R67057.0002 - 2

Manual

Sendix S3674 (Singleturn)
Sendix S3684 (Multiturn)

RS485 + SinCos
BiSS + SinCos
BiSS

BiSS INTERFACE RS485

pulses for automation

Table of Contents

1 Document

This document is the English translation of the original German version.

Publisher	Kübler Group, Fritz Kübler GmbH Schubertstraße 47 78054 Villingen-Schwenningen Germany www.kuebler.com
Issue date	02/2021
Language version	German is the original language
Copyright	© 2021, Kübler Group, Fritz Kübler GmbH

Legal Notices

2 General Information

2.1 General Information

Please read this document carefully before working with the product, mounting it or starting it up.

2.2 Target Group

The device may only be planned, mounted, commissioned and serviced by persons having the following qualifications and fulfilling the following conditions:

- Technical training.
- Briefing in the relevant safety guidelines.
- Constant access to this documentation.
- In case of electrical equipment for potentially explosive atmospheres, the specialized personnel needs knowledge about the ignition protection category concept.
- For facilities in potentially explosive atmospheres, the authorized person must comply with the applicable country-specific regulations.

2.3 Symbols used / Classification of the Warnings and Safety instructions

⚠ DANGER	**Classification:** This symbol, together with the signal word **DANGER**, warns against immediately imminent threat to life and health of persons. The non-compliance with this safety instruction will lead to death or severe adverse health effects.
⚠ WARNING	**Classification:** This symbol, together with the signal word **WARNING**, warns against a potential danger to life and health of persons. The non-compliance with this safety instruction may lead to death or severe adverse health effects.
⚠ CAUTION	**Classification:** This symbol, together with the signal word **CAUTION**, warns against a potential danger for the health of persons. The non-compliance with this safety instruction may lead to slight or minor adverse health effects.

ATTENTION	**Classification:** The non-compliance with the **ATTENTION** note may lead to material damage.
NOTICE	**Classification:** Additional information relating to the operation of the product, and hints and recommendations for efficient and trouble-free operation.

2.4 Use According to the Intended Purpose

The encoder can be used as a rotation angle, position or speed sensor.

The encoder supports functions based on speed and rotation direction information, e.g. in industrial processes or controls.

This version is specific for use in the pitch system of a wind turbine.

As a sensor, the safe encoder supports safety functions regarding speed and rotation direction. The encoder is not able to achieve on its own a safe state in the safety function.

Dangerous situations must be detected and evaluated by a superordinate controller.

They can i addition be used at locations where inflammable mixtures can develop, see chapter Explosion protection.

The measuring system and its evaluation unit must meet the requirements mentioned in chapter Technical Data.

2.5 Foreseeable Misuse

The measuring system is not suitable for the following uses:

- Under water.
- In publicly accessible areas.
- As a standalone, independent safety system.
- Outside of the specifications stated in the data sheet..
- In areas where more serious EMC events may occur than those defined in the standard.

3 Product Description

3.1 General

Mechanical characteristics for the Sendix S36 encoders

Maximum rotational speed **	12000 min^{-1}
Maximum angular acceleration	5x10^5 rad/s²
Ambient temperature **	-40 °C ... +120 °C [-40 °F … +248 °F]
Protection level according to EN 60529	IP40
Installation height	< 2000 m [6562 ft]
Weight	appr. 0.1 kg [3.53 oz]
Shock resistance according to EN 60068-2-27	1000 m/s², 6 ms
Vibration resistance according to EN 60068-2-6	8.7 … 200 Hz, 30 m/s² 200 ... 2000 Hz, 500 m/s²

** Limitations, see chapter Limitations.

Electrical characteristics for the Sendix S36 encoders

Supply voltage	5 V DC, ± 10 % 7 … 30 V DC
Current consumption (no load) 5 V DC 10 … 30 V DC	 max. 150 mA max. 90 mA
Protection class according to EN 61140	III (PELV)

General technical data of the RS485 interface

Interface Absolute information	 RS485 (EIA485 acc. to IEEE 802.3) Hiperface® [1] compatible
Code	Binary
Type of connection	PCB connector radial
Singleturn resolution	12…24 bits/per revolution
Number of revolutions	4096 (12 bits)

[1] Hiperface® is a registered trademark of Sick Stegmann GmbH.

General technical data of the BiSS interface

Interface Absolute information	 RS485 (EIA485 according to IEEE 802.3)
Code	Binary
Clock rate	50 kHz … 10 MHz
Data up-to-dateness	≤ 10 µs (depending on clock rate and data length)
Singleturn resolution	12…24 bits/per revolution
Number of revolutions	4096 (12 bits)

EMC

Relevant standards	EN 55011 Class B:2009 / A1:2010 EN 61326-1:2013 EN 61326-3-1:2008

3.2 Interface Description RS485 + SinCos

RS485 interface

This interface has been developed specially for the requirements of the digital drive control.

Features overview

- One interface at the speed controller for all applications and one line between speed controller and motor feedback.
- Serial, synchronous, continuous and bidirectional data communication.
- Two unidirectional lines: Clock and Data.
- Low cycle time (up to 10 MHz with RS422 and 100 MHz with LVDS).
- Compensation of transmission delays for high-speed data transmission.
- Safety-capable: CRC, errors, warnings.
- Bus capability for several slaves and devices in a chain.
- Hybrid interface from the analog process data channel, for differential and almost delay-free transmission of sine and cosine signals.
- Bidirectional parameters channel complying with the RS485 specification for the transmission of the absolute position and other most varied parameters.
- Electronic type label for motor feedback identification and saving drive-relevant information in the motor feedback.
- The absolute value of the mechanical shaft position can be assigned electronically for commutation setting purposes.

Motor feedback systems are a mix of incremental encoder and absolute encoder, combining the advantages of both encoder types.

The absolute value is only generated at first when switching the device on. It is provided to the external counter in the controller via the bus-capable, RS485 specification-compliant parameter interface. Starting from this absolute value, the controller continues counting incrementally using the analog sine/cosine signals.

Signals processing in the motor controller

The specification defines a standard electrical interface with only 8 lines:

• 2x supply voltage 7 … 12 V.

• 4x incremental, differentially transmitted sine/cosine signals.

• 2x digital, bidirectional RS485 interface.

Serial protocol

As a standard, the interface is configured for 9600 bauds. The bidirectional RS485 interface is in principle bus-capable. This is why every communication from the master, hence from the controller, starts with the slave address. To minimize transmission times, the information is transmitted in binary format. Every protocol ends with an easily calculated XOR checksum. The protocol end detection is based on a timeout control.

The motor feedback system has internal diagnosis functions and reports critical or faulty conditions in the answer protocol.

The physical interface requires a 130 Ω termination resistor, two biasing resistors and a standard RS485 transceiver. The protocol is executed with a standard UART, which is implemented on almost all usual microcontrollers/DSPs.

SinCos Interface

The sine/cosine signals are transmitted fully differentially. Their amplitude varies by at the maximum 20%, whatever the conditions.

Also refer to

Status Byte [▶ 19]

3.3 BiSS Interface Description

BiSS interface

The bidirectional digital sensor-actuator interface BiSS ensures the communication between intelligent sensors such as position transmitters or measuring equipment, actuators and industrial controls. It is bus-capable to serve sensors and actuators simultaneously.

The BiSS protocol classifies every device in one of the following data areas:

• Sensor data

• Actuator data

• Register data

The access to and transmission performance of these data areas are configured differently in order to cover a range of different sensor applications.

The bidirectional parameters communication required for device configuration is transferred generally in the register data area. Quickly changing sensor data is assigned to the sensor data area.

Shielding

Also the shielding concept is of major importance for the achievable global system performance. A cable shield applied on a large area both on the motor and on the controller generally guarantees here the best results. The shield should be connected to the protective earth. If large compensating currents are expected, use a separate equipotential bonding conductor.

Also refer to

Status Byte [▶ 19]

4 Installation

4.1 Electrical Installation

4.1.1 General Information for the Connection

ATTENTION	**Destruction of the device** Before connecting or disconnecting the signal cable, always disconnect the power supply and secure it against switching on again.

NOTICE	**General safety instructions** Make sure that the whole plant remains switched off during the electrical installation. • Make sure that the operating voltage is switched on or off simultaneously for the device and the downstream device.

NOTICE	**Traction relief** Always mount all cables with traction relief.

4.1.2 Terminal Assignment

Digital interface	Incremental interface	Type of connection	PCB connector (male contact), 10-pin										
1 (BiSS)	1 (SinCos)	1	Signal:	0 V	+V	D+	D-	C+	C-	A	$\overline{A}$	B	$\overline{B}$
			Pin:	1	2	3	4	5	6	7	8	9	10
suitable pre-assembled cordset, 10-core (8.0000.D111.0M50)			Core color:	BU	RD	GY	GN	YE	VT	PK	BK	WH	BN

IMG-ID: 9007199428518923

Digital interface	Incremental interface	Type of connection	PCB connector (male contact), 10-pin										
2 (RS485)	1 (SinCos)	1	Signal:	0 V	+V	D+	D-	–	–	A	$\overline{A}$	B	$\overline{B}$
			Pin:	1	2	3	4	5	6	7	8	9	10
suitable pre-assembled cordset, 10-core (8.0000.D111.0M50)			Core color:	BU	RD	GY	GN	–	–	PK	BK	WH	BN

IMG-ID: 9007199428525835

Digital interface	Incremental interface	Type of connection	PCB connector (male contact), 10-pin										
1 (BiSS)	A (ohne)	1	Signal:	0 V	+V	D+	D-	C+	C-	–	–	–	–
			Pin:	1	2	3	4	5	6	7	8	9	10
suitable pre-assembled cordset, 10-core (8.0000.D111.0M50)			Core color:	BU	RD	GY	GN	YE	VT	–	–	–	–

IMG-ID: 9007199428522379

+V:	Encoder supply voltage +V DC
0 V:	Encoder ground GND (0V)
D+, D-:	Data signal
C+, C-:	Clock signal
A, A\:	Incremental output Channel A (Cosine)
B, B\:	Incremental output Channel B (Sine)

RS485 specifies a standard voltage range for the slave supply of 7 ... 12 V DC (including the tolerances). The supply voltage is measured on the slave; the maximum voltage drop over the cable length must be considered accordingly when dimensioning the master supply.

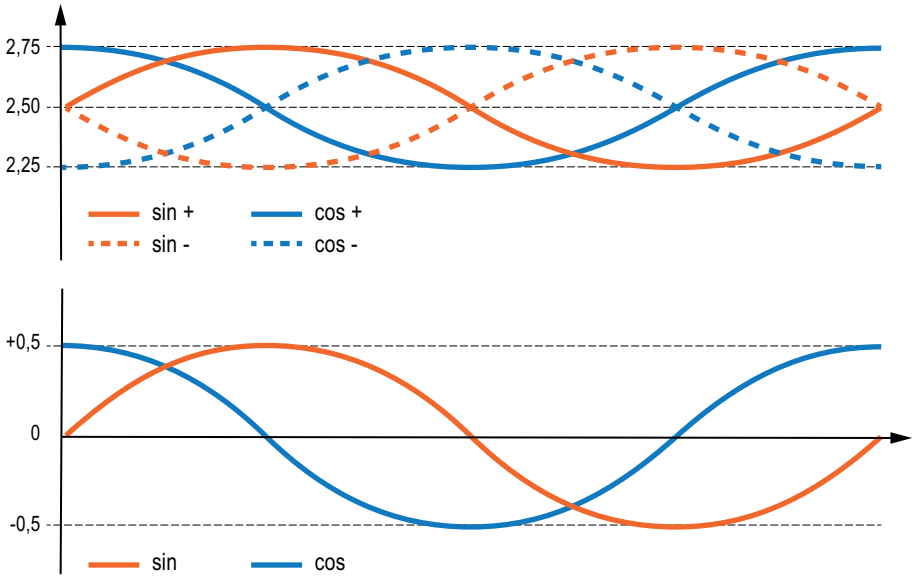
RS485 specifies a maximum current consumption by the slave of 250 mA at 7 V DC.

4.1.3 SinCos Interface

The process channel uses two sets of differential analog voltages.

Lines for transmitting sine/cosine signals. The signal links are called "SIN" and "COS", while the voltage references are called "REFSIN" and "REFCOS".

Sine/cosine voltages are transmitted according to the following parameters:



IMG-ID: 64973067

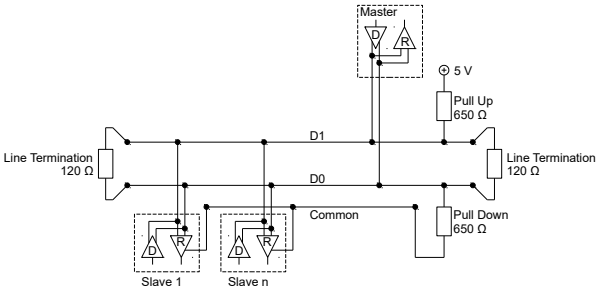
The upper signal path shows the absolute signal amplitudes. The lower picture area shows the differential amplitude.

Parameter	Value	Comment
SinCos peak voltage	2 … 3 V	Measured against GND
SinCos peak-to-peak voltage	0.4 … 0.6 V	Measured differentially
DC level	2.2 … 2.8 V	Measured against GND
SinCos bandwidth	0 … 2048 kHz	3 dB signal amplitude
Output load capacity	> ± 5 mA	

4.1.4 RS485 Network Topology

The parameters channel is an asynchronous half-duplex interface that complies physically with the RS485 specification. To ensure disturbance-free transmission, the transmission lines must be provided with pull up/down resistors.

In order to also ensure independence from line transceiver manufacturers, we give the following recommendations for the dimensioning of the interface:



IMG-ID: 58511243

The 120 … 130 ohm terminating resistor and the 650 … 1k ohm pull up/down resistors are integrated as a standard. For bus operation (more than 1 encoder), no terminating resistor is integrated in the encoder. This resistor must be mounted by the customer on the device furthest from the master.

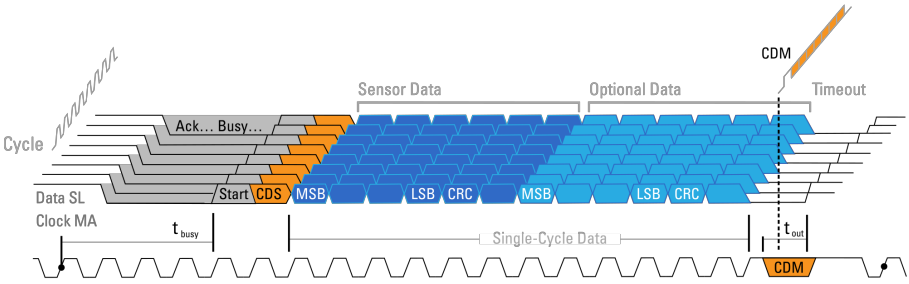
4.1.5 BiSS Interface

The BiSS-C interface provides a bidirectional isochronous connection between sensors, drives and industrial encoder evaluation equipment. This purely digital connection and its protocol have been developed for maximum performance, reliability and transmission safety.

The communication protocol integrates constant bidirectional access to the slave registers, without affecting the useful measurement data or the control cycles. So the device parameters and additional measurement data, as well as the electronic type label and the OEM data, are accessible at any time.

The interface master provides the clock signal for simultaneously triggered functions to an unlimited number of devices. So, for example, a typical RS422 connection can support a frame repetition rate of 10µs for data words with a length reaching 64 bits.

The data transmission is fully and separately secured via CRC for the bidirectional transmission of commands and registers and for every single-break channel. An initial value is assigned, which allows identifying the channel in the event of safety checks. The control data bits from several successive BiSS frames are combined by the master in a control frame. This allows the read and write access of the slaves, and the sending of commands to the slaves.



IMG-ID: 53491083

Further information is available under: www.kuebler.com/biss.

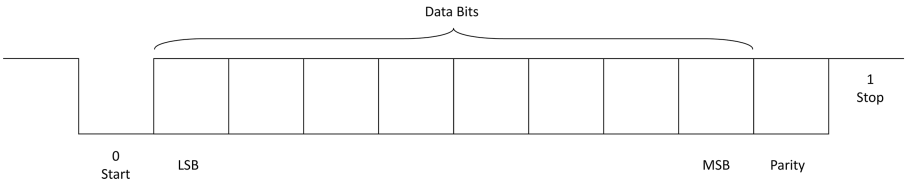
5 Commissioning and Operation

5.1 RS485

5.1.1 Data Transmission

5.1.1.1 Frames Structure

The data format is an asynchronous half-duplex UART protocol with binary data.



IMG-ID: 9007199317760139

The data at the asynchronous interface has the following basic format:

- 1 Start bit
- 8 Data bits (LSB first)
- 1 Stop bit
- 1 Parity bit (optional): Even (programmable: odd, even, none)
- Baud rate: 9600 (programmable: 600, 1200, 2400, 4800, 19200, 38400 bauds)

Data transmission on the RS485 is controlled by a timeout protocol. This means that, if no further information reaches the MFB within a defined time (timeout), the currently received protocol is processed.

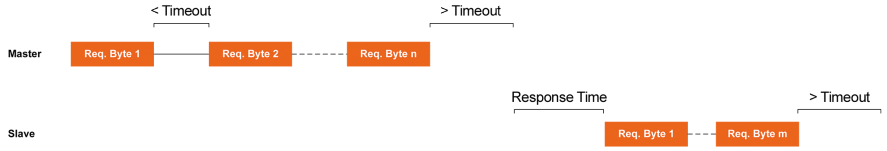
Interruptions of a data packet that last longer that the timeout lead to corresponding protocol error messages (Command byte 50h - Read encoder status [▶ 32]). The first byte after a timeout is interpreted as an address.

Every character of the master transmission must be followed by a subsequent character within the timeout condition. If the slave receives no further character within the timeout condition, the message is considered as completed and is processed.

Consequently, the end of a master transmission must be followed by a pause of at least the length of the timeout condition.

The end of a slave transmission is also indicated by a pause of at least the length of the timeout condition. The timeout condition can be programmed either on 22 s/data rate or 55 s/data.

Rate (standard, i.e. 5.73 ms for 9600 bauds)



IMG-ID: 9007199323462539

5.1.1.2 Data Addresses

The protocol allows the use of a bus topology. Consequently, an addressing scheme is implemented.

Every frame includes an address byte, a command, and optionally various data bytes, length and a subsequent checksum, to detect transmission errors. This description applies both to master ad slave transmissions.

IMG-ID: 9007199323480075

An address consists of a byte. The available address space allows 32 different address values.

Bit	Definition	Default
0 … 4	Address value	40h
5	0	
6	1	
7	0	

The protocol also implements a broadcast address (FFh) that can be used by a master. Address all slaves connected to the bus to perform a selected command. Every slave is confirmed with its specific address.

Command format

A command consists of at least one byte. The available command space allows 64 different command values. Depending on the command, additional bytes are necessary.

Bit	Definition
0 … 5	Command
6	Fix 1
7	Error bit

In the protocol, every frame ends with a check sum (both, from master and slave). The receiver of a frame uses this check sum to perform a transmission error check and can then react accordingly.

The check sum including the address is defined by a XOR operation for all bytes of the frame.

Frame	Check sum
40h 42h	02h
40h 4Eh 70h 00h	7Eh

Error displays

The protocol uses two different ways to display detected errors. Errors are always transmitted only from the slave to the master.

Warning bit

If a slave detects an operating error that is not related to a current command, a warning bit is emitted with the query and a corresponding error code is saved for query. This can be queried by the "Read encoder status" command.

Malfunctions are identified as error messages. A slave can save in parallel the error codes of up to four malfunctions. The warning bit is active when the error bit (bit 7) is set.

In this condition, the initial command prompt of the master is answered properly.

The slave emits the warning bit with every subsequent command response until the master has read the error code (with command 50h) or a slave reset (command 53h) has been performed. If several malfunctions are saved, the master must read every corresponding error code separately to reset the error bit.

Error response

If a slave detects a condition that does not allow the processing to the master, an error response is generated in the command prompt. Typical conditions for this behavior are protocol errors, invalid command arguments or internal encoder errors. Error states are listed in the error messages. In the event of an error, the slave interrupts the processing of the command and answers with an error response. The error response contains the status command value 50h, followed by a corresponding error code.

If a warning and an error condition are detected for the same command response, the slave emits an error response with a set warning bit (set value D0h). The error code corresponding to the warning must be read out.

5.1.1.2.1 Data Fields Specification

RS485 slave devices implement a function that allows the user to save any data in the non-volatile memory of the slave. RS485 allows saving in the EEPROM of the slave with the following specification:

- The data is saved in "data field" units having a size of 16 to 128 bytes.
- Prior to saving the data, a data field with a prescribed size must be created.
- Data fields are identified by a consecutive "data field number" starting with 0.
- Data fields can be linked to one of four access code values (Code0 to Code3). To read or write a data field, the user must submit the access code.
- Data fields can be set as write-protected or write-enabled. Only "write-enabled" allows writing in a data field.
- The status of a data field can be modified (linked code, read/write mode). For the last data field (highest data field number), also the data field size can be increased.
- Data fields can be erased.

The data sheet of a slave must indicate the number of bytes a user can save in data fields. The number of bytes indicated this way includes 100% of the user content.

All values required for data field management are stored separately.

User memory size is normally 1792 bytes.

5.1.1.2.2 Extended Type Label Specification

Data fields are also used for the "Extended Type Label" function.

Newer slaves allow the master reading out explicitly encoder parameters.

The access to the encoder parameters of the extended type label is identical with the reading of user data fields using the virtual data field number FFh.

Extended type label data is not saved for users in the non-volatile memory, but in a separate memory.

Encoder type	Types code	Extended type label
Previously	Not FFh	No
Currently	Not FFh	Yes
Newly and in the future	FFh	Yes

The extended type label is not counted as an active data field in the "number of data fields" of command "Memory status".

The extended type label is accessible without access code and can only be read. It cannot be erased. It has a minimum length of 64 bytes. The actual size can be larger if bit 15 of the parameters selector is activated (see following table, Offset 1Ch).

The master can read one or all bytes within the extended type description, as indicated in a "Read date" command.

The data of an extended type description is displayed according to the following table.

Offset	Length	Description	Note
00h	1	XOR check sum complete Extended Type Label	
01h	1	Encoder type Bit 0: 0 = rotary, 1 = linear Bit 1: 0 = unipolar count, 1 = bipolar count Other bits: not implemented, "0"	
02h	4	Encoder resolution (unsigned 32-bit value) number of periods / revolution for encoder period length in steps of 1 nm for linear encoders	
06h	4	Encoder range (unsigned 32-bit value) Number of coded revolutions for rotary encoders Number of periods for linear encoders	
0Ah	18	Encoder designation (ASCII characters chain, left-justified with trailing 00h characters)	
1Ch	2	682/5000 Parameters selector (indication of the available values in the extended type label) Bit 0: Temperature channel selector disabled / enabled Bit 1: Temperature minimum value disabled / enabled Bit 2: Temperature maximum value disabled / enabled Bit 3: LED current channel selector disabled / enabled Bit 4: Minimum LED values disabled / enabled Bit 5: Maximum LED value disabled / enabled Bit 6: Vector length channel selector disabled / enabled Bit 7: Vector length minimum value disabled / enabled Bit 8: Vector length maximum value disabled / enabled Bit 9: Maximum speed value disabled / enabled Bit 10: Maximum acceleration value disabled / enabled Bit 11… 14: not implemented, "0" Bit 15: Subsequent selector disabled / enabled	If the subsequent selector is enabled (bit 15 = 1), the second parameters selector follows at offset 40h and 2nd parameter settings at offset 42h etc.
1Eh	30	Parameter values as defined in the parameters selector. Every parameter is a 16-bit value Byte 0… 1: Temperature channel value (for command 44h) Byte 2… 3: Temperature min. value (signed, in ° C) Byte 4… 5: Maximum temperature value (signed, in ° C) Byte 6… 7: LED current channel value (for command 44h) Byte 8… 9: LED current min. value (unsigned, in mA) Byte 10… 11: LED current max. value (unsigned, in mA)	

Offset	Length	Description	Note
		Byte 12… 13: Vector length CHANNEL value (for command 44h) Byte 14… 15: Vector length min. value (unsigned, any units) Byte 16… 17: Vector length max. value (unsigned, any units) Byte 18… 19: Maximum speed value (unsigned, in RMP or m/min) Byte 20… 21: Maximum acceleration value (unsigned, in krad/s² or m/s²) Bit 22… 29: not implemented, read "0"	
3Ch	4	Not implemented, read "0"	

5.1.1.2.3 Status Byte

The status byte consists of 8 bytes.

Up to 4 access codes are possible. These serve as a kind of write protection ensuring that only authorized users have access to the implemented functions.

NOTICE	**Default Access Code**
	The default access code is 55 and applies to all commands.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Check data field	R/W access	Access key		Enable/disable access code	Data field size		
0 = not found	0 = read access	Access Code 0		0 = code disabled	Calculate memory requirement		(Size/16)-1
1 = found	1 = read and write access	0	0	1 = code enabled			
		Access Code 1			16 bytes		
		0	1		0	0	0
		Access Code 2			32 bytes		
		1	0		0	0	1
		Access Code 3			48 bytes		
		1	1		0	1	0
					64 bytes		
					0	1	1
					80 bytes		
					1	0	0
					96 bytes		
					1	0	1
					112 bytes		
					1	1	0
					128 bytes		
					1	1	1

5.1.1.2.4 RS485 Settings

The settings of the serial interface are defined in 8 bits. The default settings are printed in bold.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
BUS MFB	Time out	UART		not used	Baud rate		
BUS	1*11/Baud rate	8 data bits, Parity = none			600Bd		
0	0	0	0		0	0	0
Standard	4*11/Baud rate	8 data bits, Parity = odd			1200Bd		
1	**1**	**1**	**0**		0	0	1
		8 data bits, Parity = even			2400Bd		
		1	1		0	1	0
					4800Bd		
					0	1	1
					9600Bd		
					1	**0**	**0**
					19200Bd		
					1	0	1
					38400Bd		
					1	1	0

5.1.2 Supported Commands

The commands in the application layer are defined with the following values:

Parameter	Description	Note
Command identifier	Command address acc. to command format	40h … 7FH
Access code	Indication whether access to the command is only possible with the transmitted code byte ("code 0"). Standard value for code 0 is 55h.	Other codes only with data fields
Reaction time	Max. reaction time of the slave after a master query in ms	
Master data length	Number of bytes of the necessary data bytes for the query	Does not contain address, command ID or check sum
Slave data length	Number of bytes of the data bytes for the response	Does not contain address,command identification or check sum
Master data values	Definition of the data bytes for the query	
Slave data values	Definition of the data bytes for the response	
Error conditions	Corresponding error conditions	Does not contain global warnings

The following commands are supported by the system:

MFB supported commands		Comment			Reference
Command byte	Function	Code 0	MFB-ST	MFB-MT	
42h	Read position		20 (32) bits	12 (32) bits	Command byte 42h - Read position [▶ 23]
43h	Set position	Yes			Command byte 43h - Set position [▶ 23]
44h	Read analog value				Command byte 44h - Read analog value [▶ 24]
46h	Read counter				Command byte 46h - Read counter [▶ 24]
47h	Increment counter				Command byte 47h - Increment counter [▶ 25]
49h	Erase counter	Yes			Command byte 49h - Erase counter [▶ 25]
4Ah	Read data				Command byte 4Ah - Read data [▶ 26]
4Bh	Save data				Command byte 4Bh - Save data [▶ 26]
4Ch	Determine the status of a data field				Command 4Ch - Determine data field status [▶ 27]
4Dh	Create data field				Command byte 4Dh - Create data field [▶ 28]
4Eh	Determine available memory area				Command byte 4Eh - Determine available memory area [▶ 31]
4Fh	Modify access key				Command byte 4Fh - Modify access key [▶ 31]
50h	Read encoder status				Command byte 50h - Read encoder status [▶ 32]
52h	Read out type label		Encoder type = 22h	Encoder type = 27h	Command byte 52h – Read type label [▶ 32]
53h	Encoder reset				Command byte 53h – Encoder reset [▶ 33]
55h	Assign encoder address	Yes			Command byte 55h – Assign encoder address [▶ 34]
56h	Read serial number and program version				Command byte 56h – Read serial number and program version [▶ 35]
57h	Configure serial interface	Yes			Command byte 57h – Configure serial interface [▶ 35]
6Ah	Set position with synchronization				Command byte 6Ah – Set position with synchronization [▶ 36]

5.1.2.1 Command byte 42h - Read position

The command "Read position" allows the motor controller to read out the absolute position value of the encoder.

The slave type label must indicate the maximum speed at which the absolute position value can be acquired.

Position acquisition is buffered with the falling edge of the start bit of the first byte of the slave response (address byte).

The absolute position value is used to identify unambiguously one of the analog sine/cosine periods. It has a resolution of 5 bits per period in all slaves.

Parameter	Value	Note
Command identifier	42h	
Access code	-	
Reaction time	10 ms	
Master data length	0 bytes	
Slave data length	4 bytes	
Master data values	-	
Slave data values	Absolute position – unsigned 32-bit value, MSB first	
Error conditions	02h (Angle offset fault) 0Dh (Wrong argument) 1Dh (LED current high) 1Fh (Speed high) 20h (Singleturn fault) 21h (Multiturn amplitude fault) 22h (Multiturn sync fault) 23h (Multiturn vector length fault)	

5.1.2.2 Command byte 43h - Set position

The command "Set position" allows the motor controller to save a position offset in the slave.

This command simplifies the use of the absolute position for motor commutation. Normally, the master would set the absolute position of the slave to 0 when the motor shaft position is locked.

The master must transmit the desired position value (Preset) with the command "Set Position". The slave calculates the corresponding position offset and saves it internally.

The command "Set position" can only be activated when "Code0" is sent, to prevent misuse and impairment of the motor function.

The command "Set position" allows setting any position within the measuring range. This allows modifying the phase shift between the absolute position and the analog signals of the slave. At delivery, the phase shift of the slave is always 0°.

Parameter	Value	Note
Command identifier	43h	
Access code	Code 0	Default: 55h
Reaction time	40 ms	
Master data length	5 bytes	
Slave data length	0 bytes	
Master data values	Byte 0 … 3: Absolute position preset as unsigned 32-bit value, MSB first Byte 4: Code 0	
Slave data values	Absolute position – unsigned 32-bit value, MSB first	
Error conditions	02h (Angle offset fault) 05h (I2C Communication fault) 06h (EE checksum fault) 0Dh (Wrong argument) 0Fh (Wrong access code) 1Dh (LED current high) 20h (Singleturn fault) 21h (Multiturn amplitude fault) 22h (Multi sync fault) 23h (Multiturn vector length fault)	

5.1.2.3 Command byte 44h - Read analog value

Reading the analog value allows access to sensor sources in the slave. Encoder temperature would be a typical sensor source.

Single sensor sources are addressed by the master with a "CHANNEL" selector.

The available sensor sources, their coding and resolution, are always indicated in the product data sheet.

Parameter	Value	Note
Command identifier	44h	
Access code	-	
Reaction time	5 ms	
Master data length	1 byte	
Slave data length	3 bytes	
Master data values	CHANNEL selector	
Slave data values	Byte 0: CHANNEL selector Byte 1 … 2: Sensor value	
Error conditions	0Dh (Wrong argument)	

5.1.2.4 Command byte 46h - Read counter

The command "Read counter" returns the value of an implemented non-volatile counter of the slave.

The counter has a 24-bit width and allows 1 million write cycles.

Parameter	Value	Note
Command identifier	46h	
Access code	-	
Reaction time	5 ms	
Master data length	0 bytes	
Slave data length	3 bytes	
Master data values	-	
Slave data values	Byte 0 … 2: Counter – unsigned 24-bit value	
Error conditions	06h (EE checksum fault)	

5.1.2.5 Command byte 47h - Increment counter

The command "Increment counter" increases the value of the implemented non-volatile slave counter by one.

The counter has a 24-bit width and allows 1 million write cycles.

Parameter	Value	Note
Command identifier	47h	
Access code	-	
Reaction time	30 ms	
Master data length	0 bytes	
Slave data length	0 bytes	
Master data values	-	
Slave data values	-	
Error conditions	05h (I2C Communication fault) 06h (EE checksum fault) 08h (Counter fault)	

5.1.2.6 Command byte 49h - Erase counter

The command "Erase counter" writes 0 in the implemented non-volatile counter of the slave

The counter has a 24-bit width and allows 1 million write cycles.

Parameter	Value	Note
Command identifier	49h	
Access code	Code 0	Default: 55h
Reaction time	30 ms	
Master data length	1 byte	
Slave data length	0 bytes	
Master data values	Byte 0: Code 0	
Slave data values	-	
Error conditions	05h (I2C Communication fault) 06h (EE checksum fault) 08h (Counter fault)	

5.1.2.7 Command byte 4Ah - Read data

The command "Read data" allows reading data value fields from a slave.

The user gives the following information:

- Data field number (as created with command 4Dh, see Create data field [▶ 28]) or extended type label code FFh (see Specification for Extended Type Label [▶ 17]).
- Offset within the data field; must be within the data field size (0 … size-1).
- Number of bytes to be read (min. 1 byte, max. 128 bytes). Offset plus number of bytes may not exceed the size of the data field.
- Access code. If the data field is created without access code requirement, this byte can contain any value. It must however be contained in the master request.

Parameter	Value	Note
Command identifier	4Ah	
Access code	Code 0^… Code 3	Depending on the definition from the data field
Reaction time	30 ms	
Master data length	4 bytes	
Slave data length	4 … 131 bytes	Depending on the number of bytes (1 … 128)
Master data values	Byte 0: Data field number Byte 1: Offset in the data field Byte 2: Number of bytes Byte 3: Access code	
Slave data values	Byte 0: Data field number Byte 1: Offset in the data field Byte 2: Number of bytes Byte 3 ... 130: Data	
Error conditions	03h (Data field partition table fault) 05h (I2C communication fault) 0Fh (Wrong access code) 11h (Wrong data field offset) 12h (Wrong data field number)	

5.1.2.8 Command byte 4Bh - Save data

The command "Save data" allows saving user data in data fields of a slave.

The user cannot save data in a data field created with or modified for the status "read only".

Data fields are not automatically enlarged when the maximum size is reached. A data field must be explicitly enlarged with the command "Create data field (see Create data field [▶ 28]). This is only possible for the last data field created (highest data field number).

The user gives the following information:

- Data field number (as created with command 4Dh, see Command byte 4Dh - Create data field [▶ 28]).
- Offset within the data field; must be within the data field size (0 … size-1).

- Number of bytes to be written (min. 1 byte, max. 128 bytes). Offset plus number of bytes may not exceed the size of the data field.
- Access code. If the data field is created without access code requirement, this byte can contain any value. It must however be contained in the master request.
- Data bytes

Parameter	Value	Note
Command identifier	4Bh	
Access code	Code 0 … Code3	Default: 55h
Reaction time	250 ms	
Master data length	5 … 132 bytes	
Slave data length	3 bytes	
Master data values	Byte 0: Data field number Byte 1: Offset in the data field Byte 2: Number of bytes Byte 3: Access code Byte 4… 131: Data	
Slave data values	Byte 0: Data field number Byte 1: Offset in the data field Byte 2: Number of bytes	
Error conditions	03h (Data field partition table fault) 05h (I2C communication fault) 06h (EE checksum fault) 0Eh (Data field is read-only) 0Fh (Wrong access code) 11h (Wrong data field offset) 12h (Wrong data field number)	

5.1.2.9 Command 4Ch - Determine data field status

The command "Determine data field status" allows reading the data field status of a slave.

Data field number can be an existing data field (as created with command 4Dh, see Create data field [▶ 28]) or an extended type label code FFh (see Specification for Extended Type Label [▶ 17]).

The data field status is returned as a byte with the same coding as for the command "Create data field".

The counter has a 24-bit width and allows 1 million write cycles.

Parameter	Value	Note
Command identifier	4Ch	
Access code	-	
Reaction time	5 ms	
Master data length	1 byte	
Slave data length	2 bytes	
Master data values	Byte 0: Data field number	
Slave data values	Byte 0: Data field number Byte 1: Status Bit 0 … 2: Data field size in 16-byte blocks (size / 16-1), i. e. 0 = 16 bytes, 1 = 32 bytes, …, 7 = 128 bytes Bit 3: Code disabled / enabled Bit 4 ... 5: Access code, 0 = Code0, 1 = Code1, 2 = Code2, 3 = Code3 Bit 6: 0 = write-protected, 1 = write-enabled Bit 7: 1 = data field found	Code0 for data fields is identical with Code0 for the access to some commands
Error conditions	03h (Data field partition table fault) 05h (I2C communication fault) 12h (Wrong data field number)	

5.1.2.10 Command byte 4Dh - Create data field

The command "Create data field" allows creating new and modifying existing data fields of a slave.

To create new data fields, the user gives the following information:

- The data field number must be identical with the next available number (i. e. no existing data fields - 0, three existing data fields - 3, etc.).
- Data field status as a byte with the same coding as for the command "Data field status". Bit definitions, see the table below.
- Access code. The transmitted access code byte must correspond to the access code selected for the data field. This is necessary even during data field modification if the access code for the data field is disabled. These codes are not required for reading and writing if they have been disabled for the respective data field during creation or modification. In this case, any byte is transmitted instead of the code.
- The access code is disabled for the data field.

To modify data fields, the user gives the following information:

- Data field number (as created previously).
- Data field status as a Byte with the same coding as for the command "Data field status" (see Command 4Ch - Determine data field status [▷ 27]). Bit definitions, see the table below.

To create a data field, bit 7 must be set to 1.

To modify a data field, bit 7 must be set to 1. It must be set to 0 to erase a data field.

Erasing or modifying data field size is only possible for the last data field created.

For erasing, all status bits (except bit 7) must correspond to the saved values of the data field.

Parameter	Value	Note
Command identifier	4Dh	
Access code	Code0 … Code3	Depending on the user definition of the data field
Reaction time	70 ms	
Master data length	3 bytes	
Slave data length	2 bytes	
Master data values	Byte 0: Data field number Byte 1: Status Bit 0 … 2: Data field size in 16-byte blocks (size / 16 - 1), i. e. 0 = 16 bytes, 1 = 32 bytes, ... 7 = 128 bytes Bit 3: Code disabled / enabled Bit 4...5: Access code, 0 = Code0, 1 = Code1, 2 = Code2, 3 = Code3 Bit 6: 0 = write-protected, 1 = write-protected Bit 7: 0 = erase data field, 1 = cerate / modify data field Byte 2: Access code	
Slave data values	Byte 0: Data field number Byte 1: Status Bit 0 … 2: Data field size in 16-byte blocks (size / 16-1), i. e. 0 = 16 bytes, 1 = 32 bytes,…, 7 = 128 bytes Bit 3: Code disabled / enabled Bit 4 ... 5: Access code, 0 = Code0, 1 = Code1, 2 = Code2, 3 = Code3 Bit 6: 0 = write-protected, 1 = write-protected	Code0 for data fields is identical with Code0 for the access to some commands

Parameter	Value	Note
	Bit 7: 0 = data field erased, 1 = data field created / modified	
Error conditions	03h (Data field partition table fault) 05h (I2C communication fault) 06h (EE checksum fault) 0Fh (Wrong access code) 10h (Fixed data field size) 12h (Wrong data field number)	

5.1.2.11 Command byte 4Eh - Determine available memory area

The command "Determine available memory area" returns information on the free user memory and on existing data fields of the slave.

The free user memory is returned as a number of available 16-byte blocks (e. g. a value of 112/70h is returned for 1792 free bytes).

The counter has a 24-bit width and allows 1 million write cycles.

Parameter	Value	Note
Command identifier	4Eh	
Access code	-	
Reaction time	5 ms	
Master data length	0 bytes	
Slave data length	2 bytes	
Master data values		
Slave data values	Byte 0: Free user memory (in 16-byte blocks) Byte 1: Number of existing data fields	
Error conditions	06h (EE checksum fault)	

5.1.2.12 Command byte 4Fh - Modify access key

The command "Modify access key" allows modifying the access code bytes of the slave. Each of the four different code bytes (Code0, Code1, Code2 and Code3) can be modified.

The standard value of every code byte is 55h.

The interface does not provide the users with a function to reset the access code bytes to the standard values.

Parameter	Value	Note
Command identifier	4Fh	
Access code	-	
Reaction time	40 ms	
Master data length	3 bytes	
Slave data length	1 byte	
Master data values	Byte 0: Access code (00h… 03h) Byte 1: Old access code Byte 2: New access code	Default: 55h
Slave data values	Byte 0: Access code (00h… 03h)	
Error conditions	05h (I2C communication fault) 06h (EE checksum fault) 0Dh (Wrong argument) 0Fh (Wrong access code)	

5.1.2.13 Command byte 50h - Read encoder status

The command "Read encoder status" returns the active warning and error codes of the slave.

For every "Read encoder status" command, an active warning or error from the slave error stack is returned. Slaves can save up to four active warnings or errors in the stack.

If a warning or error occurs several times, the corresponding code is only saved once in the stack until it is read.

If more than four warnings or errors occur before the stack is read, the oldest entries are erased.

If no warning or error is active, "Read encoder status" returns the status code 00h.

An active warning message can only be reset by the user if he reads out all active warnings and errors.

The meaning of a warning or error code is defined in the product data sheet. Typical displays and their codes are listed in error messages.

Parameter	Value	Note
Command identifier	50h	
Access code	-	
Reaction time	5 ms	
Master data length	0 bytes	
Slave data length	1 byte	
Master data values	-	
Slave data values	Byte 0: Encoder status byte	Status 00h, warning or error code
Error conditions		

5.1.2.14 Command byte 52h – Read type label

The command "Read type label" returns slave device and interface information.

- Parameter interface settings (UART) with the same coding as for the command "Configure serial interface" (see Configure serial interface [▶ 35])
- Extended type label, see Extended Type Label Specification [▶ 17]
- Global memory size in 16-byte blocks (e. g. a value of 128 / 80h is returned for 2048 implemented memory bytes)
- Option code for determined available hardware / software / version options. This value is always 00h for new slaves.

Parameter	Value	Note
Command identifier	52h	
Access code	-	
Reaction time	5 ms	
Master data length	0 bytes	
Slave data length	4 bytes	
Master data values	-	
Slave data values	Byte 0: UART settings Bit 0 … 2: Data rate 000 = 600 bauds 001 = 1200 bauds 010 = 2400 bauds 011 = 4800 bauds 100 = 9600 bauds (standard) 101 = 19200 bauds 110 = 38400 bauds Bit 3: not implemented Bit 4 … 5: Data bits, parity 00 = 8 data bits, no parity 10 = 8 data bits, even (standard setting) 11 = 8 data bits, odd Bit 6: Timeout 0 = 2 * 11 / Data rate 1 = 5 * 11 / Data rate (standard) Bit 7: Bus 0 = Bus slave 1 = Standard slave (standard) Byte 1: Encoder type 00… FEh = determined encoder type FFh = extended type label available Byte 2: Memory size in 16-byte blocks Byte 3: Option code	Status 00h,warning or error code
Error conditions	05h (I2C communication fault)	

5.1.2.15 Command byte 53h – Encoder reset

The command "Encoder reset" allows the software reset of the slave.

The software reset leads to the same initialization process as after powering.

Some older products show small differences between powering and software reset. New and future products must show an identical behavior.

Configuration data such as the slave address and interface settings are not modified by an "Encoder reset".

A proper "Encoder reset" request is followed by a slave response message.

The "Encoder reset" is performed by the slave after the time limit of the request has elapsed. A master must wait this time plus the initialization time of 100 ms before the communication can be resumed.

Parameter	Value	Note
Command identifier	53h	
Access code	-	
Reaction time	-	No response The slave must wait for new requests after the initialization time of (100 ms) plus the timeout for this request.
Master data length	0 bytes	
Slave data length	-	No feedback
Master data values	-	
Slave data values	-	
Error conditions		

5.1.2.16 Command byte 55h – Assign encoder address

The command "Assign encoder address" allows setting the slave address of the device.

Slave addresses are valid from 40h (standard setting) up to 5Fh.

After the successful execution of this command, the slave response message immediately uses the new slave address.

Parameter	Value	Note
Command identifier	55h	
Access code	Code 0	
Reaction time	40 ms	No response The slave must wait for new requests after the initialization time of (100 ms) plus the timeout for this request.
Master data length	2 bytes	
Slave data length	0 bytes	No feedback
Master data values	Byte 0: New slave address (40 … 5Fh) Byte 1: Code0	
Slave data values	-	
Error conditions	05h (I2C communication fault) 06h (EE checksum fault) 0Dh (Wrong argument) 0Fh (Wrong access code)	

5.1.2.17 Command byte 56h – Read serial number and program version

The command "Read serial number and program version" returns the serial number and firmware information of the slave.

This command returns the following data:

- Serial number, ASCII-coded
- Firmware version, ASCII-coded, followed by a 00h character
- Firmware date, ASCII-coded (format "DD.MM.YY")

Parameter	Value	Note
Command identifier	56h	
Access code	-	
Reaction time	5 ms	
Master data length	0 bytes	
Slave data length	37 bytes	
Master data values	-	
Slave data values	Byte 0 … 8: Serial number in ASCII Byte 9 … 28: Firmware version in ASCII Byte 29 … 36: Firmware date in ASCII	
Error conditions	05h (I2C communication fault)	

5.1.2.18 Command byte 57h – Configure serial interface

The command "Configure serial interface" allows setting the UART parameters of the slave.

This command uses the settings of the parameter interface (UART) with the same coding as for the command "Read type label".

After the successful execution of this command, the slave uses the new UART settings. This is only possible after an encoder reset (see Command byte 53h – Encoder reset [▷ 33]) or after a switch off/switch on cycle.

After switching on, every slave reacts only for a determined time to the UART standard settings. Other settings are only used if this initialization time has elapsed without communication.

So far, the standard parity for slaves has been defined as "odd". The actually implemented standard parity check always corresponded to an "even" parity definition.

Accordingly, this specification corrects this definition.

Parameter	Value	Note
Command identifier	57h	
Access code	Code 0	Default 55h
Reaction time	40 ms	
Master data length	2 bytes	
Slave data length	1 byte	
Master data values	Byte 0: UART settings Bit 0 ... 2: Data rate 000 = 600 bauds 001 = 1200 bauds 010 = 2400 bauds 011 = 4800 bauds 100 = 9600 bauds (standard) 101 = 19200 bauds 110 = 38400 bauds Bit 3: not implemented Bit 4 … 5: Data bits, parity 00 = 8 data bits, no parity 10 = 8 data bits, even (standard setting) 11 = 8 data bits, odd Bit 6: Timeout 0 = 2 * 11 / Data rate 1 = 5 * 11 / Data rate (standard) Bit 7: Bus 0 = Bus slave 1 = Standard slave (standard) Byte 1: Code0	Standard value for byte 0 = E4h The actually available data rates are indicated in the product data sheet from a.
Slave data values	Byte 0: UART settings (as above)	
Error conditions	05h (I2C communication fault) 0Dh (Wrong argument) 0Fh (Wrong access code)	

5.1.2.19 Command byte 6Ah – Set position with synchronization

NOTICE	**Synchronization**
	Since no synchronization is provided for this function, this command must be called when the motor is standing still.

The command "Set position with synchronization" is an optional command.

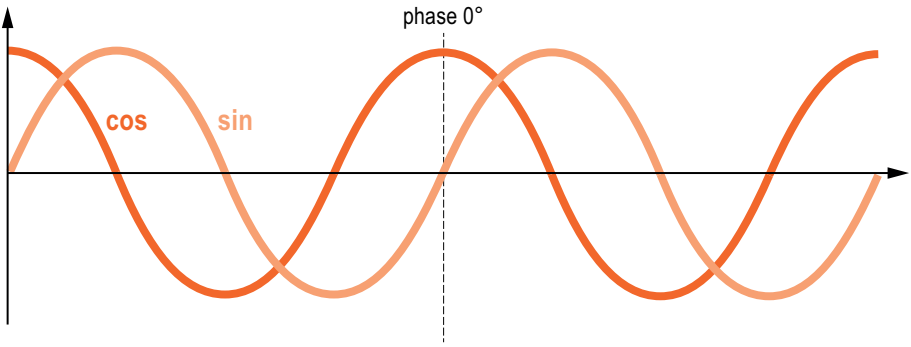
The command "Set position with synchronization" allows the motor controller to save a position offset in the slave, e. g. with the command "Set position".

The difference with this command is that "Set position with synchronization" only allows absolute position offsets in phase with the sin/cos signals. The assigned offset is rounded to this value.

All other aspects of "Set position" also relate to this command.

The master must transmit the required position value (preset). The slave calculates the corresponding position offset, rounds it up to the next suitable value for the zero point offset and saves it internally.

The command "Set position with synchronization" can only be activated when "Code0" is sent, to prevent improper use and impairment of the motor function.



20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

Example for requested preset (5 LSBs of absolute position only)

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Actually used preset with phase shift = 0°

IMG-ID: 68969227

A phase shift different from 0° limits the ability of the master to synchronize the absolute position with analog signals. This is why it can be advantageous to use this command to force a 0° phase shift. In the case of sensors with a low sine/cosine periods number, this can lead to unacceptable commutation offset errors. In this case, we recommend to save the commutation offset in a slave data field and to carry out the offset calculation in the master.

Parameter	Value	Note
Command identifier	6Ah	
Access code	Code 0	Default 55h
Reaction time	40 ms	
Master data length	5 bytes	
Slave data length	4 bytes	
Master data values	Byte 0 … 3: Absolute position set as unsigned 32-bit value, MSB first Byte 4: Code0	Standard value for byte 0 = E4h The actually available data rates are indicated in the product data sheet
Slave data values	Byte 0 … 3: actually used absolute position specification as an unsigned 32-bit value, MSB first	Can be shifted if the requested preset would have led to a phase shift ≠ 0 °
Error conditions	02h (Angle offset fault) 05h (I2C Communication fault) 06h (EE checksum fault) 0Dh (Wrong argument) 0Fh (Wrong access code) 1Dh (LED current high) 20h (Singleturn fault) 21h (Multiturn amplitude fault) 22h (Multi sync fault) 23h (Multiturn vector length fault)	

5.1.3 Status messages

The MFB interface defines error messages as defined in the following sections.

Every slave uses a subset of these messages, as specified in the product data sheet.

Errors are transmitted either with the warning bit or as a status code. This is specified in the following sections.

Error messages can be understood as belonging to different error groups. Error groups group errors according to slave functionality.

Error group	Behavior
Initialization	During slave start-up, the various electronic components are configured with data from an EEPROM. This data is protected against accidental changes by check sums. If the check for position-relevant data fails, the following position commands are carried out. (Reading position or Target position) returns the error code "02h = internal wrong angular offset". If the check fails for other data, the error code "06h = internal CRC error" is issued. An additional error code is issued to define which check sum failed. Errors during initialization affect the proper operation of the slave and lead at a later time to messages belonging to other error groups. If error messages occur during start-up, it is recommended to perform first a "Software reset". If this was not successful, perform a "power cycle".
Protocol	This group includes errors relating to command transmission and analysis. These messages are sent as a response to the command and are not saved in the errors stack. They do not lead to subsequent errors.
Data	This group includes all errors relating to user data saving, which can occur during the execution of a command or during initialization. They are not saved in the errors stack. They do not lead to subsequent errors in other areas.
Position	This group includes all errors that can occur during the reading of the command. Position (42h) or write position (43h). These messages are sent as a response to the command and are not saved in the errors stack. In this case, position measurement is stopped. For more complex error situations involving more than one error, the first error is sent and the other error messages are saved. This is specified by the error bit in the status response.
Others	This group includes warnings or errors caused by the monitoring of temperature values, the LED current or the internal users counter. The error is generated when the value exceeds the limit value. The errors are saved in the errors stack and erased when read.

The following error codes are implemented:

MFB status messages		
	Status code	**Description**
Error type	00h	The encoder detected no error
Initialization	01h	Faulty compensation data
	02h	Faulty internal angular offset
	03h	Data field partitioning table destroyed
	05h	Internal I2C bus not operational
	06h	Internal check sum error
Protocol	0Ah	Check sum of transmitted data is wrong
	0Bh	Unknown command code
	0Ch	Number of transmitted data is wrong
	0Dh	Transmitted command argument is invalid
Data	0Eh	The selected data field may not be written
	0Fh	Wrong access code
	10h	Size of specified data field cannot be modified
	11h	Specified word address is outside of data field
	12h	Access to a nonexistent data field
Others	08h	Counter overflow

5.1.3.1 Status code 0Ah – Check sum error

"Check sum error" indicates that the slave detected a check sum error in a master request message.

Slave reaction:

The message is not analyzed further.

Possible causes:

This can be due to an electrical failure in the line, a wire break or a faulty request of the master.

Parameter	Value
Error message	Error response
Error group	Initialization
Code	0Ah

5.1.3.2 Status code 01h - Faulty compensation data

"Faulty compensation data" indicates that one of the following values in the nonvolatile memory is damaged:

- singleturn resolution and multiturn range settings
- sensor calibration

Slave behavior:

The message is saved together with the error "Internal check sum error" in the errors stack.

The position output is locked when this error is detected. A "Read position" or "Set position" command always receives a slave error response with the wrong internal angle. Possible causes:

This error can be due to electrical disturbances during start-up or to hardware errors of the memory component.

Parameter	Value
Error message	Error response
Error group	Initialization
Code	01h

5.1.3.3 Status code 02h - Faulty internal angular offset

During start, "Faulty internal angular offset" indicates that one of the following values in the non-volatile memory has been damaged:

- Position offset
- Sensor synchronization

Slave reaction:

The message is saved together with the error Internal check sum (06h) in the errors stack.

The position output is locked when this error is detected. A "Read position" or "Set position" command always receives a slave error response with this error code.

Possible causes:

This error can be due to electrical disturbances during start-up or to hardware errors of the memory component.

Parameter	Value
Error message	Error response
Error group	Initialization
Code	02h

5.1.3.4 Status code 03h – Data field partitioning table destroyed

During start, "Data field partitioning table destroyed" indicates that one of the following values in the nonvolatile memory has been damaged:

- UART settings
- User memory size
- Access codes

During operation, "Data field partitioning table destroyed" indicates data corruption in user data fields during data field operations.

Slave reaction:

The message is saved together with the error "Internal check sum error" in the errors stack.

Writing in the nonvolatile memory is locked when this error is detected. Related commands always receive a slave error response with this error code.

Possible causes:

This error can be due to electrical disturbances during start-up or to hardware errors of the memory component.

Parameter	Value
Error message	Error response
Error group	Initialization
Code	03h

5.1.3.5 Status code 05h – Internal I2C bus not operational

"Internal I2C bus not operational" indicates that the slave could not establish a communication with its nonvolatile memory, or detected an error in the communication protocol with its nonvolatile memory.

In addition, "Internal I2C bus not operational" is used during the start when one of the following values has been damages in the nonvolatile memory.

- Encoder type code

Slave reaction:

"Internal I2C bus not operational" is used during operation when an error is detected in the internal check sum (06h) when starting and the user tries to use commands that use the nonvolatile memory of the slave.

Possible causes:

This error can be due to electrical disturbances during start-up or operation, or to hardware errors of the memory component.

Parameter	Value
Error message	Error response
Error group	Initialization
Code	05h

5.1.3.6 Status code 06h – Internal check sum error

"Internal check sum error" indicates that the slave detected a data corruption in its internal configuration.

Normally other error codes are set together with "Internal check sum error" to limit the concerned data. This is documented in "Faulty internal angular offset" (02h).

Slave reaction:

When an "Internal check sum error" is detected during start-up, no command is executed which would read critical data or write data in the nonvolatile memory ("Read counter", "Increment counter", "Reset counter", "Save data", "Create data field", "Define access code", "Define encoder address", "Set serial interface"). This error is displayed if the user tries to use these functions.

Possible causes:

This error can be due to electrical disturbances during start-up or operation, or to hardware errors of the memory component.

Parameter	Value
Error message	Error response
Error group	Initialization
Code	06h

5.1.3.7 Status code 0Bh – Unknown command code

"Unknown command code" indicates that the slave received a master command message with an unknown command code.

Slave reaction:

The message is not analyzed further.

Possible causes:

This can be due to an electrical failure in the line, a wire break or a faulty request of the master.

Parameter	Value
Error message	Error response
Error group	Protocol
Code	0Bh

5.1.3.8 Status code 0Ch – Wrong command length

"Wrong command length" indicates that the slave received a master command message with an invalid number of message bytes.

Slave reaction:

The message is not analyzed further.

Possible causes:

This can be due to an electrical failure in the line, a wire break or a faulty request of the master.

Parameter	Value
Error message	Error response
Error group	Protocol
Code	0Ch

5.1.3.9 Status code 0Dh – Wrong command argument

"Wrong command argument" during operation indicates that the slave received a master command message with an invalid value i one of its command argument bytes.

Slave reaction:

The message is not analyzed further.

Possible causes:

This can be due to an electrical failure in the line, a wire break or a faulty request of the master.

Parameter	Value
Error message	Error response
Error group	Protocol
Code	0Dh

5.1.3.10 Status code 0Eh – The selected data field may not be written

"The selected data field may not be written" indicates that the slave received a request for saving data either in a data field set to "Read only" or in the extended type label.

Slave reaction:

The message is not analyzed further.

Possible causes:

This can be due to an electrical failure in the line, a wire break or a faulty request of the master.

Parameter	Value
Error message	Error response
Error group	Data
Code	0Eh

5.1.3.11 Status code 0Fh – Wrong access code

"Wrong access code" during operation indicates that the slave received a master command message requiring one of the access codes (Code0, Code1, Code2 oder Code3) and that the transmitted access code byte was wrong.

Slave reaction:

The message is not analyzed further.

Possible causes:

This can be due to an electrical failure in the line, a wire break or a faulty request of the master.

Parameter	Value
Error message	Error response
Error group	Data
Code	0Fh

5.1.3.12 Status code 10h – Insufficient memory

"Insufficient memory" during operation indicates one of the following error conditions:

- The slave receiver a "Create data field" request to create a new data field with a size that does not fit in the remaining user memory.
- The slave receiver a "Create data field" request to modify the size of an existing data field with a size that does not fit in the remaining user memory.
- The slave received a request for creating a data field on the extended type label.

If the user attempts to modify the size of another data field than the last created one, the message "Wrong data field number" is returned.

Slave reaction:

The message is not analyzed further.

Possible causes:

This can be due to a faulty master request.

Parameter	Value
Error message	Error response
Error group	Data
Code	10h

5.1.3.13 Status code 11h – Specified word address is outside of data field

Indicates that the slave received a master request message for reading or writing in a data field with an offset outside of the data field size. This message is also used when a read access to the extended type label uses an offset outside of the type label size (64 bytes).

Slave reaction:

The message is not analyzed further.

Possible causes:

This can be due to a faulty master request.

Parameter	Value
Error message	Error response
Error group	Data
Code	11h

5.1.3.14 Status code 12h – Wrong data field number

During start, "Wrong data field number" indicates that one of the following values in the non-volatile memory has been damaged.

- Extended Type Label Specification [▶ 17]

"Wrong data field number" during operation indicates one of the following errors:

Conditions:

- The slave received a data field operation request with an invalid message.
- The slave received the "Create data field" request to create a new data field while the erase bit is set.
- The slave received the "Create data field" request to modify the field size data of a data field which is not the last one.

Slave reaction:

The indication at the start is issued together with the message "Internal check sum error". The position output is locked when this error is detected at the start. A "Read position“ or "Set position" command always receives a slave error response with the internal error "Angular offset" (02h). The message is not analyzed further during operation if this error is detected.

Possible causes:

This error during start-up can be due to electrical disturbances during start-up or to hardware errors of the memory component. The error during operation can be due to a faulty request from the master.

Parameter	Value
Error message	Error response
Error group	Data
Code	12h

5.1.3.15 Status code 08h – Counter overflow

During start-up, "Counter overflow" indicates that the count value of the slave in the nonvolatile memory is corrupted.

During operation, "Count overflow" indicates that the counter of the slave reached the maximum value (FF FF FFh) and that the operator attempts to increase it (see Command byte 47h - Increment counter [▶ 25]). The counter remains stuck at the maximum value until it is reset (see Command byte 49h - Erase counter [▶ 25]).

Slave reaction:

The message is saved together with the error Internal check sum (06h) in the errors stack. The counting operations are locked when this error is detected. A "Read counter", "Increment counter" or "Reset counter" command always receives a slave error response with the error "Internal check sum error" (06h).

Possible causes:

This error can be due to electrical disturbances during start-up or to hardware errors of the memory component.

Parameter	Value
Error message	Error response
Error group	Initialization
Code	08h

5.1.4 Examples

5.1.4.1 Read position

Example for position 0

Request:						
Address	Command	Check sum				
40	42	02				

Response:						
Address	Command	Position				Check sum
40	42	00	00	00	00	02

5.1.4.2 Set position

Example for position 0

Request:							
Address	Com-mand	Position				Code 0	Check sum
40	43	00	00	00	00	55	56

Response:							
Address	Com-mand	Check sum					
40	43	03					

5.1.4.3 Read analog value

Analog value and channel may also vary

Request:					
Address	Command	Channel	Check sum		
40	44	48	4C		

Response:					
Address	Command	Channel	High (Value)	Low (Value)	Check sum
40	44	48	00	25	69

5.1.4.4 Increment counter

Request:		
Address	Command	Check sum
40	47	07

Response:		
Address	Command	Check sum
40	47	07

5.1.4.5 Read counter

Count may also vary

Request:					
Address	Command	Check sum			
40	46	06			

Response:					
Address	Command	Counter High	Counter Middle	Counter Low	Check sum
40	46	00	00	00	06

5.1.4.6 Erase counter

Request:			
Address	Command	Code 0	Check sum
40	49	55	5C

Response:			
Address	Command	Check sum	
40	49	09	

5.1.4.7 Read data

Request:																		
Address	Command	Data field	Byte address	Count	Code 0	Check sum												
40	4A	00	00	10	55	4F												
		Varies																

Response:																		
Address	Command	Data field	Byte address	Count	Data in suitable length													Check sum
40	4A	00	00	10	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1A
		Varies																

5.1.4.8 Save data

Request:												
Address	Command	Data field	Byte address	Count	Code0	Data						Check sum
40	4B	00	00	08	55h	FF	00	FF	00	FF	00	FF	00	56
		Varies				Varies						

Response:														
Address	Com-mand	Data field	Byte address	Count	Check sum									
40	4B	00	00	08	03									
		Varies												

5.1.4.9 Determine the status of a data field

Request:				
Address	Command	Data field	Check sum	
40	4C	00	0C	
		Varies		

Response:				
Address	Command	Data field	Status Byte	Check sum
40	4C	00	C8	1E
		Varies		

Status Byte

Bit 0..2	Data field size
Bit 3	Code disable
Bit 4..5	Access code
Bit 6	Read-write access
Bit 7	Delete create data field

Complete description see Status Byte [▶ 19].

5.1.4.10 Create / erase data field

Request:					
Address	Command	Data field	Status Byte	Code 0	Check sum
40	4D	00	C8	55	90
		Vary			

Response:					
Address	Command	Data field	Status Byte	Check sum	
40	4D	00	C8	C5	
		Vary			

Status Byte

Bit 0..2	Data field size
Bit 3	Code disable
Bit 4..5	Access code
Bit 6	Read-write access
Bit 7	Delete create data field

Complete description see Status Byte [▶ 19].

5.1.4.11 Erase available memory area

Request:				
Address	Command	Check sum		
40	4E	0E		

Response:				
Address	Command	Free memory	Number of data fields	Check sum
40	4E	16	00	18

5.1.4.12 Read encoder status

Request:			
Address	Command	Check sum	
40	50	10	

Response:			
Address	Command	Encoder status	Check sum
40	50	00	10

5.1.4.13 Modify access key

Request:					
Address	Command	Code number	Old code	New code	Check sum
40	4F	02	55	66	3E

Response:					
Address	Command	Code number	Check sum		
40	4F	02	0D		

5.1.4.14 Read out type label

Request:						
Address	Command	Check sum				
40	52	12				

Response:						
Address	Com-mand	UART settings	Encoder type	Memory size	Option code	Check sum
40	52	E4	22	16	00	C2

UART settings

Bit 0..2	Baud rate
Bit 3	no meaning
Bit 4..5	Parity
Bit 6	Timeout
Bit 7	Bus

Complete description see RS485 Settings [▶ 20].

5.1.4.15 Encoder reset

Request:		
Address	Command	Check sum
40	53	13

NOTICE	**Response**
	There is no response to this command.

5.1.4.16 Assign encoder address

Request:				
Address	Command	New address	Code 0	Check sum
40	55	41	55	01

Response:				
Address	Command	Check sum		
41	55	14		

5.1.4.17 Read serial number / program version

Request:

Address	Command	Check sum
40	56	16

Response:

Address	Command	Serial number									Firmware version																				Firmware date								Check sum
40	56	31	32	33	34	35	36	37	38	39	76	31	2E	30	2E	38	00	00	00	00	00	00	00	00	00	00	00	00	00	00	31	32	30	37	32	30	31	39	66

5.1.4.18 Configure serial interface

Request:

Address	Command	UART settings	Code 0	Check sum
40	57	E4	55	5C

Response:

Address	Command	UART settings	Check sum	
40	57	E4	F3	

UART settings

Bit 0..2	Baud rate
Bit 3	no meaning
Bit 4..5	Parity
Bit 6	Timeout
Bit 7	Bus

See RS485 Settings [▸ 20]

5.1.4.19 Set position with synchronization

Request:

Address	Command	Absolute position				Code 0	Check sum
40	6A	00	05	D2	C7	55	6F

Response:							
Address	Com-mand	Absolute position				Check sum	
40	6A	00	05	D2	C1	3C	

5.2 BiSS

5.2.1 Data Transmission

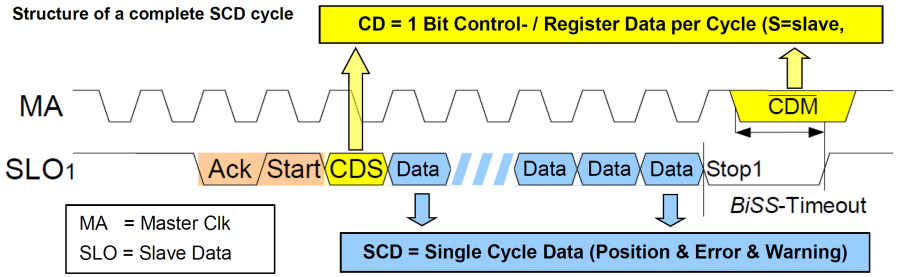
5.2.1.1 Frames Structure

Source: Protocol Description (BISS C)

The BiSS C-Mode bus protocol allows simultaneous transmission of sensor data (SD) from the slaves to the master and of master and control data (CD) from this master to the slaves.

The BiSS-Frame (transmission frame) is started, clocked and completed by the master with the MA clock. The first rising MA edge is used for slave synchronization. It allows the isochronous scanning of sensor data. With he second rising MA edge, the slaves set their SLO line to "0" and thus generate their "Ack" signal (Acknowledge); it remains active (SLO = "0"), until the start bit arrives, followed by the CDS-Bit (Control Data Sensor). Starting with the 2nd bit after the start bit and up to the stop bit of the BiSS frame comes the data area, which transmits the sensor data from the slaves to the master.

During this period no further cycles are sent from the master to the MA.



IMG-ID: 78140811

- The CDM bit is transmitted inverted with the MA clock.
- The CDS bit is the response of a slave and this response is transmitted during the following cycle to the master.
- Every SCD cycle ends with a BiSS timeout.

In every Biss frame, a bit of control data (CD) is transmitted per direction for the command or the register communication.

The control data bit of the master (CDM, Control Data Master) is sent to the slaves via the MA line as an inverted signal level of the BiSS timeout. The addressed slave responds with the CDS bit (Control Data Slave), which is always transmitted in the first bit after the start bit.

The control data bits of several consecutive Biss frames (cycles) are combined together to a control frame by the master and the slaves (see illustration below). It allows reading and writing the slave register and and sending commands to the slaves.

Bus reset

After switching on or after an error, the master must maintain a break of 40 µs prior to data transmission. This ensures that the BiSS timeout has expired and that the slaves are ready for data transmission.

Single cycle data (SCD)

A data channel with single cycle data is used for fast and cyclic sensor data entirely transmitted in a cycle. SCDs do not require addressing and have a parameterizable length of 1 to 64 data bits and a CRC check of 0 to 8 bits.

The data value

All data values are transmitted with the most significant bit first ("MSB first"). A data value can consist of several bit groups, e.g. position and several error flags.

CRC check

Every data channel (slave or sensor) can use a transmission check with CRC in addition to its data value. The features of the CRC check are defined in the sensor parameters. The CRC polynomial also indicates the transmitted CRC bits; 0 to 8 bits are possible.

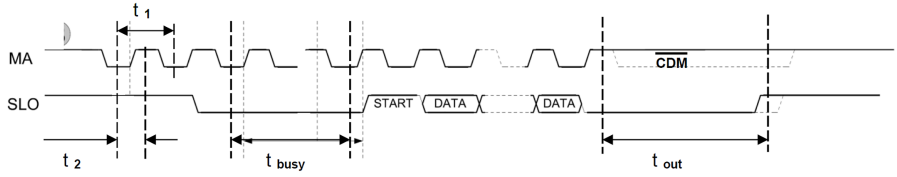
The CRC check bits are always transmitted inversely first with the most significant bit.

In general, the starting value for the CRC calculation is zero.

5.2.1.2 Data Addresses

Timing and data content of a "Point-to-Point" communication cycle:

Please read carefully the BiSS-C interface protocol description, especially the Point-to-Point link type, and then implement this interface protocol in your master (controller). BiSS C Protocol Description



BiSS-C timing (Point-to-point configuration)

IMG-ID: 78158603

MA	Master Clk
SLO	Slave data
t 1	Min. 100ns (max. clk frequency = 10 MHz) / max. 13µs (min. clk frequency = 77 kHz)
t 2	typ. 50% of t1
t out	13µs
Start	Start bit
Data	Data string according to the list: data length and description
CDM	Control Data Master

Data length and description

NOTICE	**Frame description**
	The general frame is not described here. Please note the general BiSS C Protocol Description

CDS	MT	ST	N / ERR	N / Warn	LC	CRCPOS
(1 bit)	(0-24bits)	(0-23bits)	(1 bit)	(1 bit)	(6 bits)	(6 bits)

Control Data Slave (CDS)	One bit per cycle, used for register communication	1 bits
Control Data Master (CDM)	One bit per cycle, used for register communication, (standard logic level = low)	1 bits
Multiturn data bits (MT)	Position information "Number of revolutions"	0 … 24 bits
Singleturn data bits (ST)	Position information "Resolution / 360° "	0 … 23 bits
Status bit (N / ERR)	Error bit (low active) details can be found in the data sheet of the measuring device.	1 bits
Status bit (N / Warning)	Warning bit (low active) details can be found in the data sheet of the encoder.	1 bits
Life counter (LC)	This 6-bit counter is incremented at every new communication cycle. • This is to demonstrate that the encoder is "alive". • Exception: The value '0' only exists once at system start. • The counter subsequently skips this value.	6 bits
CRC (CRCPOS)	Every data transmission is checked by a CRC (Cyclic Redundancy Check). The data includes • the multiturn data bits • the singleturn data bits • the status bits and the Life counter (LC) • The CRC polynomial is x6 +x1 +x0 or "1000011" (0x43) • The CRC starting value is "0" (0x00) • The CRC calculation can be reproduced with on-line calculators. • Refer to On-line CTC calculator	6 bits

5.2.1.3 Structure of the EDS

General section

The BiSS EDS (electronic data sheet) describes the features and operating conditions of a BiSS device and contains information on the process data and parameters. The BiSS EDS is available for every device.

With the Kübler absolute encoders, the EDS starts on bank 3 (defined in bank 0 address 0x41) and consists of 2 parts (common part and BP3 part), each part consisting of one bank.

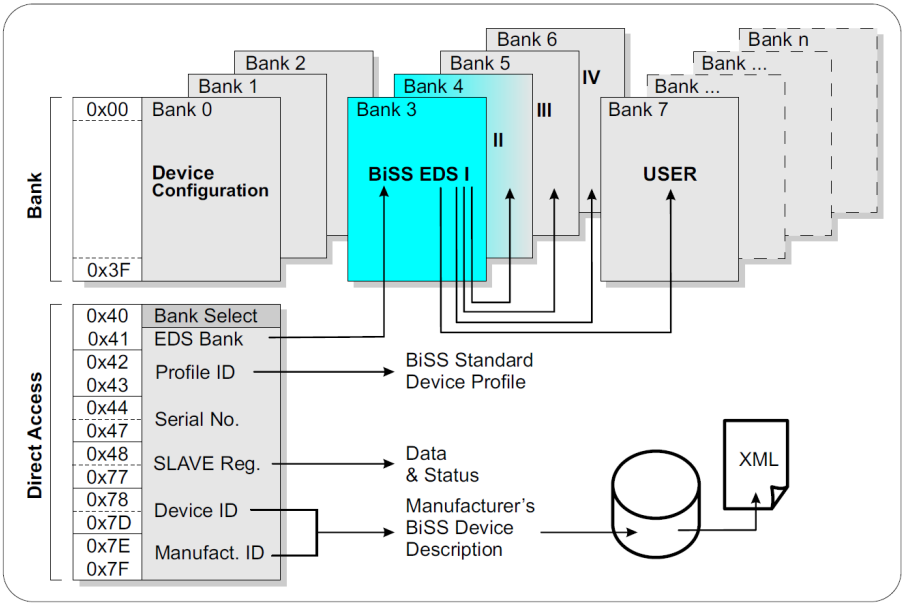
The first part (common part) contains non process-relevant information such as parameters for the master, but no process data. This includes clock frequency, timeout, delay timings, etc.

Encoder-specific part BP3 (standard encoder protocol)

The specific EDS for the encoder can be found under: https://www.kuebler.com

The second part (BP3 part) begins in bank 4 and contains the necessary information for the transmission, the product and process-relevant information for the motion control system.

Bank 5 and bank 6 are reserved. Bank 7 to bank 31 are planned as memory area for user data.



IMG-ID: 78220811

Manufacturer identification

Every BiSS device manufacturer has a manufacturer identification for all BiSS devices. It is composed of the device ID and of the manufacturer ID.

NOTICE	**Kübler manufacturer ID**
	Kübler's manufacturer ID is: **4B 55**

Every BiSS device of a same manufacturer has in addition an own device ID. Both identifications are saved in address areas provided for this purpose in bank 0:

- Device-ID → Addr. 0x78 … 0x7D.
- Kübler Manufact. ID → Addr. 0x7E, 0x7F

Device ID structure:

	Device ID																								
Addr.	**0x78**								**0x79**								**0x7A**								
Bit No.	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	
Series	**Device class**								**Device subclass**								BiSS-MOD	**RegC**	**RegB**	**MT (4:0)**					

	Device ID																								
Addr.	**0x78**								**0x79**								**0x7A**								
Bit No.	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	
Series	**Device class**								**Device subclass**								BiSS-MOD	**RegC**	**RegB**	**MT (4:0)**					

	Device ID																								
Addr.	**0x7B**								**0x7C**								**0x7D**								
Bit No.	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Area	**Reserved**			**ST (4:0)**					**Reserved**				**Timeout**				**Device revision ZZ**								

Manufacturer ID structure:

	Manufacturer ID															
Addr.	**0x7E**								**0x7F**							
Bit No.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Hex. Value	4B								55							

Together, they form the unique identification of every device type as a hex. code. Example for a S3674 with 21-bit ST resolution:

	Device ID	Manufacturer ID
S3674, 21 bits	04 00 C0 15 00 00	4B 55

5.2.1.3.1 Common Part

Addr.	Symbol	Description	Group	Format	Unit	Value
0x00	EDS_VER	EDS version (continuous number)	Orga	U8	-	0x01
0x01	EDS_LEN	EDS length (bank count completely)	Orga	U8	Banks	0x02
0x02	USR_STA	Bank address USER start (bank selection in address 64, 255= not available)	Orga	U8	-	0x07
0x03	USR_END	Bank address USER end (bank selection address 64)	Orga	U8	-	0x1F
0x04	TMA	Minimum permitted clock period on MA (TMA)	Timing	U8	1 ns	0x64
0x05	TO_MIN	Minimum *BiSS* timeout (0= adaptive)	Timing	U8	250 ns	0x28
0x06	TO_MAX	Maximum *BiSS* timeout (0= adaptive)	Timing	U8	250 ns	0x44
0x07	TOS_MIN	Minimum *BiSS* timeout_S (0= adaptive)	Timing	U8	25 ns	0x28
0x08	TOS_MAX	Maximum *BiSS* timeout_S (0= adaptive)	Timing	U8	25 ns	0x44
0x09	TCLK_MIN	Minimum sampling periode adaptive timeout (0= adaptives timeout not available)	Timing	U8	25 ns	0x00

0x0A	TCLK_MAX	Maximum sampling periode adaptive timeout (0= adaptives timeout not available)	Timing	U8	25 ns	0x00
0x0B	TCYC	Minimum cycle time (0= no limitation)	Timing	U8	250 ns	0x00
0x0C	TBUSY_S	Maximum processing time SCD	Timing	U8	250 ns	0x16
0x0D	BUSY_S	Additional processing time SCD in clocks	Timing	U8	TMA	0
0x0E	PON_DLY	Maximum "power on delay" until control communication is available	Timing	U16 1		0x00
0x0F	PON_DLY	Maximum "power on delay" until control communication is available			1 ms	0x64
0x10	DC_NUM	Number of data channel in this device (number of words)	SCD	U8	-	0x01
0x11	SL_NUM	Area of validity for this EDS (number of slave addresses)	SCD	U8	-	0x01
0x12	SL_OFF	Memory location for this EDS (slave ID within this device)	SCD	U8	-	0x00
0x13		Reserved				0x00
0x14	BANK1	Bank address for content description data channel 1 (profile EDS)	SCD	U8	-	0x04
0x15	DLEN1	Data length data channel 1	SCD	U8	bit	0x.
0x16	FORMAT1	Data format data channel 1	SCD	U8	bit	0x0A
0x17	CPOLY1	CRC polynomial (8:1) for data channel 1	SCD	U8	-	0x21

5.2.1.3.2 BP3 Part

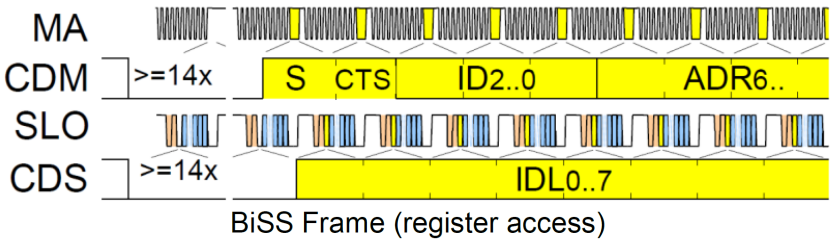
Adr.	Symbol	Description	Group	Format	Unit	Value
0x00	BP_VER	BiSS Profile 3 Version	Orga	U8	-	0x01
0x01	BP_LEN	Length of this profile	Orga	U8	Banks	0x01
0x02	BP_ID	Profile identification BP3 (content also available in adress 0x42 and 0x43)	Orga	U8	Non-safety application	0x62
0x03				U8	Numb. of Pos.bits + nE + nW	0x.
0x04	FB1	Feedback bit 1	Orga	U8	Errorbit	0x01
0x05	FB2	Feedback bit 2	Orga	U8	Warningbit	0x02
0x06	PON_PDL	Maximum "power on delay" until position data are available	Timing	U8	100ms	0x64
0x07		Reserved	Prot	U8	-	0x00
0x08	EN_TYP	Encoder type (linear oder rotary)	Orga	U8	Rotary encoder	0x00
0x09	POS_NUM	Position value	Safety	U8	Pos value 1	0x01
0x0A	MT_LEN	Data length MULTITURN	Orga	U8	Number of MT-bits	0x00
0x0B	MT_FMT	Data format MULTITURN	Meas	U8	Left-aligned	0x00

0x0C	CO_LEN	Data length COARSE	Orga	U8	Number of digital bits	0x0B
0x0D	CO_FMT	Data format COARSE	Meas	U8	Left-aligned	0x01
0x0E	FI_LEN	Data length FINE	Orga	U8	Number of Int-bits	0x.
0x0F	FI_FMT	Data format FINE	Meas	U8	Left-aligned	0x01
0x10	MT_CNT	Number of distinguishable revolutions/periods	Meas	U32 1	00 for Singleturn encoders	0x00
0x11						0x00
0x12						0x00
0x13						0x00
0x14	SIP_CNT	Number of signal periods per revolution/length of signal periode	Meas	U32 1	2048 Per.	0x00
0x15						0x00
0x16						0x08
0x17						0x00
0x18	SIP_RES	Resolution factor per signal period (LSB of the interpolation)	Mess	U32 1	Number of interpolator LSBs	0x00
0x19						0x00
0x1A						0x.
0x1B						0x.
0x1C	CPOLY	CRC polynomial (32:1) 3	Orga	U32 1	-	0x00
0x1D						0x00
0x1E						0x00
0x1F						0x21
0x20	CSTART	CRC start value	Orga	U32 1	Default 00	0x00
0x21						0x00
0x22						0x00
0x23						0x00
0x24	ABS_ACU	Absolute accuracy	Meas	U16 1	n.a.	0x00
0x25					n.a.	0x00
0x26	REL_ACU	Repeat accuracy	Meas	U16 1	n.a.	0x00
0x27						0x00
0x28	SPD_ACU	Angular speed / speed-depending accuracy	Meas	U16 1	n.a.	0x00
0x29						0x00
0x2A	HYST	Hysteresis	Meas	U16 1	n.a.	0x00
0x2B						0x00
0x2C	SPD_MAX	Maximum rotational speed / maximum speed	Mech	U16 1	n.a.	0x00
0x2D						0x00
0x2E	ACC_MAX	Maximum angular acceleration / maximum acceleration	Mech	U16 1	n.a.	0x00
0x2F						0x00

0x30	TMP_MIN	Minimum operating temperature	Mech	U16 1	n.a.	0x00
0x31						0x00
0x32	TMP_MAX	Maximum operating temperature	Mech	U16 1	n.a.	0x00
0x33						0x00
0x34	VLT_MIN	Minimum operating voltage	Elec	U16 1	n.a.	0x00
0x35						0x00
0x36	VLT_MAX	Maximum operating voltage	Elec	U16 1	n.a.	0x00
0x37						0x00
0x38	CUR_MAX	Maximum current consumption	Elec	U16 1	n.a.	0x00
0x39						0x00
0x3A		Reserved	Prot	U8	-	0x00
0x3B		Reserved	Prot	U8	-	0x00
0x3C		Reserved	Prot	U8	-	0x00
0x3D		Reserved	Prot	U8	-	0x00
0x3E		Reserved	Prot	U8	-	0x00
0x3F	CHKSUM	Checksum (addition of all bytes in this bank)	Orga	U8		

5.2.2 Register access

Structure of a starting single register access:



BiSS Frame (register access)

IMG-ID: 78164363

BiSS Frame - Register access

- At least 14 SCD cycles with "CDM = 0" are necessary to interrupt a command frame possibly started previously.
- "START = 1" indicates the start of a control communication to the slaves.
- "CTS = 1" indicates a register access to the slaves.
- CDS uses the "IDL" bits to tell the master how many of the possible 8 BiSS IDs are used. Only one for Point-to-Point communication.

Control frame

The control frame allows the protected and confirmed reading and writing of the register of a slave and the protected and confirmed sending of commands to the slaves. The control frame results from a number of BiSS frames which generate and transmit sensor data.

The register access or the command always takes place at the end of the cycle of the last COM bit, i. e. with the expiry of the BiSS timeout in the slave. The control frame can be interrupted at any time by transmitting 14 "0" bits. At least 14 cycles with CDM = "0" must precede the start bit of a control frame.

NOTICE	**Control frame**
	At least 14 bits with CDM = "0" must be transmitted before every control frame.

Slave addressing with ID assignment

Unlike the sensor data communication, the control communication requires clear addressing.

CRC check

The control communication also uses a check sum for transmission check.

The used CRC polynomial is: X4 + X1 + X1 + X0.

This makes 4 CRC bits available, which are transmitted inverted. Calculation takes place with the starting value zero via the addressing sequence or the data bits start with the most significant bit and always exclude the start bit.

The register communication

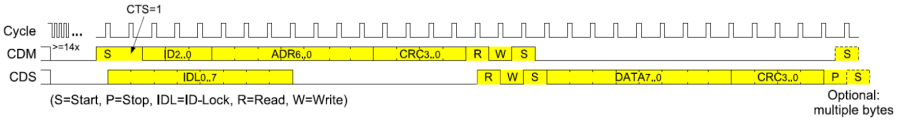
The read and write accesses to slave registers occur through a control frame with protected CTS control selection bit (CTS = "1": register access).

The register frame starts with the addressing sequence. Here the master sends the slave ID (3 bits) followed by the register address (7 bits) and a 4-bit CRC. The binary coding thus allows addressing 8 slaves with 128 registers (= 128 bytes). In the point-to-point configuration with one sensor (slave), the ID is always "000" binary.

The two following CDM bits, the R and the W bit, determine whether a read access (RW = "10") or a write access (RW = "01") is to take place. Both bits must be reversed with respect to each other, and they are sent back from the master to the slave for confirmation. They are not included in the CRC calculation.

5.2.2.1 Read access

During the read access, both read/write bits have the value RW = "10". Then comes a start bit, 12 "0" bits and a stop bit. The register data are protected during reading with a 4-bit CRC.

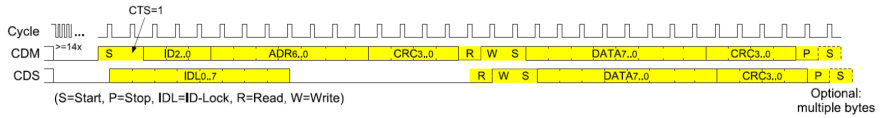


IMG-ID: 78184459

Position (output data)		
0x61	ST(1:0)	-
0x62	ST(9:2)	
0x63	ST(17:10)	
0x64	ST(25:18)	
0x65	MT(7:0)	
0x66	MT(15:8)	
0x67	MT(23:16)	

5.2.2.2 Write access

During the write access, both read/write bits have the value RW = "01". Then comes a start bit, 8 data bits, a 4-bit CRC and a stop bit. The 8 data bits are protected during writing by a 4-bit CRC, and the transmitted register data is returned.



IMG-ID: 78188299

Position correction with offset

An offset for singleturn (OFFS_ST) and multiturn (OFFS_MT) data can be defined to adjust the absolute position data to a mechanical position. These offsets are deducted from the existing data. The values are saved in the following registers.

OFFS_ST	
Addr. 0x1E; bit 7...0 R/W Addr. 0x1D; bit 7...0 Addr. 0x1C; bit 7...0 Addr. 0x1B; bit 7...6	
Code	Function
0x00 0x3FFFFFF	Offset value

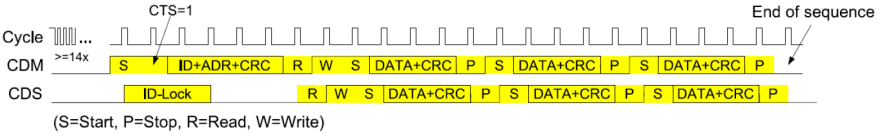
OFFS_MT	
Addr. 0x21; bit 7...0 R/W Addr. 0x20; bit 7...0 Addr. 0x1F; bit 7...0	
Code	Function
0x00 0xFFFFFF	Offset value

Offset & interpolator						
0x1B	OFFS_ST(1:0)	0	0	0	DIR	RESO_CC(1:0)
0x1C	OFFS_ST(9:2)					
0x1D	OFFS_ST(17:10)					
0x1E	OFFS_ST(25:18)					
0x1F	OFFS_MT(7:0)					
0x20	OFFS_MT(15:8)					
0x21	OFFS_MT(23:16)					

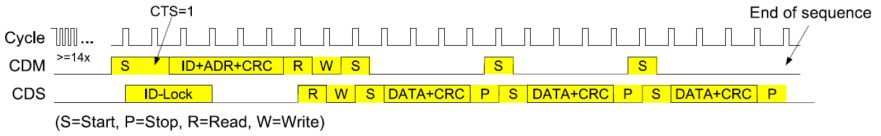
5.2.2.3 Sequential register access

It is possible to read or write several consecutive registers in the same access. To do so, the master sends an additional start bit (CDM = "1") immediately after the stop bit of the first data value. During a write access, the data byte is followed by the 4 CNC check bits and the stop bit. During a read access, only 13 "0" bits - including a stop bit - are sent.

The slave increments internally the register address at every read or write access by 1 (auto-increment). The at the maximum 64 registers can be read or written successively in one access. Sequential accesses exceeding register address 63 (0x3F) or 127 (0x7f) are not allowed. The sequential access ends when no other start bit follows in the CDM.



Writing several registers

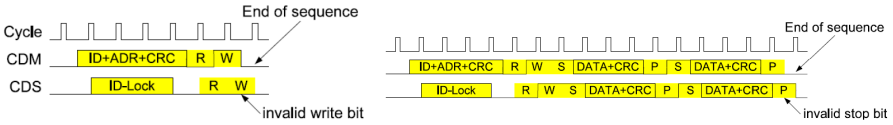


Reading several register

IMG-ID: 78186379

5.2.2.4 Not implemented registers

The registers of a BiSS-C mode slave can be "forbidden" or "not implemented". In this case, the slave reaches the access to the register by inverting the W bit returned by the CDS. For a write access, this results in RW = "00, and in RW = "11" for a read access. If several registers are written or read successively and the following register is not implemented or cannot be addressed in auto-increment, the stop bit is inverted via CDS, i. e. becomes bit "1". The access to a not implemented register ends the sequential access.

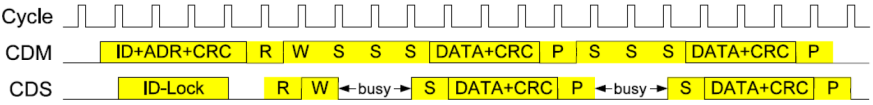


Accesses to non-implemented registers (here Write)
a) First register b) Additional register of a write sequence

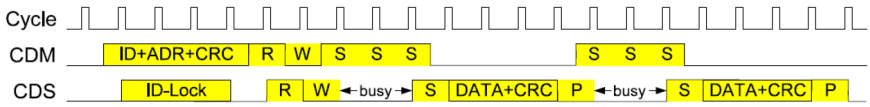
IMG-ID: 78180619

5.2.2.5 Register access processing time

If the slave requires additional processing time for reading or writing registers, it can request this individually for every byte be delaying the start bit. During this time, the master repeats the start bit. If the start bit is not transmitted within tbusy_r, the master interrupts the repetition of the start bit, which leads to consider the register access invalid and erase it. As a general rule, a register access requires processing time when an external memory is addressed. If processing time is required after the last transmitted register, e.g. to save the value, this cannot be signaled via BiSS. However, the slave can request the required processing time at the beginning of the following register access. The maximum processing time for register access is tbusy_r.



Write access with delay time



Read access with delay time

IMG-ID: 78182539

NOTICE	**CDS bit**
	The CDS bit is transmitted in the BiSS frame before the CDM bit. During register access, the master must evaluate immediately the start bit received via CDS and respond in the same BiSS frame with the CDM bit This can be either "1" if processing time is requested / must be send from the MSB, or "0". If no evaluation is possible at the end of the frame, the CDM bit can be send depending on the CDS bit (e.g. inverted).

5.2.3 Status messages

NOTICE	**Warnings - status messages**
	Warnings are disabled as a standard.

Error messages

NOTICE	**Error status**
	The error messages are automatically reset as soon as the error is resolved.

The error register has address 0x69 and has a one-byte size (8 bits).

The following messages are available:

Bit	Error reporting	Note	Value
0	Control Error	0 = no error 1 = current and amplitude values are exceeded	1
1	Signal Error	Amplitudes or offset values are exceeded 0 = no error 1 = outside of control range (e.g. coded disk broken, light source aging, sensor error due to mechanical influences)	2
2	Temperature Error	Temperature excess 1 = temperature within the specified range (-40 °C ... +120 °C) 0 = temperature outside the specified range	4
3	Synchronization Error	0 = no error 1 = synchronization error	8
4	Configuration Error	0 = no error 1 = error in the EEPROM	16
5	Interpolation Error	0 = no error 1 = internal sensor error	32
6	Absolute Data Error	0 = no error 1 = wrong data word	34
7	System Error	0 = no error 1 = general system error	128

NOTICE	**Temperature Error**
	Temperature monitoring is disabled as a standard.

All values can appear combined. If the value of the error register is = 3, there are both a Signal and a Control Error (bit 0 + bit 1).

6 Maintenance

In harsh environments, we recommend regular inspections for firm seating and possible damages at the device. Repair work may only be carried out by the manufacturer, see chapter Contact [▷ 72].

Prior to the work

- Switch off the power supply and secure it against switching on again.
- Then disconnect the power supply lines physically.
- Remove operating and auxiliary materials and remaining processing materials from the measuring system.

7 Annex

7.1 Decimal / Hexadecimal conversion table

Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex
0	*0*	51	*33*	102	*66*	153	*99*	204	*CC*
1	*1*	52	*34*	103	*67*	154	*9A*	205	*CD*
2	*2*	53	*35*	104	*68*	155	*9B*	206	*CE*
3	*3*	54	*36*	105	*69*	156	*9C*	207	*CF*
4	*4*	55	*37*	106	*6A*	157	*9D*	208	*D0*
5	*5*	56	*38*	107	*6B*	158	*9E*	209	*D1*
6	*6*	57	*39*	108	*6C*	159	*9F*	210	*D2*
7	*7*	58	*3A*	109	*6D*	160	*A0*	211	*D3*
8	*8*	59	*3B*	110	*6E*	161	*A1*	212	*D4*
9	*9*	60	*3C*	111	*6F*	162	*A2*	213	*D5*
10	*0A*	61	*3D*	112	*70*	163	*A3*	214	*D6*
11	*0B*	62	*3E*	113	*71*	164	*A4*	215	*D7*
12	*0C*	63	*3F*	114	*72*	165	*A5*	216	*D8*
13	*0D*	64	*40*	115	*73*	166	*A6*	217	*D9*
14	*0E*	65	*41*	116	*74*	167	*A7*	218	*DA*
15	*0F*	66	*42*	117	*75*	168	*A8*	219	*DB*
16	*10*	67	*43*	118	*76*	169	*A9*	220	*DC*
17	*11*	68	*44*	119	*77*	170	*AA*	221	*DD*
18	*12*	69	*45*	120	*78*	171	*AB*	222	*DE*
19	*13*	70	*46*	121	*79*	172	*AC*	223	*DF*
20	*14*	71	*47*	122	*7A*	173	*AD*	224	*E0*
21	*15*	72	*48*	123	*7B*	174	*AE*	225	*E1*
22	*16*	73	*49*	124	*7C*	175	*AF*	226	*E2*
23	*17*	74	*4A*	125	*7D*	176	*B0*	227	*E3*
24	*18*	75	*4B*	126	*7E*	177	*B1*	228	*E4*
25	*19*	76	*4C*	127	*7F*	178	*B2*	229	*E5*
26	*1A*	77	*4D*	128	*80*	179	*B3*	230	*E6*
27	*1B*	78	*4E*	129	*81*	180	*B4*	231	*E7*
28	*1C*	79	*4F*	130	*82*	181	*B5*	232	*E8*
29	*1D*	80	*50*	131	*83*	182	*B6*	233	*E9*
30	*1E*	81	*51*	132	*84*	183	*B7*	234	*EA*

Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex
31	*1F*	82	*52*	133	*85*	184	*B8*	235	*EB*
32	*20*	83	*53*	134	*86*	185	*B9*	236	*EC*
33	*21*	84	*54*	135	*87*	186	*BA*	237	*ED*
34	*22*	85	*55*	136	*88*	187	*BB*	238	*EE*
35	*23*	86	*56*	137	*89*	188	*BC*	239	*EF*
36	*24*	87	*57*	138	*8A*	189	*BD*	240	*F0*
37	*25*	88	*58*	139	*8B*	190	*BE*	241	*F1*
38	*26*	89	*59*	140	*8C*	191	*BF*	242	*F2*
39	*27*	90	*5A*	141	*8D*	192	*C0*	243	*F3*
40	*28*	91	*5B*	142	*8E*	193	*C1*	244	*F4*
41	*29*	92	*5C*	143	*8F*	194	*C2*	245	*F5*
42	*2A*	93	*5D*	144	*90*	195	*C3*	246	*F6*
43	*2B*	94	*5E*	145	*91*	196	*C4*	247	*F7*
44	*2C*	95	*5F*	146	*92*	197	*C5*	248	*F8*
45	*2D*	96	*60*	147	*93*	198	*C6*	249	*F9*
46	*2E*	97	*61*	148	*94*	199	*C7*	250	*FA*
47	*2F*	98	*62*	149	*95*	200	*C8*	251	*FB*
48	*30*	99	*63*	150	*96*	201	*C9*	252	*FC*
49	*31*	100	*64*	151	*97*	202	*CA*	253	*FD*
50	*32*	101	*65*	152	*98*	203	*CB*	254	*FE*
								255	*FF*

8 Contact

You want to contact us:

Technical advice

Kübler's worldwide applications team is available on site all over the world for technical advice, analysis or installation support.

International support (English-speaking)

+49 7720 3903 952

support@kuebler.com

Kübler Germany +49 7720 3903 849

Kübler France +33 3 89 53 45 45

Kübler Italy +39 0 26 42 33 45

Kübler Poland +48 6 18 49 99 02

Kübler Turkey +90 216 999 9791

Kübler China +86 10 8471 0818

Kübler India +91 8600 147 280

Kübler USA +1 855 583 2537

Repair service / RMA form

In case of returns, please package the product sufficiently and attach the completed "Returns form".

www.kuebler.com/rma

Please send your return to the address below.

Kübler Group

Fritz Kübler GmbH

Schubertstraße 47

D-78054 Villingen-Schwenningen

Germany

Phone +49 7720 3903 0

Fax +49 7720 21564

info@kuebler.com

www.kuebler.com

Glossary

ASCII

American Standard Code for Information Interchange. 7-bit coding

BiSS

Bidirectional / Serial / Synchronous

ccw

counterclockwise, counting direction

CRC

Cyclic Redundancy Check

cw

clockwise, counting direction

EEPROM

Electrically erasable programmable read-only memory. Nonvolatile electronic memory elements whose saved information can be erased electrically.

EIA

Electronic Industries Alliance

EN 60204-1

Safety of machinery - Electrical equipment of machines - Part 1 General requirements

HEX

Hexadecimal

IEEE 802.3

RS485 complies with standard EIA485 according to IEEE 802.3

ISO 2768-1

General tolerances, tolerances for linear and angular dimensions without individual tolerance indications

LSB

Least Significant Bit

LVDS

Low Voltage Differential Signaling. Interface standard for high-speed data transmission

MSB

Most Significant Bit

RMA

Return Material Authorization (complaint)

Sync

Synchronization

UART

Universal Asynchronous Receiver Transmitter. Electronic circuit used for realizing digital serial interfaces

Standards

ASCII

American Standard Code for Information Interchange. 7-bit coding

BiSS

Bidirectional / Serial / Synchronous

ccw

counterclockwise, counting direction

CRC

Cyclic Redundancy Check

cw

clockwise, counting direction

EEPROM

Electrically erasable programmable read-only memory. Nonvolatile electronic memory elements whose saved information can be erased electrically.

EIA

Electronic Industries Alliance

EN 60204-1

Safety of machinery - Electrical equipment of machines - Part 1 General requirements

HEX

Hexadecimal

IEEE 802.3

RS485 complies with standard EIA485 according to IEEE 802.3

ISO 2768-1

General tolerances, tolerances for linear and angular dimensions without individual tolerance indications

LSB

Least Significant Bit

LVDS

Low Voltage Differential Signaling. Interface standard for high-speed data transmission

MSB

Most Significant Bit

RMA

Return Material Authorization (complaint)

Sync

Synchronization

UART

Universal Asynchronous Receiver Transmitter. Electronic circuit used for realizing digital serial interfaces

Kübler Group
Fritz Kübler GmbH
Schubertstr. 47
D-78054 Villingen-Schwenningen
Germany
Phone +49 7720 3903-0
Fax +49 7720 21564
info@kuebler.com
www.kuebler.com