the serial number is 4 bytes. At all other data carrier models the serial number is 8 bytes.

☞ If this function is activated and dynamic mode is not set, no read data are output at CT Present, but rather only the model and UID.

Please note the basic procedure on 📖 12 and 30...36 and the examples on pages 📖 37...50.

Function Description
Parametering, Parametering Bytes

Parametering Bytes
User-Parameter Bytes

> ⚠ For parametering all 6 bytes must always be transferred in Hex. Only the bits mentioned may be changed. No guaranty will be given for the proper functioning of the BIS L-60_2 if any of the other bits are changed.

The default values (factory setting) for the 6 bytes are:

	1st byte	**2nd byte**	3rd byte	**4th byte**	**5th byte**	**6th byte**
HEX	00	80	00	82	00	02
Binary	000**0**0000	100**00**000	**00000000**	**10**0000**10**	000**00**000	**00000010**
These are used for configuration:		**bit 4**	**bit 1...8**	**bit 7** **bit 1**	**bit 4**	**bit 1...8**
	bit 5	**bit 5**		**bit 8** **bit 2**	**bit 5**	

Having the following functions:

The bits which serve for parametering have the following functions:

1st byte, bit 5, Activate CRC_16 data checking

2nd byte, bit 5, Dynamic mode on read/write head 1
(for effects on read/write times, see 51)

Bit state: 0 = no
1 = yes

2nd byte, bit 4, Activate Auto-Read function starting at specified address after CT present for Head 1 (the number of bytes read depends on the selected buffer size minus bit headers for Head 1)

3nd byte, bit 1...8 Select data-carrier model for processing:
00Hex: All data-carrier models
FEHex: Mifare: All Mifare data carriers supported by Balluff.
FFHex: ISO15693: All ISO15693 data carriers supported by Balluff.

Please note the basic procedure on 12 and 30...36 and the examples on pages 37...50.

Function Description
Parametering, Parametering Bytes

Parametering Bytes
User-Parameter Bytes
(continued)

4th byte, bit 8, Arrange a 2nd bit header at the end of the input and output buffers

> If this function is selected, then the minimum size of both buffers is 4 words (8 bytes) each.

4th byte, bit 7, Display state of the digital input in the bit header of the input buffers:
0 = no
1 = yes Input is Low: "IN" in the bit header of the input buffers = 0.
Input is High: "IN" in the bit header of the input buffers = 1.

4th byte, bit 2, Reset the BIS L-60_2 processor through the digital input:
0 = no
1 = yes Input is Low: Do not reset.
Input is High: Reset.

Bit state: 0 = no
1 = yes

4th byte, bit 1, Output data-carrier model and serial number at CT present:
0 = no At CT present the first data carrier data are output on Profibus.
1 = yes At CT present the data carrier model and serial number (UID = unique ID) are output on Profibus. A distinction is made between type 01 with a 4-byte UID and type 03 with a 5-byte UID.

5th byte, bit 5 Dynamic mode on read/write head 2
(for effects on read/write times, see 51)

5th byte, bit 4 Activate Auto-Read function for Head 2 starting at specified address after CT present (the number of bytes read depends on the selected buffer size minus bit headers for Head 2)

Please note the basic procedure on 12 and 30...36 and the examples on pages 37...50.

Function Description
Parametering, Parametering Bytes

Parametering Bytes
User-Parameter Bytes
(continued)

6th byte, bit 1...6 No. of bytes in input and output buffer which shall be used for read/write head 1, see example on 📄 16

The specification for the input and output buffer on the Master applies to both read/write heads, i.e. this buffer must be divided for both heads. The specification is done in Hex format and must be in a range between 02_{Hex} and 80_{Hex} (128 dec.).

☞ If only one read/write head (Head 1) will be used, you may enter the same value here as for the total buffer size. An entry of less than 2 bytes results in an undefined state.

Please note the basic procedure on 📄📄 12 and 30...36 and the examples on pages 📄📄 37...50.

Data-carrier models
BIS L-10_-01/L

Data carrier BIS L-10_-01/L

Model BIS L-10_-01/L data carriers have a memory capacity of 192 bytes of user data. These data can be read or programmed. These data carriers also have a unique, 4-byte serial number, which is read-only.

The data carrier also contains additional memory ranges for configuration and protected data. These areas cannot be processed using the BIS L-60_2 processor.

Model BIS L-10_-01/L data carriers are supplied with FF_{Hex} 37_{Hex} configuration. Only data carriers having this configuration are processed.

CT present

At CT Present the first user data are read from the data carrier and stored in the Profibus input buffer (see 📄 31). If the “Output data-carrier model and serial number at CT present” function is enabled, model 01_{Hex} is output in Byte 1 of the input buffer and then the 4 bytes representing the unique serial number.

Functions

The full command set of the BIS L-60_2 processor can be used with model BIS L-10_01/L data carriers.

Device parameters

When using model BIS L-10_-01/L data carriers, the device parameterizing depends mainly on the number of bytes to be read and programmed per head.

☞ Please refer to 📄📄 13ff and 📄📄 23ff.

Data-carrier models
BIS L-20_-03/L

Data carrier BIS L-20_-03/L

Model BIS L-20_-03/L data carriers have a unique serial number consisting of 5 bytes. These are read-only and are considered like user data.

CT present

At CT Present the 5 bytes of the serial number are read from the data carrier and stored in the Profibus input buffer (see 📖 31). If the “Output data-carrier model and serial number at CT present” function is enabled, model 03_{Hex} is output in Byte 1 of the input buffer and then the 5 bytes representing the unique serial number.

Functions

With model BIS L-20_-03/L data carriers, all data are read and output as soon as CT present occurs. No other BIS L-60_2 processor commands are usable.

Device parameters

When using model BIS L-20_-03/L data carriers, set the following parameters:

Total buffer size on Profibus: 16 bytes
(8 bytes if only one read/write head is used)

Parameterizing bytes: **00_{Hex} 80_{Hex} 00_{Hex} 82_{Hex} 00_{Hex} 08_{Hex}**

or: **00_{Hex} 80_{Hex} *03*$_{Hex}$ 82_{Hex} 00_{Hex} 08_{Hex}**

→ only model BIS L-20_-03/L data carriers are processed.

00_{Hex} 80_{Hex} 00_{Hex} *83*$_{Hex}$ 00_{Hex} 08_{Hex}

→ output data carrier model and serial number at CT present.

Function Description
Processing data carriers

Reading and writing

To carry out a read or write job, the data carrier must be located in the active zone of the read/write head.

A read/write job has the following sequence (see examples on 📖 37ff):

1. The host sends to the output buffer:
   - the command designator to subaddress 01_{Hex},
   - the start address for reading or writing to subaddress 02_{Hex}/03_{Hex},
   - the number of bytes for reading or writing to subaddress 04_{Hex}/05_{Hex},
   - and sets the AV bit in the bit header to high.
2. The processor:
   - takes the request (AA in the bit header of the input buffer to high),
   - begins to transport the data;
     read = from data carrier to input buffer,
     write = from output buffer to data carrier.
     (Larger data quantities are sent in blocks
     block size with 2nd bit header = buffer size - 2,
     block size without 2nd bit header = buffer size - 1).
     The toggle bits in the two bit headers are used as a kind of handshaking between the host and the BIS L-60_2 processor.
3. The processor has processed the command correctly (AE bit in the bit header of the input buffer). If an error occurred during execution of the command, an error number will be written to subaddress 01_{Hex} of the input buffer and the AF bit in the bit header of the input buffer will be set.

Function Description
Processing data carriers

Codetag Present (CP bit)

As soon as the data carrier enters the active one of the read/write head, the processor indicates this by setting the CP bit (Codetag Present).

☞ To accelerate the reading of small amounts of data, the ID system makes the first bytes of the data carrier available in the input buffer of the respective read/write head as soon as the tag is detected. The number of bytes sent corresponds to the configured buffer size – 1 byte (2 bytes for 2nd bit header).

If a model BIS L-20_-03L is present at the read/write head, a maximum of 5 bytes are output.

If the parameter “Output data-carrier model and serial number at CT Present” is set, the data-carrier model and unique serial number are output instead of the read data. For model BIS L-20_-03/L data carriers the read data correspond to the serial number.

The data are only valid after the rising edge of the CP bit in the bit header of the input buffer. They remain valid until the falling edge of the CP bit, or until the controller issues a new job.

Start address for Auto-Read

If the Auto-Read function is activated, the data are read starting with a specified start address as soon as the data carrier is recognized. The rising edge of the CP bit is used to provide these data in the input buffer. The start address must be specified for each head using command identifier 07_{Hex} the start addresses may be different. The number of bytes read is determined by the selected size of the input buffer, which is distributed over both heads when 2 are used.

Function Description
Processing data carriers

Reading and writing in dynamic mode

In normal operation a read/write job is rejected by the BIS L-60_2 processor by setting the AF bit and an error number if there is no data carrier in the active zone of the read/write head. If dynamic mode is configured, the processor accepts the read/write job and stores it. When a data carrier is recognized, the stored job is carried out.

Reading and writing with simultaneous data transmission

Reading without simultaneous data transmission: In the case of a read job the processor first reads our all requested data from the data carrier after receiving the start address and the desired number of bytes, and then sets the AE bit. Then the data read from the data carrier are written to the input buffer. In the case of larger data amounts this is done in blocks, controlled by the handshake with the toggle bits as described on 🗎 30.

Reading with simultaneous data transmission: In the case of a read job the processor begins by transmitting the data into the input buffer as soon as the first 30 bytes (with 2nd bit header, or 31 bytes without 2nd bit header, or less if the buffer size was set smaller) have been read from the data carrier beginning with the start address, and indicates this by inverting the TO bit. As soon as the controller inverts the TI bit, the processor sends the data, which have in the meantime been read, to the input buffer. This is repeated until the processor has read out all the desired data from the data carrier. Now the processor sets the AE bit and outputs the remaining data on the input buffer.

Writing without simultaneous data transmission: In the case of a write job the processor waits until it has received all the data that need to be written from the controller. Only then are the data written to the data carrier as described on 🗎 30.

Writing with simultaneous data transmission: In the case of a write job the processor begins to write the data to the data carrier as soon as it has received the first data to be written from the controller’s output buffer. Once all the data have been written to the data carrier, the AE bit is set.

Function Description
Processing data carriers

Mixed Data Access

Small read/write programs can be stored in the BIS L-60_2 processor's EEPROM.

The Mixed Data Access function is useful when the required information is stored on the data carrier at various addresses. This function makes it possible to read out this "mixed", i.e. non-contiguously stored data from the data carrier in a single procedure and using just one command.

Up to 10 programs with up to 25 instructions can be stored. Each program instruction contains a "start address" and a "number of bytes" specification. The amount of data for reading may not exceed 2 kB.

Storing a program:
The command identifier 06$_{\text{Hex}}$ is used to send the read/write program to the BIS L-60_2 processor. One program per command can be stored. All 25 program records plus an additional 2 bytes with FF$_{\text{Hex}}$FF$_{\text{Hex}}$ as a terminator must always be sent. This means a total of **104 bytes** of information per program must be sent (including the command identifier and program number).

☞

> The individual program records must all be contiguous. They must be sent one after the other and be terminated with FF$_{\text{Hex}}$FF$_{\text{Hex}}$ as a terminator. It is recommended that the remaining, unused memory sector be filled with FF$_{\text{Hex}}$FF$_{\text{Hex}}$.
>
> If an address range is selected twice, the data will also be output twice.

Function Description
Processing data carriers

Mixed Data Access
(continued)

The following shows the structure of a program:

Program structure	Subaddress	Value	Range
Command designator	01$_{\text{Hex}}$	06$_{\text{Hex}}$	
1. Program record			
Program number	02$_{\text{Hex}}$	01$_{\text{Hex}}$	01$_{\text{Hex}}$ to 0A$_{\text{Hex}}$
1st data record:			
Start address Low Byte	03$_{\text{Hex}}$		
Start address High Byte	04$_{\text{Hex}}$		
Number of bytes Low Byte	05$_{\text{Hex}}$		
Number of bytes High Byte	06$_{\text{Hex}}$		
2nd data record:			
...			
25th data record:			
Start address Low Byte	03$_{\text{Hex}}$		
Start address High Byte	04$_{\text{Hex}}$		
Number of bytes Low Byte	05$_{\text{Hex}}$		
Number of bytes High Byte	06$_{\text{Hex}}$		
Terminator	FF$_{\text{Hex}}$ FF$_{\text{Hex}}$		

To store a second program, repeat this process.

The procedure for writing these settings to the EEPROM is described in the 7th example on 🗍 45...47.

Replacing the EEPROM is described on 🗋 58 for BIS L-6002 and on 🗋 68 for BIS L-6022.

Function Description
Processing data carriers

Read from data carrier, with program Mixed Data Access

The command identifier 21_{Hex} can be used to read out the program records stored in the program from the data carrier. The user must document exactly which data are to be read from where and with what number of bytes for the respective program (see example 8 on ⎘ 48)

Write to data carrier, with program Mixed Data Access

The command identifier 22_{Hex} can be used to write the program records stored in the program to the data carrier. The user must document exactly which data are to be written from where and with what number of bytes for the respective program (see example 9 on ⎘ 49)

CRC_16 initialization

To be able to use the CRC_16 check, the data carrier must first be initialized with the command identifier 12_{Hex} (see ⎘ 37). The CRC_16 initialization is used like a normal write job. The latter is rejected (with an error message) if the processor recognizes that the data carrier does not contain the correct CRC_16 checksum. Data carriers as shipped from the factory (all data are 0) can immediately be written with CRC-checked data.

If CRC_16 data checking is activated, a special error message is output to the interface whenever a CRC_16 error is detected.

If the error message is not caused by a failed write request, it may be assumed that one or more memory cells on the data carrier is defective. That data carrier must then be replaced.

If the CRC error is however due to a failed write request, you must reinitialize the data carrier in order to continue using it.

Data-carrier models
BIS L-10_-01/L

Data carrier BIS L-10_-01/L

Model BIS L-10_-01/L data carriers have a memory capacity of 192 bytes of user data. These data can be read or programmed. These data carriers also have a unique, 4-byte serial number, which is read-only.

The data carrier also contains additional memory ranges for configuration and protected data. These areas cannot be processed using the BIS L-60_2 processor.

Model BIS L-10_-01/L data carriers are supplied with FF_{Hex} 37_{Hex} configuration. Only data carriers having this configuration are processed.

CT present

At CT Present the first user data are read from the data carrier and stored in the Profibus input buffer (see ⎘ 31). If the “Output data-carrier model and serial number at CT present” function is enabled, model 01_{Hex} is output in Byte 1 of the input buffer and then the 4 bytes representing the unique serial number.

Functions

The full command set of the BIS L-60_2 processor can be used with model BIS L-10_01/L data carriers.

Device parameters

When using model BIS L-10_-01/L data carriers, the device parameterizing depends mainly on the number of bytes to be read and programmed per head.

☞ Please refer to ⎘⎘ 13ff and ⎘⎘ 23ff.

Function Description
Examples for protocol sequence

Example No. 1

For configuring with double bit header and 8-byte buffer size!

Initializing the data carrier for the CRC_16 data checking

The processing of this command is similar to a write command. Start address and number of bytes have to correspond to the maximum number of data to be used.
In this example the complete memory range of a data carrier with 192 bytes shall be used (BIS L-10 _-01/L). Because 2 bytes are used for the CRC only 168 bytes can be used as data bytes, hence: start address = 0, number of bytes = 168.

Host:

1.) Process subaddresses of the output buffer in the order shown:

01$_{Hex}$	Command designator 12$_{Hex}$
02$_{Hex}$	Start address 00$_{Hex}$
03$_{Hex}$	Start address 00$_{Hex}$
04$_{Hex}$	No. of bytes 92$_{Hex}$
05$_{Hex}$	No. of bytes 02$_{Hex}$
00$_{Hex}$/07$_{Hex}$	Set AV-Bit

BIS L-60_2 Identification System:

2.) Process subaddresses of the input buffer in the order shown:

00$_{Hex}$/07$_{Hex}$	Set AA-Bit, invert TO-Bit

3.) Process subaddresses of the output buffer:

01...06$_{Hex}$	Enter first 6 bytes of data
00$_{Hex}$/07$_{Hex}$	Invert TI-Bit

4.) Process subaddresses of the output buffer:

01...06$_{Hex}$	Copy first 6 data bytes

Process subaddress of the input buffer:

00$_{Hex}$/07$_{Hex}$	Invert TO-Bit

5.) Process subaddresses of the output buffer:

01...06$_{Hex}$	Enter the second 6 data bytes
00$_{Hex}$/07$_{Hex}$	Invert TI-Bit

6.) Process subaddresses of the output buffer:

01...06$_{Hex}$	Copy second 6 data bytes

Process subaddress of the input buffer:

00$_{Hex}$/07$_{Hex}$	Invert TO-Bit

... To be continued until the complete memory range is written. See next 📄.

Function Description
Examples for protocol sequence

Example No. 1 (continued)

For configuring with double bit header and 8-byte buffer size!

Host:

7.) Process subaddresses of the output buffer:

01...06$_{Hex}$	Enter the remaining data byte
00$_{Hex}$/07$_{Hex}$	Invert TI-Bit

BIS L-60_2 Identification System:

8.) Process subaddresses of the output buffer:

01...06$_{Hex}$	Copy the remaining data byte

Process subaddress of the input buffer:

00$_{Hex}$/07$_{Hex}$	Set AE-Bit

9.) Process subaddresses of the output buffer:

00$_{Hex}$/07$_{Hex}$	Reset AV-Bit

10.) Process subaddresses of the input buffer:

00$_{Hex}$/07$_{Hex}$	Reset AA-Bit and AE-Bit

Function Description
Examples for protocol sequence

Example No. 2

For configuring with double bit header and 8-byte buffer size!

Read 17 bytes starting at data carrier address 10 (data carrier type BIS L-10 _-01/L)**:**

Host:

1.) Process subaddresses of the output buffer in the order shown:

01$_{Hex}$	Command designator 01$_{Hex}$
02$_{Hex}$	Start address Low Byte 0A$_{Hex}$
03$_{Hex}$	Start address High Byte 00$_{Hex}$
04$_{Hex}$	No. of bytes Low Byte 11$_{Hex}$
05$_{Hex}$	No. of bytes High Byte 00$_{Hex}$
00$_{Hex}$/07$_{Hex}$	Set AV-Bit

BIS L-60_2 Identification System:

2.) Process subaddresses of the input buffer in the order shown:

00$_{Hex}$/07$_{Hex}$	Set AA-Bit
01...06$_{Hex}$	Enter first 6 bytes of data
00$_{Hex}$/07$_{Hex}$	Set AE-Bit

3.) Process subaddresses of the input buffer:

01...06$_{Hex}$	Copy first 6 data bytes

Process subaddress of the output buffer:

00$_{Hex}$/07$_{Hex}$	Invert TI-Bit

4.) Process subaddresses of the input buffer:

01...06$_{Hex}$	Enter the second 6 data bytes
00$_{Hex}$/07$_{Hex}$	Invert TO-Bit

5.) Process subaddresses of the input buffer:

01...06$_{Hex}$	Copy second 6 data bytes

Process subaddress of the output buffer:

00$_{Hex}$/07$_{Hex}$	Invert TI-Bit

6.) Process subaddresses of the input buffer:

01...05$_{Hex}$	Enter the remaining 5 data bytes
00$_{Hex}$/07$_{Hex}$	Invert TO-Bit

7.) Process subaddresses of the input buffer:

01...05$_{Hex}$	Copy the remaining 5 data bytes

Process subaddress of the output buffer:

00$_{Hex}$/07$_{Hex}$	Reset AV-Bit

8.) Process subaddresses of the input buffer:

00$_{Hex}$/07$_{Hex}$	Reset AA-Bit and AE-Bit

Function Description
Examples for protocol sequence

Example No. 3 like 2nd example but with simultaneous data transmission

For configuring with double bit header and 8-byte buffer size!

Read 17 bytes starting at data carrier address 10, with simultaneous data transmission (data carrier type BIS L-10 _-01/L)**:**

While the read job is being carried out and as soon as the input buffer is filled, the first data are sent. The AE bit is not set until the "Read" operation is completed by the processor.

The reply "Job End" = AE bit is reliably set no later than before the last data are sent. The exact time depends on the requested data amount, the input buffer size and the timing of the controller. This is indicated in the following by the note *Set AE-Bit* (in italics).

Host:

1.) Process subaddresses of the output buffer in the order shown:

01$_{Hex}$	Command designator 01$_{Hex}$
02$_{Hex}$	Start address Low Byte 0A$_{Hex}$
03$_{Hex}$	Start address High Byte 00$_{Hex}$
04$_{Hex}$	No. of bytes Low Byte 11$_{Hex}$
05$_{Hex}$	No. of bytes High Byte 00$_{Hex}$
00$_{Hex}$/07$_{Hex}$	Set AV-Bit

BIS L-60_2 Identification System:

2.) Process subaddresses of the input buffer in the order shown:

00$_{Hex}$/07$_{Hex}$	Set AA-Bit
01...06$_{Hex}$	Enter first 6 bytes of data
00$_{Hex}$/07$_{Hex}$	Invert TO-Bit
00$_{Hex}$/07$_{Hex}$	*Set AE-Bit*

3.) Process subaddresses of the input buffer:

01...06$_{Hex}$	Copy first 6 data bytes

Process subaddress of the output buffer:

00$_{Hex}$/07$_{Hex}$	Invert TI-Bit

4.) Process subaddresses of the input buffer:

01...06$_{Hex}$	Enter the second 6 data bytes
00$_{Hex}$/07$_{Hex}$	Invert TO-Bit
00$_{Hex}$/07$_{Hex}$	*Set AE-Bit*

Continued on next 🗋.

Function Description
Examples for protocol sequence

Example No. 3
(continued)

like 2nd example but with simultaneous data transmission

For configuring with double bit header and 8-byte buffer size!

Host:

5.) Process subaddresses of the input buffer:

01...06$_{Hex}$	Copy second 6 data bytes

Process subaddress of the output buffer:

00$_{Hex}$/07$_{Hex}$	Invert TI-Bit

7.) Process subaddresses of the input buffer:

01...05$_{Hex}$	Copy the remaining 5 data bytes

Process subaddress of the output buffer:

00$_{Hex}$/07$_{Hex}$	Reset AV-Bit

BIS L-60_2 Identification System:

6.) Process subaddresses of the input buffer:

01...05$_{Hex}$	Enter the remaining 5 data bytes
00$_{Hex}$/07$_{Hex}$	Invert TO-Bit
00$_{Hex}$/07$_{Hex}$	*Set AE-Bit*

8.) Process subaddresses of the input buffer:

00$_{Hex}$/07$_{Hex}$	Reset AA-Bit and AE-Bit

Function Description
Examples for protocol sequence

Example No. 4

For configuring with double bit header and 8-byte buffer size!

Read 30 bytes starting at data carrier address 10 with read error
(data carrier type BIS L-10 _-01/L)**:**

Host:

1.) Process subaddresses of the output buffer in the order shown:

01$_{Hex}$	Command designator 01$_{Hex}$
02$_{Hex}$	Start address Low Byte 0A$_{Hex}$
03$_{Hex}$	Start address High Byte 00$_{Hex}$
04$_{Hex}$	No. of bytes Low Byte 1E$_{Hex}$
05$_{Hex}$	No. of bytes High Byte 00$_{Hex}$
00$_{Hex}$/07$_{Hex}$	Set AV-Bit

3.) Process subaddress of the input buffer:

01$_{Hex}$	Copy error number

Process subaddress of the output buffer:

00$_{Hex}$/07$_{Hex}$	Reset AV-Bit

BIS L-60_2 Identification System:

2.) Process subaddresses of the input buffer in the order shown:

If an error occurs right away:

00$_{Hex}$/07$_{Hex}$	Set AA-Bit
01$_{Hex}$	Enter error number
00$_{Hex}$/07$_{Hex}$	Set AF-Bit

4.) Process subaddresses of the input buffer:

00$_{Hex}$/07$_{Hex}$	Reset AA-Bit and AE-Bit

Function Description
Examples for protocol sequence

Example No. 5

For configuring with double bit header and 8-byte buffer size!

Write 16 bytes starting at data carrier address 20 (data carrier type BIS L-10 _-01/L)**:**

Host:

1.) Process subaddresses of the output buffer in the order shown:

01_{Hex}	Command designator 02_{Hex}
$02_{\text{Hex}}/03_{\text{Hex}}$	Start address 14_{Hex} / 00_{Hex}
$04_{\text{Hex}}/05_{\text{Hex}}$	No. of bytes 10_{Hex} / 00_{Hex}
$00_{\text{Hex}}/07_{\text{Hex}}$	Set AV-Bit

BIS L-60_2 Identification System:

2.) Process subaddresses of the input buffer in the order shown:

$00_{\text{Hex}}/07_{\text{Hex}}$	Set AA-Bit, invert TO-Bit

3.) Process subaddresses of the output buffer:

01...06_{Hex}	Enter the first 6 data bytes
$00_{\text{Hex}}/07_{\text{Hex}}$	Invert TI-Bit

4.) Process subaddresses of the output buffer:

01...06_{Hex}	Copy first 6 data bytes

Process subaddress of the input buffer:

$00_{\text{Hex}}/07_{\text{Hex}}$	Invert TO-Bit

5.) Process subaddresses of the output buffer:

01...06_{Hex}	Enter the second 6 data bytes
$00_{\text{Hex}}/07_{\text{Hex}}$	Invert TI-Bit

6.) Process subaddresses of the output buffer:

01...06_{Hex}	Copy second 6 data bytes

Process subaddress of the input buffer:

$00_{\text{Hex}}/07_{\text{Hex}}$	Invert TO-Bit

7.) Process subaddresses of the output buffer:

01...04_{Hex}	Enter the remaining 4 data bytes
$00_{\text{Hex}}/07_{\text{Hex}}$	Invert TI-Bit

8.) Process subaddresses of the output buffer:

01...04_{Hex}	Copy the remaining 4 data bytes

Process subaddress of the input buffer:

$00_{\text{Hex}}/07_{\text{Hex}}$	Set AE-Bit

9.) Process subaddresses of the output buffer:

$00_{\text{Hex}}/07_{\text{Hex}}$	Reset AV-Bit

10.) Process subaddresses of the input buffer:

$00_{\text{Hex}}/07_{\text{Hex}}$	Reset AA-Bit and AE-Bit

Function Description
Examples for protocol sequence

Example No. 6
Address assignment for the Auto-Read function

For configuring with double bit header and 8-byte buffer size!

Programming start address 75:

Host:

1.) Process subaddresses of the output buffer in the order shown:

01_{Hex}	Command designator 07_{Hex}
02_{Hex}	Start address Low Byte $4B_{\text{Hex}}$
03_{Hex}	Start address High Byte 00_{Hex}
$00_{\text{Hex}}/07_{\text{Hex}}$	Set AV-Bit

BIS L-60_2 Identification System:

2.) Process subaddresses of the input buffer:

$00_{\text{Hex}}/07_{\text{Hex}}$	Set AA-Bit and AE-Bit

3.) Process subaddresses of the output buffer:

$00_{\text{Hex}}/07_{\text{Hex}}$	Reset AV-Bit

4.) Process subaddresses of the input buffer:

$00_{\text{Hex}}/07_{\text{Hex}}$	Reset AA-Bit and AE-Bit

☞ To ensure correct data output, use command identifier 07_{Hex} for each distributed buffer Head 1 and/or Head 2.

If the Auto-Read function is not activated, the processor runs in standard mode and sends starting with data carrier address 0 until the buffer is filled.

Function Description
Examples for protocol sequence

Example No. 7
Store Mixed Data Access program

For configuring with double bit header and 8-byte buffer size!

Storing a program for reading out 3 data records:

1st data record	Start address	5	Number of bytes	7
2nd data record	Start address	75	Number of bytes	3
3rd data record	Start address	112	Number of bytes	17

Total number of bytes exchanged in the operation: 27 bytes

All 104 bytes are written for the programming.

Host:

1.) Process subaddresses of the output buffer in the order shown:

01$_{Hex}$	Command designator 06$_{Hex}$
02$_{Hex}$	Program number 01$_{Hex}$
00$_{Hex}$/07$_{Hex}$	Set AV-Bit

BIS L-60_2 Identification System:

2.) Process subaddresses of the input buffer:

00$_{Hex}$/07$_{Hex}$	Set AA-Bit, invert TO-Bit

3.) Process subaddresses of the output buffer:

01$_{Hex}$	1st start address	(Low Byte) 05$_{Hex}$
02$_{Hex}$		(High Byte) 00$_{Hex}$
03$_{Hex}$	1st number of bytes	(Low Byte) 07$_{Hex}$
04$_{Hex}$		(High Byte) 00$_{Hex}$
05$_{Hex}$	2nd start address	(Low Byte) 4B$_{Hex}$
06$_{Hex}$		(High Byte) 00$_{Hex}$
00$_{Hex}$/07$_{Hex}$	Invert TI-Bit	

4.) Process subaddresses of the input buffer:

00$_{Hex}$/07$_{Hex}$	Invert TO-Bit

Continued on next

Function Description
Examples for protocol sequence

Example No. 7
Store Mixed Data Access program
(continued)

For configuring with double bit header and 8-byte buffer size!

Host:

5.) Process subaddresses of the output buffer:

01$_{Hex}$ 02$_{Hex}$	2nd number of bytes	(Low Byte) 03$_{Hex}$ (High Byte) 00$_{Hex}$
03$_{Hex}$	3rd start address	(Low Byte) 70$_{Hex}$
04$_{Hex}$		(High Byte) 00$_{Hex}$
05$_{Hex}$	3rd number of bytes	(Low Byte) 11$_{Hex}$
06$_{Hex}$		(High Byte) 00$_{Hex}$
00$_{Hex}$/07$_{Hex}$	Invert TI-Bit	

BIS L-60_2 Identification System:

6.) Process subaddresses of the input buffer:

00$_{Hex}$/07$_{Hex}$	Invert TO-Bit

7.) Process subaddresses of the output buffer:

01$_{Hex}$/02$_{Hex}$	Terminator	FF$_{Hex}$/FF$_{Hex}$
03$_{Hex}$/04$_{Hex}$	(not used)	FF$_{Hex}$/FF$_{Hex}$
05$_{Hex}$/06$_{Hex}$	(not used)	FF$_{Hex}$/FF$_{Hex}$
00$_{Hex}$/07$_{Hex}$	Invert TI-Bit	

8.) Process subaddresses of the input buffer:

00$_{Hex}$/07$_{Hex}$	Invert TO-Bit

Fill all unused start addresses and number of bytes with FF$_{Hex}$!

Continued on next

Function Description
Examples for protocol sequence

Example No. 7
Store Mixed Data Access program
(continued)

For configuring with double bit header and 8-byte buffer size!

Host:

9.) Process subaddresses of the output buffer:

01$_{\text{Hex}}$/02$_{\text{Hex}}$	(not used)	FF$_{\text{Hex}}$/FF$_{\text{Hex}}$
03$_{\text{Hex}}$/04$_{\text{Hex}}$	(not used)	FF$_{\text{Hex}}$/FF$_{\text{Hex}}$
05$_{\text{Hex}}$/06$_{\text{Hex}}$	(not used)	FF$_{\text{Hex}}$/FF$_{\text{Hex}}$
00$_{\text{Hex}}$/07$_{\text{Hex}}$	Invert TI-Bit	

11.) Process subaddresses of the output buffer:

00$_{\text{Hex}}$/07$_{\text{Hex}}$	Reset AV-Bit

BIS L-60_2 Identification System:

10.) Process subaddresses of the input buffer:

00$_{\text{Hex}}$/07$_{\text{Hex}}$	Set AE-Bit

12.) Process subaddresses of the input buffer:

00$_{\text{Hex}}$/07$_{\text{Hex}}$	Reset AA-Bit and AE-Bit

☞ We recommend that you carefully document which parameters are used for start addresses and number of bytes for writing/reading the desired data records.

The data are sequenced in the exact order specified in the program.

Function Description
Examples for protocol sequence

Example No. 8
Use Mixed Data Access program

For configuring with double bit header and 8-byte buffer size!

Read data carrier using Program No. 1 (data carrier type BIS L-10 _-01/L)**:**

Host:

1.) Process subaddresses of the output buffer in the order shown:

01$_{\text{Hex}}$	Command designator 21$_{\text{Hex}}$
02$_{\text{Hex}}$	Program number 01$_{\text{Hex}}$
00$_{\text{Hex}}$/07$_{\text{Hex}}$	Set AV-Bit

BIS L-60_2 Identification System:

2.) Process subaddresses of the input buffer in the order shown:

00$_{\text{Hex}}$/07$_{\text{Hex}}$	Set AA-Bit
01...06$_{\text{Hex}}$	Enter first 6 bytes of data
00$_{\text{Hex}}$/07$_{\text{Hex}}$	Set AE-Bit

3.) Process subaddresses of the input buffer:

01...06$_{\text{Hex}}$	Copy first 6 data bytes

Process subaddress of the output buffer:

00$_{\text{Hex}}$/07$_{\text{Hex}}$	Invert TI-Bit

4.) Process subaddresses of the output buffer:

01...06$_{\text{Hex}}$	Enter the second 6 data bytes
00$_{\text{Hex}}$/07$_{\text{Hex}}$	Invert TO-Bit

... A total of 27 bytes of data are exchanged.
For the remainder of the procedure, see Example 2 on ⎘39.

☞ Dynamic mode is turned off while the Mixed Data Access program is being run.

Function Description
Examples for protocol sequence

Example No. 9
Use Mixed Data Access program

For configuring with double bit header and 8-byte buffer size!

Write data carrier using Program No. 1 (data carrier type BIS L-10 _-01/L)**:**

Host:

1.) Process subaddresses of the output buffer in the order shown:

01$_{Hex}$	Command designator 22$_{Hex}$
02$_{Hex}$	Program number 01$_{Hex}$
00$_{Hex}$/07$_{Hex}$	Set AV-Bit

BIS L-60_2 Identification System:

2.) Process subaddresses of the input buffer in the order shown:

00$_{Hex}$/07$_{Hex}$	Set AA-Bit, invert TO-Bit

3.) Process subaddresses of the output buffer:

01...06$_{Hex}$	Enter first 6 bytes of data
00$_{Hex}$/07$_{Hex}$	Invert TI-Bit

4.) Process subaddresses of the output buffer:

01...06$_{Hex}$	Copy first 6 data bytes

Process subaddress of the input buffer:

00$_{Hex}$/07$_{Hex}$	Invert TO-Bit

... A total of 27 bytes of data are exchanged.
For the remainder of the procedure, see Example 5 on 🗋 43.

☞ Dynamic mode is turned off while the Mixed Data Access program is being run.

Function Description
Examples for protocol sequence

Example No. 10

Put the relevant read/write head into ground state:
Both read/write heads can be independently set to the ground state.

Host:

1.) Process subaddresses of the output buffer:

00$_{Hex}$/07$_{Hex}$	Set GR-Bit

BIS L-60_2 Identification System:

2.) Go to ground state;
Process subaddresses of the input buffer:

00$_{Hex}$/07$_{Hex}$	Reset BB-Bit

3.) Process subaddresses of the output buffer:

00$_{Hex}$/07$_{Hex}$	Reset GR-Bit

4.) Process subaddresses of the input buffer:

00$_{Hex}$/07$_{Hex}$	Set BB-Bit

Example No. 11

Read/write head deactivation:
In normal operation both heads are active. If the installation is less than ideal, there may be mutual interference between the heads. In this case the unused head should be turned off to prevent interference.

Host:

1.) Process subaddresses of the output buffer:

00$_{Hex}$/07$_{Hex}$	Set KA-Bit

Resetting the KA bit turns the read/write head back on. It may take up to a second to reactivate the head, whereas turning it off takes much less time.

Read/Write Times

Read times

Data carrier with each 16 bytes/block	BIS L-1_ _-01	BIS L-1_ _-02
Time for data carrier recognition/serial ID	≤ 20 ms	≤ 30 ms
Read bytes 0 to 15	≤ 20 ms	≤ 30 ms
For each additional 16 bytes add another	≤ 10 ms	≤ 15 ms

Data carrier BIS L-2_ _
Recognize data carrier + read data carrier ≈ 270 ms

Write times

Data carrier with each 16 bytes/block	BIS L-1_ _-01	BIS L-1_ _-02
Time for data carrier recognition/serial ID	≤ 20 ms	≤ 30 ms
Write bytes 0 to 15	≤ 40 ms	≤ 65 ms
For each additional 16 bytes add another	≤ 30 ms	≤ 45 ms

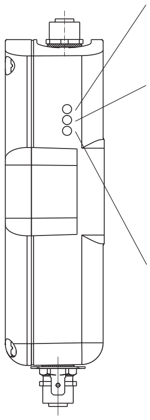
Data carrier BIS L-2_ _
Writing not possible

☞ All data are typical values. Deviations are possible depending on the application and combination of read/write head and data carrier!
The data apply to static operation, no CRC_16 data checking.

LED Display

Function displays on BIS L-60_2

The BIS L-60_2 uses the three side-mounted LED's to indicate important conditions of the identification system.



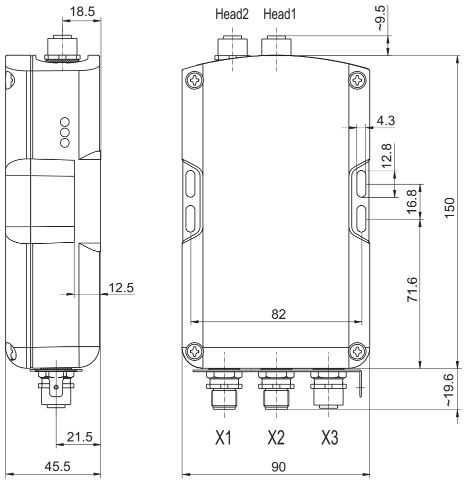
Status	LED	Meaning
Ready / Bus active	red	Supply voltage OK; no hardware error, however, bus not active.
	green	Supply voltage / hardware OK, bus active.
CT1 Present / operating	green	Data carrier read/write-ready at read/write head 1.
	yellow	Read/write command at read/write head 1 in process.
	yellow flashes [f ≈ 2 Hz]	Cable break to read/write head or not connected.
	yellow flashes faster [f ≈ 4 Hz]	Communication with R/W Head 1 is faulty or R/W Head 1 is defective.
	off	No data carrier in read/write range of read/write head 1.
CT2 Present / operating	green	Data carrier read/write-ready at read/write head 2.
	yellow	Read/write command at read/write head 2 in process.
	yellow flashes [f ≈ 2 Hz]	Cable break to read/write head or not connected.
	yellow flashes faster [f ≈ 4 Hz]	Communication with R/W Head 2 is faulty or R/W Head 2 is defective.
	off	No data carrier in read/write range of read/write head 2.

If all three LED's are synchronously flashing, it means a hardware error. Return the unit to the factory.

BIS L-6002
Mounting the Processor

Mounting the BIS L-6002 processor

The processor is attached using 4 M4 screws.



BIS L-6002 dimensions

BIS L-6002
Opening the Processor / Interface Information

Opening the BIS L-6002 processor

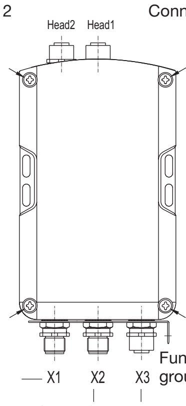
To set the PROFIBUS-DP address, activate or deactivate the internal termination resistor, or to change the EEPROM, you must open up the BIS L-6002 processor.

Remove the 4 screws on the BIS L-6002 and lift off the cover. See the following 📖 for additional information.

BIS L-6002 interfaces

Connection for read/write head 2

Connection for read/write head 1



Be sure before opening that the unit is disconnected from power. ⚠

Mounting of the cover (4 screws),
max. permissible tightening torque: 0.15 Nm

Supply voltage digital input

Function ground FE

PROFIBUS-DP Input

PROFIBUS-DP Output

Connection locations and names

BIS L-6002
Interface Information / Wiring Diagrams

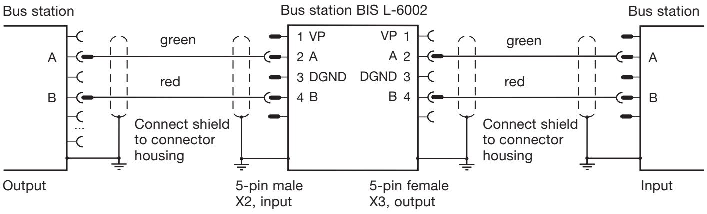
Wiring

To insert BIS L-6002 processor into the serial PROFIBUS and to connect the supply voltage and the digital input, the cables have to be connected to the terminals of the processor. The read/write heads have to be connected to the terminals Head 1 and Head 2.

PROFIBUS-DP

Ensure that the device is turned off.

To insert BIS L-6002 processor into the serial PROFIBUS-DP, there are the terminal X2 for the PROFIBUS input and the terminal X3 for the PROFIBUS output.



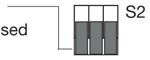
BIS L-6002
Interface Information / Wiring Diagrams

PROFIBUS-DP Terminating resistor

The last bus module must terminate the bus with a resistor. In the case of the BIS L-6002, this can be realized in two different ways:

1. **In the device** by closing the switch S2 (factory standard is open)

   Note: Output terminal must be closed off with a screw cover in order to maintain the enclosure rating.



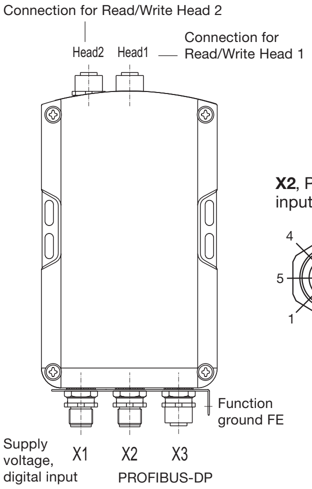
S2	Terminating resistor
closed	active
open	passive

2. **Outside the device** in a connector to socket X3. In this case the signal VP (pin 1) and DGND (pin 3) should be brought out in order to connect the external resistor to the potential.

   Note: In this case S2 has to be open!

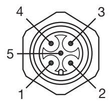
BIS L-6002
Interface Information / Wiring Diagrams

Wiring diagram for BIS L-6002 processor


Connection for Read/Write Head 2
Head2
Head1
Connection for Read/Write Head 1
Function ground FE
Supply voltage, digital input
X1
X2
X3
PROFIBUS-DP

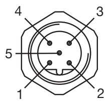
Terminal location and designation

X1, supply voltage, digital input

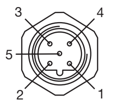


Pin	Function
1	+Vs
2	–IN
3	–Vs
4	+IN
5	n.c.

X2, PROFIBUS-input (male)



X3, PROFIBUS-output (female)



Pin	Function
1	VP
2	A
3	DGND
4	B
5	n.c.

n.c. = do not connect

The function-ground connector FE should be connected to earth directly or through a RC combination depending on the system (potential counterpoise).

When connecting the bus leads, make sure that the shield has proper connection to connector housing.

BIS L-6002
Changing the EEPROM

Changing the EEPROM in the BIS L-6002 processor

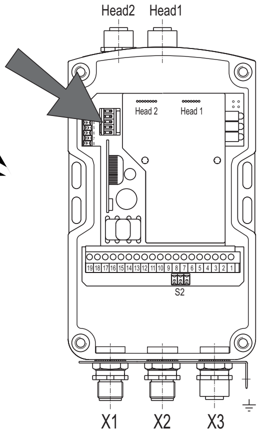
To replace the EEPROM, open up the processor as described on ⎍ 54.

⚠

Be sure before opening that the unit is disconnected from power.

To avoid damaging the EEPROM, please observe the requirements for handling electrostatically sensitive components.

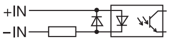
The EEPROM is replaced by unplugging and plugging back into the socket.



Location of the EEPROM

BIS L-6002 Technical Data

Dimensions, Weight	**Housing**	Plastic
	Dimensions	ca. 179 x 90 x 45,5 mm
	Weight	ca. 500 g
Operating Conditions	Ambient temperature	0 °C to + 60 °C
Enclosure Rating	Enclosure rating	IP 65 (when connected)
Connections	Integral connector X1 for **V_S, IN**	5-pin (male)
	Integral connector X2 for **PROFIBUS-DP** Input	5-pin (male)
	Integral connector X3 for **PROFIBUS-DP** Output	5-pin (female)
Electrical Connections	**Supply voltage V_s**	DC 24 V ± 20 % LPS Class 2
	Ripple	≤ 10 %
	Current draw	≤ 400 mA
	PROFIBUS-DP slave	Terminal block, electrically isolated
	Digital Input (+IN, –IN)	Optocoupler isolated
	Control voltage active	4 V to 40 V
	Control voltage inactive	1.5 V to –40 V
	Input current at 24 V	11 mA
	Delay time, typ.	5 ms
	Read/Write Head	2 x connectors 8-pin (female) for all read/writ heads BIS L-3_ _ with 8-pin connector (male)



BIS L-6002 Technical Data

Function displays	BIS operating messages:	
	Ready / Bus active	LED red / green
	CT1 Present / operating	LED green / yellow
	CT2 Present / operating	LED green / yellow

c UL US LISTED

Process Control Equipment

Control No 3TLJ

File No E227256

CE Declaration of Conformity and user safety

CE *This product was developed and produced considering the claimed European standards and guidelines.*

☞ You can separately request a Declaration of Conformity.

Further safety measures you can find in chapter *Safety* (see 🗋 4).

BIS L-6002
Ordering Information

Ordering Code

BIS L-6002-019-050-03-ST11

Balluff Identification System

Type L Read/Write System

Hardware-Typ
6002 = plastic housing, PROFIBUS-DP

Software-Typ
019 = PROFIBUS-DP

Read/Write Head
050 = with two connections for external read/write heads BIS L-3_ _

Interface
03 = BUS versions

User Connection
ST11 = Connector version X1, X2, X3 (2× male 5-pin, 1× female 5-pin)

BIS L-6002
Ordering Information

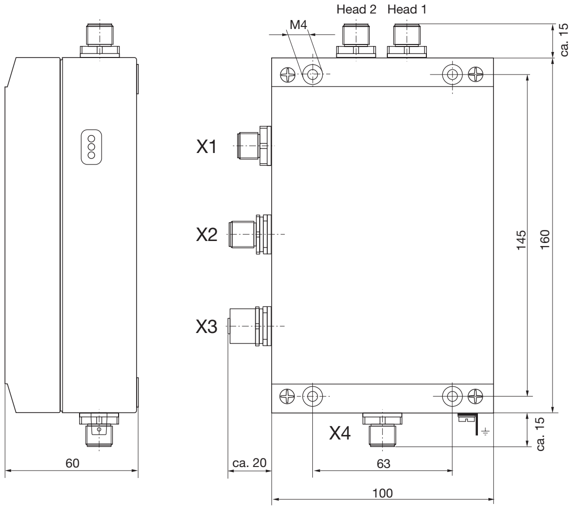
Accessory
(optional,
not included)

Type		**Ordering code**
Connector	for X1	BKS-S 79-00
	for X2	BKS-S103-00
	for X3	BKS-S105-00
Termination	for X3	BKS-S105-R01
Protective cap	for X3	BKS 12-CS-00
Protective cap	for Head 1, Head 2	BES 12-SM-2
Connector	for Head 1, Head 2 no cable	BKS-S117-00
Connection cable	for Head 1, Head 2; 5 m	BIS-L-500-PU-05
Connection cable	for Head 1, Head 2; 10 m	BIS-L-500-PU-10
Connection cable	for Head 1, Head 2; 25 m one end with molded-in connector, one end for user-assembled connector, length as desired	BIS-L-501-PU-25

BIS L-6022
Mounting the Processor

Mounting the BIS L-6022 processor

The processor is mounted using 4 M4 screws.



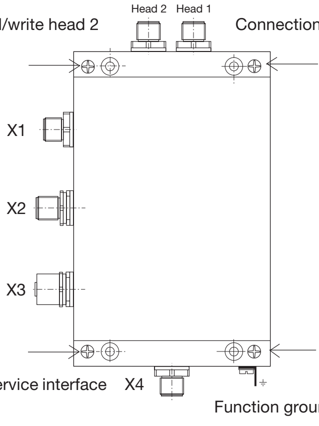
BIS L-6022
Opening the Processor / Interface Information

Opening the BIS L-6022 processor

To set the PROFIBUS-DP address, activate or deactivate, or to change the EEPROM, you must open up the BIS L-6022 processor.

Remove the 4 screws on the BIS L-6022 and lift off the cover. See the following 🕮 for additional information.

BIS L-6022 interfaces



Connection for read/write head 2

Connection for read/write head 1

Supply voltage
digital input

PROFIBUS-DP
Input

PROFIBUS-DP
Output

Service interface

Function ground FE

Be sure before opening that the unit is disconnected from power. ⚠

Mounting of the cover (4 screws), max. permissible tightening torque: 0.15 Nm

Connection locations and names

BIS L-6022
Interface Information / Wiring Diagrams

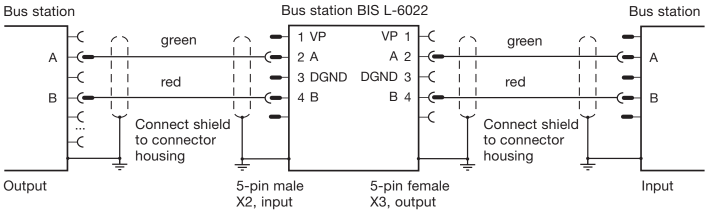
Wiring

To insert BIS L-6022 processor into the serial PROFIBUS and to connect the supply voltage and the digital input, the cables have to be connected to the terminals of the processor. The read/write heads have to be connected to the terminals Head 1 and Head 2.

PROFIBUS-DP

Ensure that the device is turned off.

To insert BIS L-6022 processor into the serial PROFIBUS-DP, there are the terminal X2 for the PROFIBUS input and the terminal X3 for the PROFIBUS output.



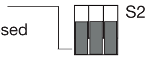
BIS L-6022
Interface Information / Wiring Diagrams

PROFIBUS-DP Terminating resistor

The last bus module must terminate the bus with a resistor. In the case of the BIS L-6022, this can be realized in two different ways:

1. **In the device** by closing the switch S2 (factory standard is open)

   Note: Output terminal must be closed off with a screw cover in order to maintain the enclosure rating.



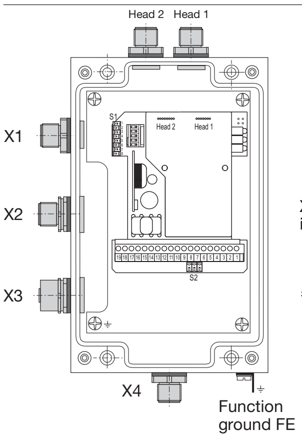
S2	Terminating resistor
closed	active
open	passive

2. **Outside the device** in a connector to socket X3. In this case the signal VP (pin 1) and DGND (pin 3) should be brought out in order to connect the external resistor to the potential.

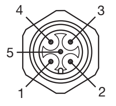
   Note: In this case S2 has to be open!

BIS L-6022
Interface Information / Wiring Diagrams

Wiring diagram for BIS L-6022 processor

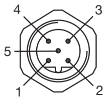


X1, supply voltage, digital input

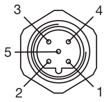


Pin	Function
1	+Vs
2	–IN
3	–Vs
4	+IN
5	n.c.

X2, PROFIBUS-input (male)

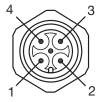


X3, PROFIBUS-output (female)



Pin	Function
1	VP
2	A
3	DGND
4	B
5	n.c.

X4, Service interface



Pin	Function
1	n.c.
2	TxD
3	GND
4	RxD

n.c. = do not connect!

The function-ground connector FE should be connected to earth directly or through a RC combination depending on the system (potential counterpoise).

When connecting the bus leads, make sure that the shield has proper connection to connector housing.

BIS L-6022
Changing the EEPROM

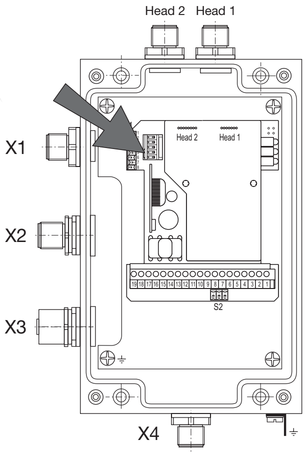
Changing the EEPROM in the BIS L-6022 processor

To replace the EEPROM, open up the processor as described on ☐ 64.

Be sure before opening that the unit is disconnected from power.

To avoid damaging the EEPROM, please observe the requirements for handling electrostatically sensitive components.

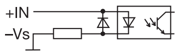
The EEPROM is replaced by unplugging and plugging back into the socket.



Location of the EEPROM

BIS L-6022
Technical Data

Dimensions, Weight	**Housing**	Metal
	Dimensions	190 x 120 x 60 mm
	Weight	820 g
Operating Conditions	Ambient temperature	0 °C to +60 °C
Enclosure	Protection class	IP 65 (when connected)
Connections	Integral connector X1 for **V_S, IN**	5-pin (male)
	Integral connector X2 for **PROFIBUS-DP** input	5-pin (male)
	Integral connector X3 for **PROFIBUS-DP** output	5-pin (female)
	Integral connector X4 for **Service interface**	4-pin (male)
Electrical Connections	**Supply voltage V_s**	DC 24 V ± 20 % LPS Class 2
	Ripple	≤ 10 %
	Current draw	≤ 400 mA
	Digital input +IN	Optocoupler isolated
	Control voltage active	4 V to 40 V
	Control voltage inactive	1.5 V to –40 V
	Input current at 24 V	11 mA
	Delay time, typ.	5 ms
	PROFIBUS-DP, Connector X2, X3	serial interface for PROFIBUS stations
	Head 1, Head 2, Read/Write Head	via 2 x connectors 8-pin connector (female) for all read/write heads BIS L-3_ _ with 8-pin connector (male)
	Service interface X4	RS 232



BIS L-6022
Technical Data

Function displays	BIS operating messages:	
	Ready / Bus active	LED red / green
	CT1 Present / operating	LED green / yellow
	CT2 Present / operating	LED green / yellow

c UL us LISTED

Process Control Equipment

Control No 3TLJ

File No E227256

CE Declaration of Conformity and user safety

CE *This product was developed and produced considering the claimed European standards and guidelines.*

☞ You can separately request a Declaration of Conformity.

Further safety measures you can find in chapter *Safety* (see ⎕ 4).

BIS L-6022
Ordering Information

Ordering code

BIS L-6022-019-050-03-ST14

Balluff Identification System

Type L Read/Write System

Hardware-Type
6022 = metal housing, PROFIBUS-DP

Software-Type
019 = PROFIBUS-DP

Adapter
050 = with two connectors for read/write heads BIS L-3_ _

Interface
03 = BUS versions

User Connection
ST14 = Connector version X1, X2, X3, X4 (male: 2 × 5-pin, 1 × 4-pin, female: 1 × 5-pin)

BIS L-6022
Ordering Information

Accessory
(optional,
not included)

Type		**Ordering code**
Mating connector	for X1	BKS-S 79-00
	for X2	BKS-S103-00
	for X3	BKS-S105-00
	for X4	BKS-S 10-3
Termination	for X3	BKS-S105-R01
Protective cap	for X3	BKS 12-CS-00
Protective cap	for Head _, X4	BES 12-SM-2
Connector	for Head 1, Head 2 no cable	BKS-S117-00
Connection cable	for Head 1, Head 2; 5 m	BIS-L-500-PU-05
Connection cable	for Head 1, Head 2; 10 m	BIS-L-500-PU-10
Connection cable	for Head 1, Head 2; 25 m one end with molded-in connector, one end for user-assembled connector, length as desired	BIS-L-501-PU-25

Symbols / Abbreviations

⎓	DC Current
LPS	Limited Power Source Class 2
⏚	Function ground
	ESD Symbol

Appendix, ASCII Table

Deci-mal	Hex	Control Code	ASCII	Deci-mal	Hex	Control Code	ASCII	Deci-mal	Hex	ASCII	Deci-mal	Hex	ASCII	Deci-mal	Hex	ASCII	Deci-mal	Hex	ASCII
0	00	Ctrl @	NUL	22	16	Ctrl V	SYN	44	2C	,	65	41	A	86	56	V	107	6B	k
1	01	Ctrl A	SOH	23	17	Ctrl W	ETB	45	2D	-	66	42	B	87	57	W	108	6C	l
2	02	Ctrl B	STX	24	18	Ctrl X	CAN	46	2E	.	67	43	C	88	58	X	109	6D	m
3	03	Ctrl C	ETX	25	19	Ctrl Y	EM	47	2F	/	68	44	D	89	59	Y	110	6E	n
4	04	Ctrl D	EOT	26	1A	Ctrl Z	SUB	48	30	0	69	45	E	90	5A	Z	111	6F	o
5	05	Ctrl E	ENQ	27	1B	Ctrl [	ESC	49	31	1	70	46	F	91	5B	[	112	70	p
6	06	Ctrl F	ACK	28	1C	Ctrl \	FS	50	32	2	71	47	G	92	5C	\	113	71	q
7	07	Ctrl G	BEL	29	1D	Ctrl]	GS	51	33	3	72	48	H	93	5D	]	114	72	r
8	08	Ctrl H	BS	30	1E	Ctrl ^	RS	52	34	4	73	49	I	94	5E	^	115	73	s
9	09	Ctrl I	HT	31	1F	Ctrl _	US	53	35	5	74	4A	J	95	5F	_	116	74	t
10	0A	Ctrl J	LF	32	20		SP	54	36	6	75	4B	K	96	60	`	117	75	u
11	0B	Ctrl K	VT	33	21		!	55	37	7	76	4C	L	97	61	a	118	76	v
12	0C	Ctrl L	FF	34	22		"	56	38	8	77	4D	M	98	62	b	119	77	w
13	0D	Ctrl M	CR	35	23		#	57	39	9	78	4E	N	99	63	c	120	78	x
14	0E	Ctrl N	SO	36	24		$	58	3A	:	79	4F	O	100	64	d	121	79	y
15	0F	Ctrl O	SI	37	25		%	59	3B	;	80	50	P	101	65	e	122	7A	z
16	10	Ctrl P	DLE	38	26		&	60	3C	<	81	51	Q	102	66	f	123	7B	{
17	11	Ctrl Q	DC1	39	27		'	61	3D	=	82	52	R	103	67	g	124	7C	|
18	12	Ctrl R	DC2	40	28		(	62	3E	>	83	53	S	104	68	h	125	7D	}
19	13	Ctrl S	DC3	41	29		)	63	3F	?	84	54	T	105	69	i	126	7E	~
20	14	Ctrl T	DC4	42	2A		*	64	40	@	85	55	U	106	6A	j	127	7F	DEL
21	15	Ctrl U	NAK	43	2B		+												