BECKHOFF New Automation Technology

Documentation | EN

ILxxxx-B520

Fieldbus Box modules for DeviceNet

Inhaltsverzeichnis

BECKHOFF

1 Foreword

1.1 Notes on the documentation

Intended audience

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning these components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.

We reserve the right to revise and change the documentation at any time and without prior announcement.

No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH. Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents: EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702 with corresponding applications or registrations in various other countries.

Copyright

1.2 Safety instructions

Safety regulations

Please note the following safety instructions and explanations!
Product-specific safety instructions can be found on following pages or in the areas mounting, wiring, commissioning etc.

Exclusion of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation and drive engineering who are familiar with the applicable national standards.

Description of instructions

In this documentation the following instructions are used.
These instructions must be read carefully and followed without fail!

⚠ DANGER
Serious risk of injury!
Failure to follow this safety instruction directly endangers the life and health of persons.

⚠ WARNING
Risk of injury!
Failure to follow this safety instruction endangers the life and health of persons.

⚠ CAUTION
Personal injuries!
Failure to follow this safety instruction can lead to injuries to persons.

NOTE
Damage to environment/equipment or data loss
Failure to follow this instruction can lead to environmental damage, equipment damage or data loss.

● ℹ **Tip or pointer**
This symbol indicates information that contributes to better understanding.

1.3   Documentation Issue Status

Version	Comment
1.3	• Structure update
1.0 … 1.3	Previous versions

Firmware and hardware versions

This documentation refers to the firmware and hardware version that was applicable at the time the documentation was written.

The module features are continuously improved and developed further. Modules having earlier production statuses cannot have the same properties as modules with the latest status. However, existing properties are retained and are not changed, so that older modules can always be replaced with new ones.

2   Introduction

The purpose of this document is to provide DeviceNet Specific Information needed to run Beckhoff ILxxxx-B520 fieldbus coupler boxes in a DeviceNet Network.

DeviceNet Characteristics

Characteristic	Description
DeviceNet Functionality	Group Two Only Slave for DeviceNet Master / Scanner
DeviceType	Communications Adapter
IO – Modes	Polling, Bit Strobe, Change of State / Cyclic
IO – Data Length	The IO Data length for the IO-Modes Polling and COS/Cyclic is limited to 512 Bytes in each direction
Configuration	Switches (node address) , Configuration Objects, Electronic Data Sheet (EDS)
LEDs	Module / Network Status LED, Vendor Specific IO LEDs
Electronic Data Sheet	Electronic Data Sheet for each type of ILxxxx-B520 (www.beckhoff.com)
Connector	Sealed Micro Style Connector
Baud Rates	125 Kbaud, 250 Kbaud, 500 Kbaud, Auto Baud Detection

3   IO Data Mapping

IO Data Mapping

The IO Data Mapping describes the contents of the IO Data of the ILxxxx-B520 in Receive and Transmit direction. The description is done by DeviceNet Assembly Objects. Each of the ILxxxx-B520 supports Assembly Objects in each data direction. The input data/status or output data is mapped to a byte stream exchanged with the DeviceNet Master / Scanner by IO-Data transfer.

3.1   IO Data Assignment of IL230x and Extension Boxes

**Digital Signals
(bit-oriented)**

The digital Signals are bit-oriented. This means that one bit in the process image is assigned to each channel. The ILxxxx-B520 creates a memory area containing the current input bits and ensures that the bits in a second memory area dedicated to the output channels are written out immediately.

**Analog Signals
(byte-oriented)**

The processing of analog signals is always byte-oriented. Analog input and output values are represented in memory by two bytes each. Values are represented in unsigned/signed integer or two's complement format (see IExxxx manual). The number "0" stands for the input/output value "0 V", "0 mA" or "4 mA". Per default, Control and Status Bytes for each analog channel are **not** mapped to the IO-data image. An analog channel is represented in the process image by two bytes.

**Special Signals and interfaces
(byte-oriented)**

The ILxxxx-B520 supports extension boxes with other interfaces such as RS232, RS485 incremental encoder, SSI Sensor interface. These signals can considered similarly to the analog signals named above. For some special signals the width of 16 Bit is not sufficient. The IL230x can Support any byte width.

**Assignment of inputs/ outputs to the process image
(Default Assignment / Assembly Object)**

Once it has been switched on, the ILxxxx-B520 finds out how many Extension Boxes are connected and creates an assignment list. The analog and digital channels, divided into inputs and outputs, are assembled into different parts of the list. The assignment starts on the first extension box next to the ILxxxx-B520. The software in the ILxxxx-B520 collects the individual entries for each of the channels in order to create the assignment list.

**IO Data from Master / Scanner to ILxxxx-B520
(Default Assignment / Assembly Object)**

IO-Data which is transferred from the DeviceNet Master / Scanner to the ILxxxx-B520 begins with byte-oriented values which is the data for the analog output and special signal extension boxes. The bit-oriented data for the digital outputs of the ILxxxx-B520 and the digital output extension boxes is transmitted after the byte-oriented data. If the total number of digital inputs is not a multiple of 8, there will be a number of bits left over in the last data byte. These will be discarded.

**IO Data from ILxxxx-B520 to Master / Scanner to
(Default Assignment / Assembly Object)**

IO-Data which is transferred from the ILxxxx-B520 to the DeviceNet Master / Scanner begins with byte-oriented values which is the data from the analog input and special signal extension boxes. The bit-oriented data for the digital inputs of the ILxxxx-B520 and the digital input extension boxes is transmitted after the byte-oriented data. If the total number of digital outputs is not a multiple of 8, there will be a number of bits left over in the last data byte. These will be discarded.

Status Byte at the end of the input data

An extra status byte is transferred at the end of the Input Data and returns the status of the ILxxxx-B520 with the following meaning:

- Bit0: IL_Error : IO Error, internal Data exchange ILxxxx-B520 has failed
- Bit1: IL-Cfg : ILxxxx-B520 Configuration Error
- Bit2: reserved
- Bit3: Diag : Diagnosis of analog Channel
- Bit4: reserved
- Bit5: reserved
- Bit6: reserved
- Bit7: FB_Error : Fieldbus Error / Idle Mode

The status byte corresponds to the Attribute "IL-Status" of the IL-Config Object.

IO Data from Master / Scanner to ILxxxx-B520   IO Data from ILxxxx-B520 to Master / Scanner
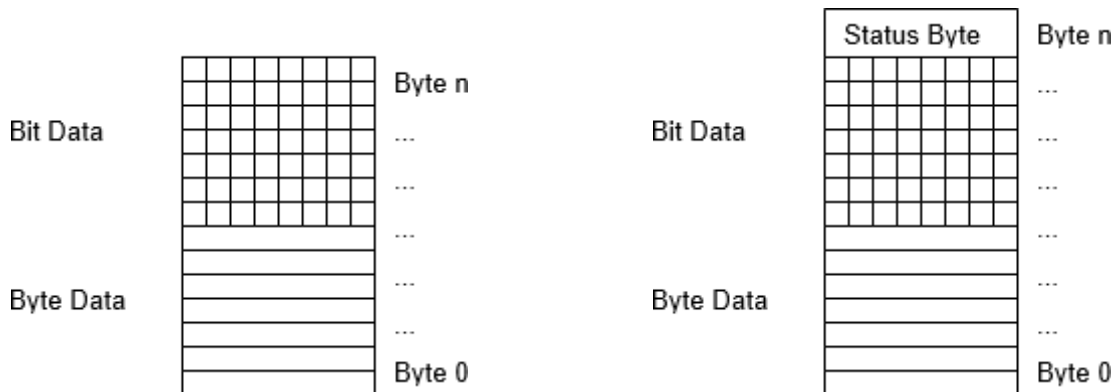
3.1.1   IL230x-B520

Description	4 x Digital Input + 4 x Digital Output, 24 V_{DC}
IO Data Type	Digital Signal, bit-oriented
Input Data	4 Bit
Output Data	4 Bit

3.1.2   IE100x

Description	8 Channel Digital Input, 24 V_{DC}
IO Data Type	Digital Signal, bit-oriented
Input Data	8 Bit
Output Data	none

3.1.3   IE1502

Description	2 Channel Up/Down Counter, 24 V_{DC}, 100 kHz
IO Data Type	Analog Signal, byte-oriented
Input Data	10 Byte • Byte 0: Status, Channel 1 • Byte 1: LowByte DataIn[0], Channel 1 • Byte 2: HighByte DataIn[0], Channel 1 • Byte 3: LowByte DataIn[1], Channel 1 • Byte 4: HighByte DataIn[1], Channel 1 • Byte 5: Status, Channel 2 • Byte 6: LowByte DataIn[0], Channel 2 • Byte 7: HighByte DataIn[0], Channel 2 • Byte 8: LowByte DataIn[1], Channel 2 • Byte 9: HighByte DataIn[1], Channel 2
Output Data	10 Byte • Byte 0: Control, Channel 1 • Byte 1: LowByte DataOut[0], Channel 1 • Byte 2: HighByte DataOut [0], Channel 1 • Byte 3: LowByte DataOut [1], Channel 1 • Byte 4: HighByte DataOut [1], Channel 1 • Byte 5: Control, Channel 2 • Byte 6: LowByte DataOut [0], Channel 2 • Byte 7: HighByte DataOut [0], Channel 2 • Byte 8: LowByte DataOut [1], Channel 2 • Byte 9: HighByte DataOut [1], Channel 2

(detailed description, see IE1502 manual)

3.1.4   IE200x

Description	8 Channel Digital Output, 24 V_{DC}
IO Data Type	Digital Signal, bit-oriented
Input Data	none
Output Data	8 Bit

3.1.5   IE230x

Description	4 x Digital Input + 4 x Digital Output, 24 V_{DC}
IO Data Type	Digital Signal, bit-oriented
Input Data	4 Bit
Output Data	4 Bit

3.1.6   IE240x

Description	8 Channel Digital Combi Input / Output, 24 V_{DC}
IO Data Type	Digital Signal, bit-oriented
Input Data	8 Bit
Output Data	8 Bit

3.1.7   IE3102

Description	4 Channel Analog Input, ± 10 V
IO Data Type	Analog Signal, byte-oriented
Input Data	8 Byte • Byte 0: LowByte Channel 1 • Byte 1: HighByte Channel 1 • Byte 2: LowByte Channel 2 • Byte 3: HighByte Channel 2 • Byte 4: LowByte Channel 3 • Byte 5: HighByte Channel 3 • Byte 6: LowByte Channel 4 • Byte 7: HighByte Channel 4
Output Data	none

(detailed description, see IE3102 manual)

3.1.8   IE3112

Description	4 Channel Analog Input 0-20 mA
IO Data Type	Analog Signal, byte-oriented
Input Data	8 Byte • Byte 0: LowByte Channel 1 • Byte 1: HighByte Channel 1 • Byte 2: LowByte Channel 2 • Byte 3: HighByte Channel 2 • Byte 4: LowByte Channel 3 • Byte 5: HighByte Channel 3 • Byte 6: LowByte Channel 4 • Byte 7: HighByte Channel 4
Output Data	none

(detailed description, see IE3112 manual)

3.1.9   IE3202

Description	4 Channel Analog Input Pt100 (RTD)
IO Data Type	Analog Signal, byte-oriented
Input Data	8 Byte • Byte 0: LowByte Channel 1 • Byte 1: HighByte Channel 1 • Byte 2: LowByte Channel 2 • Byte 3: HighByte Channel 2 • Byte 4: LowByte Channel 3 • Byte 5: HighByte Channel 3 • Byte 6: LowByte Channel 4 • Byte 7: HighByte Channel 4
Output Data	none

(detailed description, see IE3202 manual)

3.1.10   IE3312

Description	4 Channel Analog Input thermocouple
IO Data Type	Analog Signal, byte-oriented
Input Data	8 Byte • Byte 0: LowByte Channel 1 • Byte 1: HighByte Channel 1 • Byte 2: LowByte Channel 2 • Byte 3: HighByte Channel 2 • Byte 4: LowByte Channel 3 • Byte 5: HighByte Channel 3 • Byte 6: LowByte Channel 4 • Byte 7: HighByte Channel 4
Output Data	none

(detailed description, see IE3312 manual)

3.1.11   IE4112

Description	4 Channel Analog Output, 0 … 20 mA
IO Data Type	Analog Signal, byte-oriented
Input Data	none
Output Data	8 Byte • Byte 0: LowByte Channel 1 • Byte 1: HighByte Channel 1 • Byte 2: LowByte Channel 2 • Byte 3: HighByte Channel 2 • Byte 4: LowByte Channel 3 • Byte 5: HighByte Channel 3 • Byte 6: LowByte Channel 4 • Byte 7: HighByte Channel 4

(detailed description, see IE4112 manual)

3.1.12   IE4132

Description	4 Channel Analog Output, -10 V / 0 … 10 V
IO Data Type	Analog Signal, byte-oriented
Input Data	none
Output Data	8 Byte • Byte 0: LowByte Channel 1 • Byte 1: HighByte Channel 1 • Byte 2: LowByte Channel 2 • Byte 3: HighByte Channel 2 • Byte 4: LowByte Channel 3 • Byte 5: HighByte Channel 3 • Byte 6: LowByte Channel 4 • Byte 7: HighByte Channel 4

(detailed description, see IE4132 manual)

3.1.13   IE5009

Description	1 Channel SSI Sensor Interface
IO Data Type	Analog Signal, byte-oriented
Input Data	4 Byte • Byte 0: LowByte PCDL • Byte 1: HighByte PCDL • Byte 2: LowByte PCDH • Byte 3: HighByte PCDH
Output Data	none

(detailed description, see IE5009 manual)

3.1.14 IE5109

Description	1 Channel Incremental Encoder Interface, 1 MHz
IO Data Type	Analog Signal, byte-oriented
Input Data	6 Byte • Byte 0: Status • Byte 1: Low Byte Counter • Byte 2: High Byte Counter • Byte 3: Latch • Byte 4: LowByte Period • Byte 5: HighByte Period
Output Data	3 Byte • Byte 0: Control • Byte 1: Low Byte RegData • Byte 2: High Byte RegData

(detailed description, see IE5109 manual)

3.1.15 IE6002

Description	1 Channel Serial Interface, RS232 C
IO Data Type	Serial Signal, byte-oriented
Input Data	6 Byte • Byte 0: Status • Byte 1: Data In 0 • Byte 2: Data In 1 • Byte 3: Data In 2 • Byte 4: Data In 3 • Byte 5: Data In 4
Output Data	6 Byte • Byte 0: Control • Byte 1: Data In 0 • Byte 2: Data In 1 • Byte 3: Data In 2 • Byte 4: Data In 3 • Byte 5: Data In 4

(detailed description, see IE6002 manual)

BECKHOFF

3.1.16   IE6012

Description	1 Channel Serial Interface, 0 … 20 mA (TTY)
IO Data Type	Serial Signal, byte-oriented
Input Data	6 Byte • Byte 0: Status • Byte 1: Data In 0 • Byte 2: Data In 1 • Byte 3: Data In 2 • Byte 4: Data In 3 • Byte 5: Data In 4
Output Data	6 Byte • Byte 0: Control • Byte 1: Data In 0 • Byte 2: Data In 1 • Byte 3: Data In 2 • Byte 4: Data In 3 • Byte 5: Data In 4

(detailed description, see IE6012 manual)

3.1.17   IE6022

Description	1 Channel Serial Interface, RS422 / RS485
IO Data Type	Serial Signal, byte-oriented
Input Data	6 Byte • Byte 0: Status • Byte 1: Data In 0 • Byte 2: Data In 1 • Byte 3: Data In 2 • Byte 4: Data In 3 • Byte 5: Data In 4
Output Data	6 Byte • Byte 0: Control • Byte 1: Data In 0 • Byte 2: Data In 1 • Byte 3: Data In 2 • Byte 4: Data In 3 • Byte 5: Data In 4

(detailed description, see IE6022 manual)

3.2 IO Mapping Example

The example below shows the IO-Data mapping with default IO-Assignment:

Configuration

- IL2301 +
- IE1001 +
- IE2001 +
- IE2401 +
- IE3102 +
- IE4112

Input Data

12 Byte

- Byte 0: IE3102, Low Byte Channel 1
- Byte 1: IE3102, High Byte Channel 1
- Byte 2: IE3102, Low Byte Channel 2
- Byte 3: IE3102, High Byte Channel 2
- Byte 4: IE3102, Low Byte Channel 3
- Byte 5: IE3102, High Byte Channel 3
- Byte 6: IE3102, Low Byte Channel 4
- Byte 7: IE3102, High Byte Channel 4
- Byte 8:
  - Bit 0: IL2301, Bit0
  - Bit 1: IL2301, Bit1
  - Bit 2: IL2301, Bit2
  - Bit 3: IL2301, Bit3
  - Bit 4: IE1001, Bit0
  - Bit 5: IE1001, Bit1
  - Bit 6: IE1001, Bit2
  - Bit 7: IE1001, Bit3
- Byte 9:
  - Bit 0: IE1001, Bit0
  - Bit 1: IE1001, Bit1
  - Bit 2: IE1001, Bit2
  - Bit 3: IE1001, Bit3
  - Bit 4: IE2401, Bit0
  - Bit 5: IE2401, Bit1
  - Bit 6: IE2401, Bit2
  - Bit 7: IE2401, Bit3
- Byte 10:
  - Bit 0: IE2401, Bit4
  - Bit 1: IE2401, Bit5
  - Bit 2: IE2401, Bit6
  - Bit 3: IE2401, Bit7
  - Bit 4: not used

- ○ Bit 5: not used
- ○ Bit 6: not used
- ○ Bit 7: not used
- Byte 11: IL-Status Byte
  - ○ Bit0: IL_Error
  - ○ Bit1: IL-Cfg
  - ○ Bit2: reserved
  - ○ Bit3: Diag
  - ○ Bit4: reserved
  - ○ Bit5: reserved
  - ○ Bit6: reserved
  - ○ Bit7: FB_Error

Output Data

11 Byte

- Byte 0: IE4112, Low Byte Channel 1
- Byte 1: IE4112, High Byte Channel 1
- Byte 2: IE4112, Low Byte Channel 2
- Byte 3: IE4112, High Byte Channel 2
- Byte 4: IE4112, Low Byte Channel 3
- Byte 5: IE4112, High Byte Channel 3
- Byte 6: IE4112, Low Byte Channel 4
- Byte 7: IE4112, High Byte Channel 4
- Byte 8:
  - ○ Bit 0: IL2301, Bit0
  - ○ Bit 1: IL2301, Bit1
  - ○ Bit 2: IL2301, Bit2
  - ○ Bit 3: IL2301, Bit3
  - ○ Bit 4: IE2001, Bit0
  - ○ Bit 5: IE2001, Bit1
  - ○ Bit 6: IE2001, Bit2
  - ○ Bit 7: IE2001, Bit3
- Byte 9:
  - ○ Bit 0: IE2001, Bit4
  - ○ Bit 1: IE2001, Bit5
  - ○ Bit 2: IE2001, Bit6
  - ○ Bit 3: IE2001, Bit7
  - ○ Bit 4: IE2401, Bit0
  - ○ Bit 5: IE2401, Bit1
  - ○ Bit 6: IE2401, Bit2
  - ○ Bit 7: IE2401, Bit3
- Byte 10:
  - ○ Bit 0: IE2401, Bit4
  - ○ Bit 1: IE2401, Bit5
  - ○ Bit 2: IE2401, Bit6
  - ○ Bit 3: IE2401, Bit7

- Bit 4: not used
- Bit 5: not used
- Bit 6: not used
- Bit 7: not used

3.3   Assembly Objects

The Assembly Object binds attributes of multiple objects, which allows data to or from each object to be sent or received over a single connection (IO or Explicit). Assembly objects are used to bind input data and output data.

Class Code: 4 (04$_{hex}$)

Class Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
1	Get	Revision	UINT	Revision of implementation	2

Instance Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
3 (03$_{hex}$)	Get/set	Value	Array of Byte	Input or Output Data of the ILxxxx-B520.	

Common Services

Service Code	Service Name	Description
14 (0E$_{hex}$)	Get_Attribute_Single	Returns the contents of the specified attribute
16 (10$_{hex}$)	Set_Attribute_Single	Modifies an attribute Value

IO Assembly Instances

Number	Type	Name
101	Output	Analog and Digital Outputs
102	Output	Digital Outputs
103	Output	Analog Outputs
111	Input	Analog and Digital Inputs and Status Byte
112	Input	Digital Inputs and Status Byte
113	Input	Analog Inputs and Status Byte

3.3.1   Input Assembly

Default Instance

Number	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
111	0 to (n-1)	Analog Inputs							
	n to (m-1)	Digital Inputs							
	m	IL-Status Byte							

Number	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
112	0 to (n-1)	Digital Inputs							
	n	IL-Status Byte							

Number	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
113	0 to (n-1)	Analog Inputs							
	n	IL-Status Byte							

3.3.2   Output Assembly

Default Instance

Number	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
101	0 to (n-1)	Analog Outputs							
	n to m	Digital Outputs							

Number	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
102	n	Digital Outputs							

Number	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
103	n	Analog Outputs							

3.3.3   Mapping IO Assembly Data to DeviceNet Objects

Data Component Name	Class name	Class number	Instance	Attribute name	Attribute number
Discrete Input n	Digital Input Channel	101	n	Value	1
Discrete Output n	Digital Output Channel	102	n	Value	1
Analog Input n	Analog Input Channel	103	n	Value	1
Analog Output n	Analog Output Channel	104	n	Value	1
IL-Status Byte	IL Config Object	100	1	IL-Status	5

4   IL Config Object

The ILxxxx-B520 provides the vendor specific object class to access its status, diagnostic and configuration data. Within the IL Config Object the full range of Registers and Status-Information of the ILxxxx-B520 and the connected extension boxes is available.

The IL Config Object Class is Vendor specific and within in the Range of the Vendor specific Class Codes.

Class Code: 100 (64_{hex})

Class Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
1	Get	Revision	UINT	Revision of implementation	1
2	Get	Max. Instance	UINT	Max. number of instances	1

Instance Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
1	Get/Set	Terminal Number	USINT	Number of terminal	0: Coupler >0: Channels
2	Get/Set	Table Number	USINT	Number of table	See ILxxxx-B520 / Channel Description
3	Get/Set	Register Number	USINT	Number of Register	See ILxxxx-B520 / Channel Description
4	Get/Set	Register Data	DWORD	Register Value	see semantics
5	Get	IL-Status	BYTE	Status of the ILxxxx-B520	see semantics
6	Get	Extension Box-Diagnosis	WORD	Diagnosis of the extension boxes	see semantics
10	Get/Set	IO Error Action	BYTE	Action to be performed if a fieldbus error occurs	see semantics
11	Get/Set	Poll produced data type	BYTE	Type of IO data produced via the Poll mode	see semantics
12	Get/Set	COS/ Cyclic produced data type	BYTE	Type of IO data produced via the Change of State / Cyclic mode	see semantics
13	Get/Set	Bit Strobe produced data type	BYTE	Type of IO data produced via the Bit Strobe / Cyclic mode	see semantics
14	Get/Set	Poll / COS/ Cyclic consumed data type	BYTE	Type of IO data consumed via the Poll / Change of State / Cyclic mode	see semantics

Instance Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
15	Get	Input Size Poll Mode	BYTE	Number of Bytes produced via the Poll mode	
16	Get	Input Size Bit Strobe Mode	BYTE	Number of Bytes produced via the Bit Strobe mode	
17	Get	Input Size COS / Cyclic Mode	BYTE	Number of Bytes produced via the Change of State / Cyclic mode	
18	Get	Output Size Poll / COS/ Cyclic Mode	BYTE	Number of Bytes consumed via the Poll / Change of State /Cyclic mode	
19	Get/Set	Device Diagnostics	BYTE	Enables Diagnosis of complex (analog/special-function) extension boxes	see semantics
20	Get	Analog Out Length	BYTE	Size of analog output data in bits	
21	Get	Analog In Length	BYTE	Size of analog input data in bits	
22	Get	Digital In Length	BYTE	Size of digital output data in bits	
23	Get	Digital Out Length	BYTE	Size of digital input data in Bits	
25	Get/Set	Bus Off Behavior	BYTE	Behavior of ILxxxx-B520 after detection of a Bus-Off event	see semantics

4.1 Semantics

Register Data

Within the response of a Get_Attribute_Single Service to the „Register Data" Attribute the status of the internal reading and the registers data is returned by the ILxxxx-B520. The meaning of the registers data is described in the ILxxxx-B520 manual.

Within the request of a Set_Attribute_Single Service to the „Register Data" Attribute the Low-Word of the attribute „RegisterData" is used to send the Register data to the ILxxxx. The meaning of the registers data is described in the ILxxxx-B520 manual.

Response Data of Get_Attribute_Single

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Status (Low Byte)							
1	Status (High Byte): 0 = OK, >0 = Error							
2	Register data (Low Byte)							
3	Register data (High Byte)							

Request Data of Set_Attribute_Single

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Register data (Low Byte)							
1	Register data (High Byte)							
2	Not used							
3	Not used							

Response Data of Set_Attribute_Single

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Status (Low Byte)							
1	Status (High Byte) : 0 = OK, >0 = Error							

IL_Status

The „IL Status" attribute shows the actual status of the ILxxxx-B520.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte	FB_Error	res.	res.	res.	Diag	res.	IL_Cfg	IL_Error

IL_Error : IO Error, internal Data exchange ILxxxx-B520 has failed

IL_Cfg : ILxxxx-B520 Configuration Error (EEPROM check failed)

Diag : Diagnosis of analog Channel

FB_Error : Fieldbus Error / Idle Mode

Extension Box Diagnosis

The „Extension Box Status" attribute describes which of the extension boxes has encountered a diagnosis event.

> ℹ After reading the extension box Status, the attribute is cleared until the next diagnosis appears. Reading of the attribute also clears the "Diag-Bit" within the attribute "IL Status". The function of this attribute is only active if the diagnosis of analog extension boxes is enabled.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Extension Box Number							
1	Status	Error Code					Channel Number	

Extension Box Number : Number of failed extension box

Channel Number : Number of failed channel of the extension box

Error Code : Extension box specific Error Code (see extension box manual)

Status : 0 = Error is reseted

1 = Error occurred

IO Error Action

Action to be performed if a fieldbus error occurred.

> ℹ To activate a new IO error Action setting a device reset has to be performed either by executing a power cycle to the ILxxxx-B520 or by executing a Reset Service (Service Code 5) to the ILxxxx-B520 Identity Object (Class Id 1, Instance 1)

Value	Description
0	Leave local IO Cycle
1	Leave local IO Cycle and reset outputs (default)
2	freeze outputs

Poll produced data type
COS / Cyclic produced data type
Poll / COS / Cylic consumed data type

Type of IO data produced/consumed via the Poll mode and Change of State / Cyclic mode.

Value	Description
0	Analog and digital IO Data with Status Byte (default)
1	Digital IO Data with Status Byte
2	Analog IO Data with Status Byte

Bit Strobe produced data type

Type of IO data produced via the Bit Strobe mode.

Value	Description
0	Diagnostic Data (see Attribute extension box diagnosis)
1	Digital Inputs with Status Byte (default)

Device Diagnostics

Enables Diagnostics of complex (analog/special function) extension boxes. Setting this attribute enables the diagnosis of complex channels which is needed to read the extension box diagnosis.

> **i** To activate the diagnosis functions a device reset has to be performed either by executing a power cycle to the ILxxxx-B520 or executing a Reset Service (Service Code 5)to the Identity Object (Class Id 1, Instance 1).

Value	Description
0	Device Diagnostic OFF (default)
1	Device Diagnostic ON

Bus Off Behavior

Defines the Behavior of the ILxxxx-B520 after detection of a Bus-Off event.

This attribute is available from Firmware Revision 1.2 upwards.

Value	Description
0	Hold the ILxxxx-B520 in Bus-Off State (default)
1	Reset ILxxxx-B520 and restart communication. After detection of 255 Bus-Off events the ILxxxx-B520 keeps in the reset state.

5 Digital Input Channel Object

The ILxxxx-B520 provides Digital Input Channel Objects to access the digital input channels data of the ILxxxx-B520 and the connected extension boxes. For each digital input channel exits one instance of the Digital Input Channel Object Class.

The Digital Input Channel Object Class is Vendor specific and within in the range of the Vendor specific Class Codes.

Class Code: 101 (65_{hex})

Class Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
1	Get	Revision	UINT	Revision of implementation	1
2	Get	Max. Instance	UINT	Max. number of instances	

Instance Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
1 (01_{hex})	Get	Value	BOOL	Input Channel Value	0: OFF 1: ON

Common Services

Service Code	Service Name	Description
14 ($0E_{hex}$)	Get_Attribute_Single	Returns the contents of the specified attribute

6 Digital Output Channel Object

The ILxxxx-B520 provides Digital Output Channel Objects to access the digital output channels data of the ILxxxx-B520 and the connected extension boxes. For each digital output channel exits one instance of the Digital Output Channel Object Class.

The Digital Output Channel Object Class is Vendor specific and within in the range of the Vendor specific Class Codes.

Class Code: 102 (66_{hex})

Class Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
1	get	Revision	UINT	Revision of implementation	1
2	get	Max. Instance	UINT	Max. number of instances	

Instance Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
1 (01_{hex})	Get/Set	Value	BOOL	Output Channel Value	0: OFF 1: ON

Common Services

Service Code	Service Name	Description
14 ($0E_{hex}$)	Get_Attribute_Single	Returns the contents of the specified attribute
16 (10_{hex})	Set_Attribute_Single	Modifies an attribute Value

7   Analog Input Channel Object

The Analog Input Channel Class allows the access to the IO-Data and the Register Data of each analog channel of the ILxxxx-B520 and the connected extension boxes. At Boot Up the ILxxxx-B520 determines the number of analog input channels and creates one instance of the object class for each channel.

The Analog Input Channel Object Class is Vendor specific and within in the range of the Vendor specific Class Codes.

Class Code: 103 (67$_{hex}$)

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description	Semantics of Value
1	get	Revision	UINT	Revision of implementation	1
2	get	Max. Instance	UINT	Max. number of instances	

Instance Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
1 (01$_{hex}$)	get	Value	Array of BYTE	Value of analog input channel	Number of Bytes differs between analog and special signal extension boxes
2 (02$_{hex}$)	get	Value Length	USINT	Value length in Bytes	
100 (64$_{hex}$)	get/set1	Register 0	DWORD	Value of Register 0 of the analog input channel	see semantics
:	:	:	:	:	:
106 (6A$_{hex}$)	get	Register 6	DWORD	Diagnosis Register	see semantics
:	:	:	:	:	:
163 (A3$_{hex}$)	get/set1	Register 63	DWORD	Value of Register 63 of the analog input channel	see semantics

1: before writing the registers the write protection of the registers has to be disabled.

7.1   Semantics

Register 0 ... 63

Within the response of a Get_Attribute_Single Service to the „Register Data" Attribute the status of the internal reading and the registers data is returned by the ILxxxx-B520. The meaning of the registers data is described in the ILxxxx-B520 manual.

Within the response of a Set_Attribute_Single Service to the „Register Data" Attribute the status of the internal reading is returned by the ILxxxx-B520.

Get_Attribute Response

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Status (Low Byte)							
1	Status (High Byte) : 0 = OK, >0 = Error							
2	Register data (Low Byte)							
3	Register data (High Byte)							

Set_Attribute Request

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Register data (Low Byte)							
1	Register data (High Byte)							
2	Not used							
3	Not used							

Set_Attribute Response

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Status (Low Byte)							
1	Status (High Byte) : 0 = OK, >0 = Error							

Common Services

Service Code	Service Name	Description
14 (0E$_{hex}$)	Get_Attribute_Single	Returns the contents of the specified attribute
16 (10$_{hex}$)	Set_Attribute_Single	Modifies an attribute Value

8 Analog Output Channel Object

The Analog Output Channel Class allows the access to the IO-Data and the Register Data of each analog output channel of the ILxxxx-B520 and the connected extension boxes. At Boot Up the ILxxxx-B520 determines the number of analog output channels and creates one instance of the object class for each channel.

The Analog Output Channel Object Class is Vendor specific and within in the range of the Vendor specific Class Codes.

Class Code: 104 (68_{hex})

Class Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
1	get	Revision	UINT	Revision of implementation	1
2	get	Max. Instance	UINT	Max. number of instances	

Instance Attributes

Attribute ID	Access rule	Name	Data type	Description	Semantics of Value
1 (01_{hex})	Get/set	Value	Array of BYTE	Value of analog output channel	Number of Bytes differs between analog and special signal extension boxes
2 (02_{hex})	get	Value Length	USINT	Value length in Bytes	
100 (64_{hex})	get/set1	Register 0	DWORD	Value of Register 0 of the analog output channel	see semantics
:	:	:	:	:	:
106 ($6A_{hex}$)	get	Register 6	DWORD	Diagnosis Register	see semantics
:	:	:	:	:	:
163 ($A3_{hex}$)	get/set1	Register 63	DWORD	Value of Register 63 of the analog output channel	see semantics

1: before writing the registers the write protection of the registers has to be disabled.

8.1 Semantics

Register 0 – 63

Within the response of a Get_Attribute_Single Service to the „Register Data" Attribute the status of the internal reading and the registers data is returned by the ILxxxx-B520. The meaning of the registers data is described in the ILxxxx-B520 manual.

Within the response of a Set_Attribute_Single Service to the „Register Data" Attribute the status of the internal reading is returned by the coupler.

Get_Attribute Response

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Status (Low Byte)							
1	Status (High Byte) : 0 = OK, >0 = Error							
2	Register data (Low Byte)							
3	Register data (High Byte)							

Set_Attribute Request

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Register data (Low Byte)							
1	Register data (High Byte)							
2	Not used							
3	Not used							

Set_Attribute Response

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Status (Low Byte)							
1	Status (High Byte) : 0 = OK, >0 = Error							

Common Services

Service Code	Service Name	Description
14 (0E$_{hex}$)	Get_Attribute_Single	Returns the contents of the specified attribute
16 (10$_{hex}$)	Set_Attribute_Single	Modifies an attribute Value

9 Indicators and Switches

9.1 Start-up procedure and Diagnostics LEDs

Start-up procedure and Diagnostic

After switching on, the ILxxxx-B520 immediately checks the connected configuration. Error-free start-up is signalled by the red "I/O ERR" LED being extinguished. If the "I/O ERR" LED blinks, an error in the area of the extension boxes is indicated. The error code can be determined from the frequency and number of blinks. This permits rapid rectification of the error. There is a detailed description in the section on "The diagnostic LEDs".

The diagnostic LEDs

The ILxxxx-B520 has two groups of LEDs for the display of status. The upper group with two LEDs indicates the status of the respective fieldbus. The significance of the "fieldbus status" LED is explained in the relevant sections of this manual - it conforms to conventional fieldbus displays.

On the bottom of the ILxxxx-B520 are two more green LEDs that indicate the supply voltage. The left hand LED indicates the presence of the 24 V supply for the ILxxxx-B520. The right hand LED indicates the presence of the supply to the power contacts.

Local errors

Two LEDs, the "I/O LEDs", in the area below the field bus status LEDs referred to above, serve to indicate the operating status of the ILxxxx-B520 and the connected extension boxes. The green LED lights up in order to indicate fault-free operation. The red LED blinks with two different frequencies in order to indicate an error. The error is encoded in the blinks as follows:

Blink code

Fast blinking	Start of the error code
First slow sequence	Error code
Second slow sequence	Error code argument

Error location

Error code	Error code argument	Description
1 pulse	0	EEPROM checksum error
	1	Inline code buffer overflow
	2	Unknown data type
2 pulses		Programmed configuration
	0	Incorrect table entry / Ilxxxx-B520
	n (n > 0)	Incorrect table comparison (extension box n)
3 pulses	0	Interruption of IP-Link bus
	n	Break behind extension box (0: ILxxxx-B520)
4 pulses	0	IP-Link bus data error (incorrect telegram)
	n	Extension box n (0: Ilxxxx-B520)
5 pulses	n	IP-Link bus error in register communication with extension box n
6 pulses	0	Special fieldbus error
	n (n > 0)	
11 pulses	n	IR Error, no communication, extension box n is not processing the IR commands
12 pulses	n	More than 120 extension boxes
		n extension boxes too much
13 pulses	n	unknown box type, extension box n

The number of pulses in the first sequence indicates the error type, while the second sequence indicates the position of the last extension box before the fault.

In the case of some errors, rectification does not cause the ILxxxx-B520 to leave the blink sequence. The ILxxxx-B520 stays in the "Stop" state. The ILxxxx-B520 can only be re-started either by switching the power supply off and on again, or by a scanner reset.

9.2   Fieldbus / DeviceNet LEDs

DeviceNet Status LED

The red/green LED pair provides information about the device and communication status of the ILxxxx-B520. The LEDs acts as the bi-color combined Module/Network Status LED defined in the DeviceNet Specification.

The LED pair is located next to the configuration interface for adjustment of the DeviceNet address (MacId)

States of Module / Status LED

LED State	Description
Green Flashing	Boot Up OK, Device has executed Duplicate MacId Check and is ON-Line. The ILxxxx-B520 is not allocated by a Master / Scanner, no Data Exchange with a Master / Scanner
Green ON	No Error, ILxxxx-B520 is allocated by a Master / Scanner, Data Exchange (Explicit or IO) with Master / Scanner is OK
Green OFF	• Bus Sense Error (24V DeviceNet Voltage in not available) (all LEDs off, including IO-Run, IO-Error LEDs) • No BaudRate, ILxxxx-B520 is not able to detect BaudRate (IO-Run, IO-Error LEDs On)
Red Flashing	Time Out, IO-Connection has timed out
Red ON	• Duplicate MacId Fault, check for same Address in Network • Bus-Off, check cabling, check bus termination, check bus length • Receive/Transmit Overrun, reduce IO-Cycle Time / Interscan delay at Master / Scanner
Red OFF	• Bus Sense Error (24V DeviceNet Voltage in not available) (all LEDs off, including IO-Run, IO-Error LEDs) • No BaudRate, ILxxxx-B520 is not able to detect BaudRate (IO-Run, IO-Error LEDs On)

9.3 DeviceNet Node Address Switches

Node Address Switches

The Node Address Switches consist of two, ten position rotary switches within the Configuration Interface of the ILxxxx-B520.

Node Address Switch

Node Address	Description
0 - 63	Node Address from Switches is valid, not programmable
> 63	Node Address is programmable by Master / Scanner

10 Appendix

10.1 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on her internet pages: https://www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline: +49 5246 963 157
Fax: +49 5246 963 9157
e-mail: support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline: +49 5246 963 460
Fax: +49 5246 963 479
e-mail: service@beckhoff.com

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone: +49 5246 963 0
Fax: +49 5246 963 198
e-mail: info@beckhoff.com
web: https://www.beckhoff.com