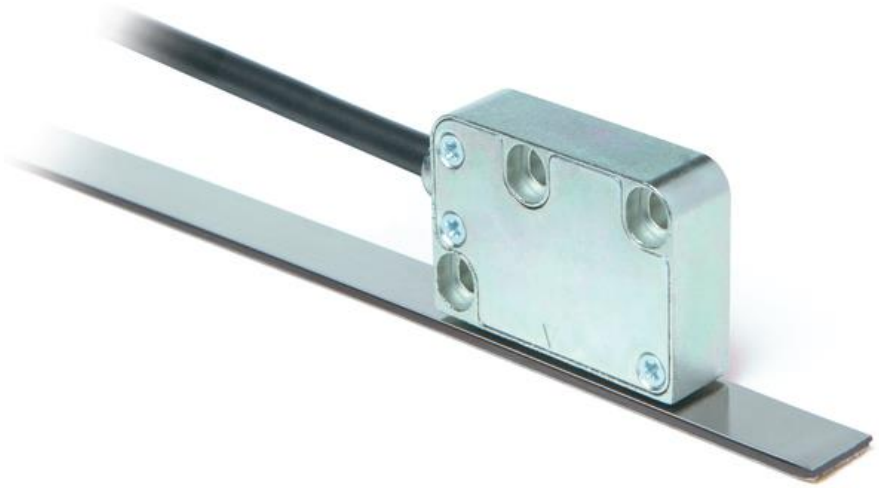
MSK5000CAN

Incremental magnetic encoder with CANopen interface

User manual



CANopen®

SIKO

152/23

Table of contents

General Information

1.1 Documentation

The following documents are associated with this document:

- The data sheet describes the technical data, the dimensions, the pin assignment, the accessories and the order key.
- The installation instructions describe the mechanical and electrical installation with all safety-relevant conditions and the associated technical specifications.
- The User manual for actuator commissioning and integration into a fieldbus system.

You can also download these documents at **http://www.siko-global.com/p/MSK5000CAN**

1.2 History

Mod. status	Date	Description
152/23	14.09.2023	Document prepared

1.3 Definitions

Decimal values are specified as digits without a suffix (e.g., 1234), except when specified in direct conjunction with binary or hexadecimal values. Then the extension d will be used (e.g. 1234d). Binary values are marked with b (e.g. 1011b) and hexadecimal values with h (e.g. 280h) behind the digits.

Intended Use

The encoder detects the incremental poles of the magnetized magnetic tape MB500/1 and thereby generates an incremental position value. The encoder can be parameterized and read via the CAN interface using the CANopen protocol. Consequently, the position value can be read with a resolution of 0.005 mm by a higher-level controller, for example.

If the encoder is lifted off the tape, an error is detected and a position value of 0 is output.

2.1 Switching on the supply voltage

The MSK5000CAN initializes after it is switched on. The configuration parameters are loaded from the nonvolatile memory into the main memory of the controller.

As long as no changes have been made to the encoder, the encoder works with its default values. If parameters have been changed, the encoder works with the changed data. If these

are also to be used after a power off/on, they must be stored.
Upon completion of the initialization procedure, it sends a special NMT command, the boot-up message, to notify the system of its existence. The MSK5000CAN is then in pre-operational mode. In this state, the encoder can be parameterized according to the requirements of the application via SDO commands. This concerns both the configuration parameters and the way in which it makes its position values available to the system (asynchronous or synchronous data transmission).

2.2 Setup Guide – Guideline for easy commissioning

To get the MSK5000CAN up and running quickly and easily, proceed as follows:

Function description

Position value

NOTICE	The output position value is not an absolute value. After the encoder has been switched on, it must be referenced; otherwise, the encoder does not provide any position values (object 0x5115h)

3.1 Measurement range

The figures in the following representations refer to a resolution of 0.005 mm.
MSK5000CAN with magnetic tape MB500/1 with incremental coding enables a maximum measuring range of *-2,147,483,647* to *+2,147,483,647* measuring steps at said resolution.

Numerical value display:

The numerical value output by the MSK5000CAN encoder via the interface always represents a multiple of the set resolution.

Example:

Encoder output value = 340603; set resolution = 0.005 mm
Position value = 340603 * 0.005 mm = 1703.015 mm

Counting direction

The encoder provides increasing numerical values when it is moved in the direction of the connector outlet. This property can be changed via the object (decreasing numerical values when moving in the direction of the connector outlet).

3.2 Calibration

MSK5000CAN is an incremental measuring system with an absolute interface, i.e., the information of the position value is stored as an absolute value in the encoder. When the encoder is switched on, it must therefore be referenced, since no absolute position can be generated from the magnetic tape.

3.3 Restore factory setting

To restore the delivery state of the encoder, the following options exist:

Access	Coding		The following are set to the factory setting
CANopen (Cf. Object 1010h: Store Parameter)	1011h "load"	Sub-index 1	All parameters
		Sub-index 2	Only bus parameters
		Sub-index 3	Only CiA DS-406 parameters
		Sub-index 4	Only manufacturer-specific parameters

Table 1: Access factory settings

Communication via CAN bus (CANopen)

The basis for the MSK5000CAN is the CANopen communication profile CiA 301 V4.2, and the indicator specification CiA DS-303 Part 3 V1.4.0 is the basis for the device profile for encoders CiA 406 V3.2 as well as the CAN diagnosis. The MSK5000CAN supports device class C1 and to some extent C2. The details necessary for understanding operation are provided in this documentation. If you require more detailed information, we recommend the relevant specialist literature on CAN or CANopen.

4.1 Telegram structure

The data telegram of a CAN message consists of the following fields:

SOF	Identifier (COB ID)	Control field	Data field (a maximum of 8 bytes)	CRC	ACK/EOF

SOF:

(Start of Frame) Start bit of the telegram

Identifier (COB ID):

All bus participants use the identifier to check whether the message is relevant to them.

The identifier sets the priority of the message. The lower the value of the identifier, the higher the priority of the message. As a result, important messages are preferably transmitted via the bus.

The identifier field contains the identifier as well as bits for identifying the length of the identifier (11 or 29 bits). In addition, the identifier defines the encoder address, the channel selection and the data direction.

Consequently, the 11-bit identifier (COB identifier) is composed of a 4-bit function code and a 7-bit node number:

Bit no.	10	9	8	7	6	5	4	3	2	1	0
Type	Function code				Node number (Node ID)						
Assignment	x	x	x	x	0	0	x	x	x	x	X

The following function codes are defined in the "Pre-Defined Connection Set" (only those function codes that are used in this encoder are shown):

Object	Function code	Resulting COB ID	Object	Page
Network Management (NMT)	0000b	0	-	8
SYNC message	0001b	128d (80h)	1005h	24
Emergency message	0001b	128d (80h) + Node ID	1014h	32
TPD01	0011b	384d (180h) + Node ID	1800h	35
TPD02	0101b	640d (280h) + Node ID	1801h	36
SDO (tx)	1011b	1408d (580h) + Node ID	1200h	34
SDO (rx)	1100b	1536d (600h) + Node ID	1200h	34
Heartbeat message	1110b	1792d (700h) + Node ID	-	20
Node guard message	1110b	1792d (700h) + Node ID	-	20

Table 2: Overview of COB Identifiers

Changes to COB IDs are only possible in the PRE-OPERATIONAL NMT state. Via bit 31 = 1b, the COB ID must first be invalidated before it can be changed and reactivated.
An exception is the COB ID of the sync object. Bit 30 = 0b must be set there to change the COB ID. Since bit 30 cannot be set to 1b in MSK5000CAN, the COB ID could be changed at any time.

The node number (Node ID) (cf. Object

Data content	0 = Terminating resistor not present 1 = Terminating resistor present

5F0Ah: Node ID and baud rate Bus CAN) is assigned once in each bus system from the master to the MSK5000CAN during the configuration. The node numbers are in the range from 1 to 127. Node ID = 0 is reserved and may not be used.

The transfer of a newly set node number takes place only at a renewed initialization (cf. Chapter 4.2.1: Network Management Services (NMT)).

MSK5000CAN is delivered ex-works with Node ID 1 (1h).

Control field:

Contains bit-by-bit information about the number of user data and decides whether it is a data frame or remote transmission request (RTR) frame.

Data field:

Contains up to 8 bytes of user data. Depending on the channel selection, the user data has different meanings.

CRC:

Contains bits for error detection.

ACK/EOF:

The ACK/EOF field contains telegram confirmation bits as well as bits to identify the telegram end.

The exact description of the telegram can be found in the relevant CAN literature. In the following telegram descriptions, only the identifier (COB ID) and the data field are discussed for the sake of simplicity.

4.2 Node control

4.2.1 Network Management Services (NMT)

The master uses the NMT service to configure, manage and monitor network nodes. The encoder is always in one of the four communication states: "INITIALIZATION", "PRE-OPERATIONAL", "OPERATIONAL" or "STOPPED" (cf. Fig. 1)

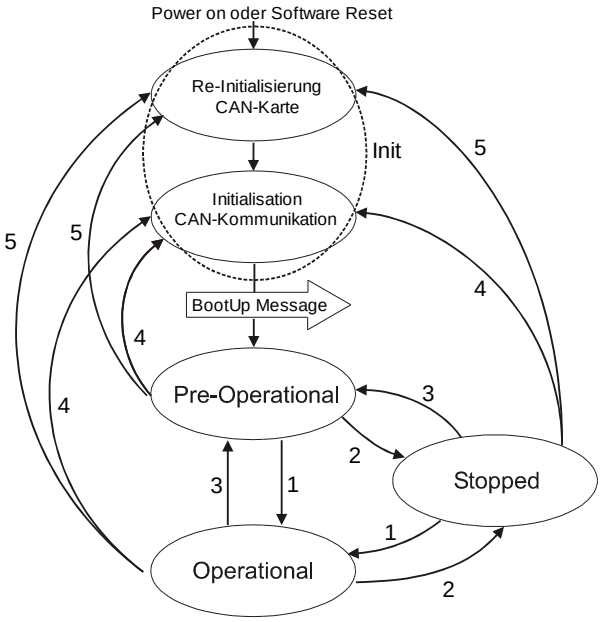


Fig. 1: NMT Status Diagram

4.2.1.1 NMT Communication States

NMT Status INITIALIZATION

In this state, the encoder is not involved in what is happening on the bus. All hardware and software components are initialized. This state is reached after the encoder is switched on or after receiving the command code 81h ("Reset Node") of its own or the

global address. After receiving the command code 82h ("Reset Communication"), the display is also in the initialization state. However, only the hardware and software are reinitialized that are related to the CAN communication. The encoder automatically signals the completion of initialization with a boot-up message. As soon as the boot-up message has been sent successfully, the encoder is in the "PRE-OPERATIONAL" state.

NMT PRE-OPERATIONAL State

Parameterization data (SDO) can be exchanged in Pre-Operational mode. However, no process data (PDO) are transmitted.

NMT OPERATIONAL State

The exchange of process data is also released. COB ID and Transmit PDO Mapping parameters cannot be changed in this state.

NMT STOPPED State

With the exception of Heartbeat and Node Guarding, communication is stopped. Only NMT communication is possible.

4.2.1.2 Switching between the NMT communication states

To switch between the communication states, telegrams with the following structure are used:

State Change		Transition into Fig. 1	COB ID	Command	Node ID
From	To				
PRE-OPERATIONAL/STOPPED	OPERATIONAL	1d	0h	01h	x
OPERATIONAL/PRE-OPERATIONAL	STOPPED	2d	0h	02h	x
OPERATIONAL/STOPPED	PRE-OPERATIONAL	3d	0h	80h	x
OPERATIONAL/ PRE-OPERATIONAL/STOPPED	INITIALIZATION (Reset Node)	5d	0h	81h	x
OPERATIONAL/ PRE-OPERATIONAL/STOPPED	INITIALIZATION (Reset Communication)	4d	0h	82h	x

Table 3: Switching between communication states

If x = 0h is transmitted as Node ID, then the message is intended for all bus subscribers.

4.2.2 Boot-Up

The COB ID of the boot-up message consists of 700h and the Node ID. The NMT state "initialization" is output as data content.

COB ID	Byte 0
700h + Node ID	00h

Table 4: Boot-up message

4.2.3 SYNC Object

CANopen makes it possible to interrogate inputs and set outputs simultaneously. A synchronization message serves for this (SYNC), which is a high priority CAN message. The identifier of the sync object can be set via the object 1005h (cf. 1005h: COB-ID SYNC Message).

4.3 Process Data Exchange

4.3.1 Transmission of Process Data Objects (PDO)

Process Data Objects (PDO) are used for a fast exchange of process data. A maximum of 8 bytes of user data can be transferred in a PDO. MSK5000CAN supports the transmit PDO services TPDO1 and TPDO2 according to CiA -301 and CiA -406. The data content can be individually adapted via variable mapping.

4.3.1.1 Transmit PDO (from the MSK5000CAN to the master)

A PDO transmission from the display to the bus master (TPDO) can be initiated by various events:

Asynchronously controlled by internal device timer

Synchronously in response to a SYNC message

In response to an RTR message

TPDO1 and TPDO2 are formed from the position value and the speed value at delivery. The transmission behavior of TPDO1 is defined by the objects 1800h, 1A00h and 6200h and is assigned to asynchronous transmission. The TPDO2 is defined via the objects 1801h and 1A01h and serves for synchronous transmission.

The messages are structured as in Table 5, whereby the mapping is variable (cf. Chapter 4.3.1.2 Variable TPDO Mapping) and can be changed.

COB ID	User Data in Binary Code					
	Byte 0 (LSB)	Byte 1	Byte 2	Byte 3 MSB	Byte 4 (LSB)	Byte 5 MSB
TPDO1 Configure Node ID	Position value				Speed value	
TPDO2 280h + Node ID						

Table 5: TPDO message

Asynchronous Data Transmission (TPDO1)

If a TPDO1 is to be sent cyclically, the cycle time in milliseconds must be entered in the object 1800h, Sub-index 05h. If the value 0 ms is written, TPDO1 is not sent. The function is switched off. The minimum value to be set is 1h (= 1 ms). Alternatively, the value can also be written in the internally linked object 6200h.

Synchronous data transmission (TPDO2)

Upon delivery, the encoder responds to each received SYNC message with the output of the TPDO2 message. In object 1801h, sub-index 02h, 1h is entered for synchronous transmission. If a value n is entered between 1d and 240d (= F0h), the encoder responds to every nth SYNC message.

RTR

Requests can be sent via RTR (cf. Chapter 0 The basis for the MSK5000CAN is the CANopen communication profile CiA 301 V4.2, and the indicator specification CiA DS-303 Part 3 V1.4.0 is the basis for the device profile for encoders CiA 406 V3.2 as well as the CAN diagnosis. The MSK5000CAN supports device class C1 and to some extent C2. The details necessary for understanding operation are provided in this documentation. If you require more detailed information, we recommend the relevant specialist literature on CAN or CANopen.

Telegram structure, Control Field) to TPDO1 and TDPO2.

4.3.1.2 Variable TPDO Mapping

By changing the objects 1A00h and 1A01h, you can set which data content is to be transmitted in the TPDOs. A maximum of 8 data bytes can be mapped in a TPDO.

Procedure for changing the TPDO mapping:

1. The encoder must be in the Pre-Operational NMT state.
2. By setting the COB ID Valid Bit to 1, the corresponding TPDO is deactivated.
3. Mapping is deactivated by writing the Sub-index 00h to 0h.
4. The mapping is changed by writing the desired objects and the data length to the desired subindices.
5. The maximum Sub-index used must be entered in Sub-index 00h for the mapping to be enabled.
6. The TPDO is reactivated by deleting the valid bit of the COB ID to 0.

Example of changing a TPDO1 mapping:

Default setting:

Node ID 1h

TPDO1: COB ID 00000181h

Mapping

1A00.0h 2

60040020h (position value of object 6004h, Sub-index 00h, 32 bit)

60300110h (Speed Object 6030h, Sub-index 01h, 16 bit)

Desired mapping:

1A00.0h 3

1A00.1h (position value of object 6004h, Sub-index 00 h, 32 bit)

1A00.2h 51220108h (Sys Register Object 5122h, Sub-index 01h, 8 bit)

1A00.3h 51220208h (Flag 0 Register Object 5122h, Sub-index 02h, 8 bit)

7. The encoder must be in the Pre-Operational NMT state.

Switch TPDO1 to inactive via COB ID = 80000181h.

COB ID	User Data							
	Command	Index L	Index H	Sub-index	Data 0	Data 1	Data 2	Data 3
601h	23h	00h	18h	01h	81h	01h	00h	80h

Deactivate current mapping with 1A00.0h = 0.

COB ID	User Data							
	Command	Index L	Index H	Sub-index	Data 0	Data 1	Data 2	Data 3
601h	23h	00h	1Ah	00h	00h	00h	00h	00h

Make desired changes to the mapping.

COB ID	User Data							
	Command	Index L	Index H	Sub-index	Data 0	Data 1	Data 2	Data 3
601h	23h	00h	1Ah	01h	20h	00h	04h	60h
601h	23h	00h	1Ah	02h	08h	01h	22h	51h
601h	23h	00h	1Ah	03h	08h	02h	22h	51h

Activate mapping via 1A00.0h = 3.

COB ID	User Data							
	Command	Index L	Index H	Sub-index	Data 0	Data 1	Data 2	Data 3
601h	23h	00h	1Ah	00h	03h	00h	00h	00h

Activate TPDO1 via COB ID = 00000181h.

COB ID	User Data							
	Command	Index L	Index H	Sub-index	Data 0	Data 1	Data 2	Data 3
601h	23h	00h	18h	01h	81h	01h	00h	00h

4.4 Parameter Data Exchange

4.4.1 Transmission of Service Data Objects (SDO)

Service data objects are mainly used for encoder configuration via the object directory. SDOs are supported in the expedited request/response ("accelerated request and confirmation procedure") and in the normal request/response.

The identifier is set to 11 bits and cannot be changed.

Two different SDO services are available:

SDO (rx) (Master → MSK5000CAN): 600h + Node ID

SDO (tx) (MSK5000CAN → Master): 580h + Node ID

These SDO identifiers cannot be changed!

4.4.1.1 Accelerated request and confirmation procedure

Except for the reading of the object 1008h: Manufacturer Device Name, all SDOs are exchanged between two parties in the "expedited request/response" procedure. The user data is already delivered with the initialization message.

These SDO messages have the following structure:

COB ID	User data in binary code							
	Byte 0 read / write	Byte 1 LSB	Byte 2 MSB	Byte 3	Byte 4 LSB	Byte 5	Byte 6	Byte 7 MSB
SDO rx/tx + Node ID	Command byte	Index		Sub-index	User data (parameters)			

Command Byte, Byte 0:

The command byte determines the type of access and the number of valid data bytes. The following command bytes are valid for the MSK5000CAN:

Command Byte		Type	Function
Write Request	23h	SDO (rx), Initiate Download Request, expedited	Send parameter to slave (all 4 data bytes valid)
Write Request	2Bh	SDO (rx), Initiate Download Request, expedited	Send parameter to slave (2 bytes of 4 data bytes valid)
Write Request	2Fh	SDO (rx), Initiate Download Request, expedited	Send parameter to slave (1 byte of 4 data bytes valid)
Write Response	60h	SDO (tx), Initiate Download Response	Confirmation of data transfer to the master
Read Request	40h	SDO (rx), Initiate Upload Request	Request parameters of the slave
Read Response	43h	SDO (tx), Initiate Upload Response, expedited	Report parameters to Master (all 4 data bytes valid)
Read Response	4Bh	SDO (tx), Initiate Upload Response, expedited	Report parameters to Master (2 bytes of 4 data bytes valid)
Read Response	4Fh	SDO (tx), Initiate Upload Response, expedited	Report parameters to Master (1 byte of 4 data bytes valid)
Error Response	80h	SDO (tx), Abort Domain Transfer	Slave reports error code to master

Table 6: Command encoding

Index, Bytes 1 and 2:

The index (object number) is entered in Intel data format in user data byte 2 (low byte) and user data byte 3 (high byte). The index of the object to be parameterized is entered there.

Sub-index, Byte 3:

For objects that are executed as an array, the Sub-index indicates the number of the field.

User Data (Parameters), Bytes 4-7:

In the user data, the value of the parameter is entered in left-aligned Intel representation. Byte 4 = low Byte ... Byte 7 = high Byte

4.4.1.2 Normal Request and Confirmation Procedure

If more than 4 bytes of service data have to be transmitted, the data is exchanged between two participants using the “normal request/response procedure”. This procedure is also initiated by an initialization message, and the actual user data are then transmitted in the following segment messages.

With the MSK5000CAN, this is only the case when the object 1008h: Manufacturer Device Name is read.

The initialization message has the following structure:

COB ID	User data in binary code							
	Byte 0 read / write	Byte 1 LSB	Byte 2 MSB	Byte 3	Byte 4 LSB	Byte 5	Byte 6	Byte 7 MSB
SDO rx/tx + Node ID	Command byte	Index		Sub-index	User data (number of user data)			

The segment message has the following structure:

COB ID	User data in binary code							
	Byte 0 read / write	Byte 1 LSB	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7 MSB
SDO rx/tx + Node ID	Command byte	User Data						

Initialization and segment message: Command Byte, Byte 0:

The command byte determines the type of access and the number of valid data bytes. The following command bytes are valid for the MSK5000CAN:

Command Byte		Type	Function
Read Request	40h	SDO (rx), Normal Initiate Upload Request	Request parameter from slave (number of bytes to be transmitted)
Read Request	60h	SDO (rx), Normal Segment Upload Request	Request parameters from slave (user data)
Read Response	41h	SDO (tx), Normal Initiate Upload Response	Report parameter to master (number of bytes to be transferred)
Read Response	03h	SDO (tx), Normal Segment Upload Response	Report parameters to Master (user data)
Error Response	80h	SDO (tx), Abort Domain Transfer	Slave reports error code to master

Table 7: Command encoding

Initialization Message: Index, Bytes 1 and 2:

The index (object number) is entered in Intel data format in user data byte 2 (low byte) and user data byte 3 (high byte). The index of the object to be parameterized is entered there.

Initialization Message: Sub-index, Byte 3:

For objects that are executed as an array, the Sub-index indicates the number of the field.

Initialization Message: User Data (Parameters), Bytes 4-7:

In the service data area, the value of the parameter is entered in left-aligned Intel representation. Byte 4 = low Byte ... Byte 7 = high Byte

Segment message: User Data (Parameters), Bytes 1-7:

In the user data area, the value of the parameter is entered in left-aligned Intel representation. Byte 1 = low Byte ... Byte 7 = high Byte

4.4.1.3 Error response in SDO exchange

If the access is invalid, an error message (abort) is returned to the master.
The error codes are described in the CANopen profile (CiA -301) or in the encoder profile (CiA 406). The following table shows the error codes used:

Error Code	Description
05030000h	Toggle bit in normal transfer of request/response unequal.
06010000h	Incorrect access to an object.
06010001h	Read access to write-only.
06010002h	Write access to read-only.
06020000h	Object does not exist in the object dictionary.
06040041h	Object cannot be mapped to PDO.
06040042h	The number and length of objects to be mapped exceed the PDO length.
06090011h	Sub-index does not exist.
06090030h	Value range of the selected parameter incorrect.
08000020h	Parameters cannot be transferred or saved to the application.
08000022h	Parameters cannot be transferred or saved to the application due to the current state of the device.
08000024h	No data available

Table 8: Error codes

4.4.1.4 SDO Examples

Example of Reading SDO Parameters with Accelerated Request and Confirmation Procedure:

From the slave with device address 1h, the calibration value stored in object 6010h Sub-index 6003h of the object directory is to be read.

Calculation of the identifier: 600h + Node ID = 600h +1h = 601h

Command: 40h
Index: 6003h
Sub-index: 00h
The current value is 510d = 01 FEh

Request from master to slave with Node ID 1h:

COB ID	User Data							
	Command	Index L	Index H	Sub-index	Data 0	Data 1	Data 2	Data 3
601h	40h	03h	60h	00h	x	x	x	x

Slave response to the request:

Calculation of the identifier: 580h + Node ID = 581h

COB ID	User Data							
	Command	Index LB	Index HB	Sub-index	Data 0	Data 1	Data 2	Data 3
581h	43h (4 bytes valid)	03h	60h	00h	FEh	01h	00h	00h

Example of Writing SDO Parameters with Accelerated Request and Confirmation Procedure:

In the slave with device address 1h, the calibration value, which is stored with 2 bytes in object 6200h of the object directory, is to be changed.

Calculation of the identifier: 600h + Node ID = 600h + 1h = 601h

Command: 2 bytes are to be written: 2Bh
Index: 6200h
Sub-index: 00h
The new value should be 4500d = 1194h

Write a value from the master to the slave with Node ID 1h:

COB ID	User Data							
	Command	Index L	Index H	Sub-index	Data 0	Data 1	Data 2	Data 3
601h	2Bh (2 bytes valid)	00h	62h	00h	94h	11h	00h	00h

Response of the slave to the command:

Calculation of the identifier: 580h + Node ID = 580h + 1h = 581h

COB ID	User Data							
	Command	Index L	Index H	Sub-index	Data 0	Data 1	Data 2	Data 3
581h	60h	00h	62h	00h	00h	00h	00h	00h

Example of Reading SDO Parameters with Normal Request and Confirmation Procedure:

From MSK5000CAN with device address 1h, the manufacturer device name stored in object 1008h of the object directory is to be read.

Calculation of the identifier: 600h + Node ID = 600h +1h = 601h

Command: 40h
Index: 1008h
Sub-index: 00h

First request (initialization) from master to slave with Node ID 1h:

COB ID	User Data							
	Command	Index L	Index H	Sub-index	Data 0	Data 1	Data 2	Data 3
601h	40h	08h	10h	00h	x	x	x	x

Slave response to the request:

Calculation of the identifier: 580h + Node ID = 581h

COB ID	User Data							
	Command	Index LB	Index HB	Sub-index	Data 0	Data 1	Data 2	Data 3
581h	41h	08h	10h	00h	06h	00h	00h	00h

Expected number of user data bytes: 6

Second request from master to slave with Node ID 1h:

COB ID	User Data							
	Command	Index L	Index H	Sub-index	Data 0	Data 1	Data 2	Data 3
601h	60h	08h	10h	00h	x	x	x	x

Slave response to the request:

COB ID	User Data							
	Command	Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6
581h	03h	4Dh	53h	4Bh	35h	30h	30h	30h
		"M"	"S"	"K"	"5"	"0"	"0"	"0"

4.5 Node monitoring

4.5.1 Emergency service (EMCY)

The state of the bus participant is transmitted in the event of a fault via high-priority emergency messages. These messages have a data length of 8 bytes and contain error information.

The emergency message is transmitted as soon as an encoder or communication error has occurred or is rectified. The cause of the fault is stored in the fault buffer (cf. Object 1003h: Pre-defined Error Field). An emergency object is only sent once per error event. If a cause of the fault has been eliminated, this is signaled by sending an emergency message with error code 0000h (No Error). If there are several malfunctions and one cause of the fault is eliminated, the error code 0000h is also output, but the remaining error state is specified in the error register.

Identifier	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
11/ 29 Bit	Emergency Error Code		Error Register (Object 1001h)	Manufacturer-specific error field (not used)				

Emergency Error Code

Error Description	Error Code
Malfunction cause eliminated	0000h

Bus status switches to Error Passive Mode	8120h
Recovered from Bus Off	8140h
Manufacturer specific: Tape encoder	FF10h
Manufacturer specific: Speed error	FF12h
Manufacturer specific: Plausibility error.	FF13h
Manufacturer specific: Alignment active t?	FF14h
Manufacturer specific: Incorrect values detected	FF15h
Manufacturer specific: Checksum error	FF16h
Manufacturer specific: Read/write error EEPROM	FF17h

Table 9: Emergency Error Code

The identifier of the emergency object is set to 80h + Node ID by default, but can be changed via object 1014h (cf. 1014h: COB ID emergency message). Sending an emergency message is only possible in the NMT state "OPERATIONAL" or "PRE-OPERATIONAL". The sending of emergency messages can be switched off by setting the COB ID valid bit to 1.

4.5.2 Node Guarding

Node Guarding is available for failure monitoring of the CANopen network. With Node Guarding, the master sets remote frames (RTR, remote transmission request) to the guarding identifiers of the bus nodes to be monitored. These respond with the guarding message. This contains the current NMT state of the node as well as a toggle bit whose value must change with each message. If the NMT state or toggle bit does not match the value expected by the master or there is no response occurs, the master assumes a node failure.

The time interval (life time) is set via the objects 100Ch (Guard Time) and 100Dh (Life Time Factor), within which the NMT master expects a message. The time interval "Life-Time" is calculated from the cycle time "Guard-Time", multiplied by the factor "Life-Time-Factor". If the NMT master does not receive a response to its RTR frame within the "life time", it can react with suitable measures. After powering up, Node Guarding is activated by sending the master's first RTR frame to the slave. If the value of one of the two objects (100Ch or 100Dh) is set to 0h, Node Guarding is deactivated.

The response of the node to the RTR frame of the master is structured as follows:

Identifier	**Byte 0**	
700h + Node-ID	Bit 7: Toggle Bit	Bit 6 ... 0: NMT state

Toggle Bit:

The toggle bit must alternate between two successive responses of the encoder. After the guarding protocol has been activated, the toggle bit must have the value 0 for the first response.

NMT State:

4: STOPPED

5: OPERATIONAL

127: PRE-OPERATIONAL

The identifier of the Node Guarding protocol is permanently set to 700h + Node ID and cannot be changed. Sending a Node Guard message is possible in the NMT sate "OPERATIONAL", "PREOPERATIONAL" or "STOPPED".

Note:

The literature recommends using the heartbeat for node monitoring. With the Node Guarding protocol, only the master can detect missing communication. The heartbeat, on the other hand, can be received by all participants.

4.5.3 Heartbeat

The master monitors the state of the slave device via the heartbeat protocol. The encoder automatically sends its NMT status cyclically in this case. The MSK5000CAN is a heartbeat producer; it does not receive and process heartbeat protocols itself. The cycle time of the heartbeat message is set via object 1017h. If the cycle time is 0h, the heartbeat protocol is deactivated.

The heartbeat message consists of the COB ID and an additional byte. This byte stores the current NMT state.

COB ID	Byte 0
700h + Node-ID	NMT state

NMT State:

4: STOPPED

5: OPERATIONAL

127: PRE-OPERATIONAL

The identifier of the Heartbeat protocol is permanently set to 700h + Node ID and cannot be changed. Sending a Heartbeat message is possible in the NMT state "OPERATIONAL", "PREOPERATIONAL" or "STOPPED".

4.6 Object Directory

4.6.1 Object Overview

The following table provides an overview of the objects of the device.

Name	Description	Cf. page
1000h Device Type	Device profile and encoder type	22
1001h: Error Register	Current error state of the encoder	23
1002h: Manufacturer Status Register	Contains the Transmit Error Counter and the Receive Error Counter	23
1003h: Pre-defined Error Field	The object stores the 8 most recent error states	23
1005h: COB-ID SYNC Message	Setting of the COB ID of the SYNC object	24

Name	Description	Cf. page
1008h: Manufacturer Device Name	Device name in ASCII characters	25
1009h: Manufacturer Hardware Version	Specifies the hardware version of the encoder	25
100Ah: Manufacturer Software Version	Specifies the software version of the encoder	26
100Ch: Guard Time	Parameters for node guarding	26
100Dh: Life Time Factor	Parameters for node guarding	26
1010h: Store Parameter	Object for parameter storage in EEPROM so that they are voltage fail-safe.	27
1011h: Restore Parameter	Object for restoring the factory settings	29
1014h: COB ID emergency message	COB ID of the emergency object	32
1015h: Inhibit time EMCY	Emergency message delay time	32
1017h: Producer Heartbeat Time	Setting of the cycle time of the heartbeat timer	32
1018h: Identity Object	Contains the manufacturer number	33
1200h: Server SDO Parameter	SDO parameter	34
1800h: 1. Transmit PDO Parameter	Transmit PDO for asynchronous transmission (timer-controlled)	35
1801h: 2. Transmit PDO Parameter	Transmit PDO for synchronous transmission	36
1A00h: 1. Transmit PDO Mapping Parameter	Describes the arrangement of the objects, which are shown in TPDO1	38
1A01h: 2. Transmit PDO Mapping Parameter	Describes the arrangement of the objects that are displayed in TPDO2	38
5115h: Referencing encoders	Set position value to calibration value	39
5122h: Register	Reading various registers	39
5F09h: Bus terminal.	Bus terminal.	40
5F0Ah: Node ID and baud rate Bus CAN	Setting of the Node ID and the baud rate	41
5FFAh Speed alarm	Setting the alarm for max. speed	43
6000h: Operating parameters	Setting of the scaling and the direction of rotation	43
6003h: Preset value (calibration value)	Setting of the calibration value	44
6004h: Position value	Position value (offset against calibration and offset value)	44
6005h: Resolution	Resolution and speed increment	46
6030h: Speed value	Speed value	46
6200h: Cycle Timer	Identical with object 1800h, Sub-index 5	46
6500h: Operating Status	Output of the scaling and the direction of rotation	47
6501h: Measuring step	The physical number of measuring steps per revolution	48

Name	Description	Cf. page
6502h: Number of distinguishable revolutions	Number of revolutions that the encoder can detect	48
6507h: Profile and Software Version	Displays the version number of the device profile used and the version number of the device firmware	48
6508h: Operating Time	Operating hour meter (function is not supported)	49
6509h: Offset value	Encoder state at the time of calibration	49
650Ah: Module identification	Specifies the manufacturer-specific offset value as well as the smallest and largest transferable position value	49
650Bh: Serial number	Specifies the serial number	51

Table 10: Object Overview

4.6.1.1 1000h Device Type

The object 1000h specifies the device profile number.

Sub-index	00h			
Description	Information about device profile and device type			
Access	ro			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	00010196h			
EEPROM	No			
Data content	Device profile number		Encoder type	
	Byte 0	Byte 1	Byte 2	Byte 3
	96h	01h	01h	00h

0196h (= 406d): CANopen Device Profile for Encoders

0007h: Incremental Linear Encoder

4.6.1.2 1001h: Error Register

The object 1001h displays the error state of the encoder.

Sub-index	00h	
Description	Currently existing error state	
Access	ro	
PDO mapping	No	
Data type	UNSIGNED 8	
Default	0h	
EEPROM	No	
Data content	Bit	Significance
	0	Set bit indicates the occurrence of any error state
	4	Set bit indicates communication errors on the CAN bus (passive or Bus off)
	7	Manufacturer-specific (encoder error)
	1-3, 5-6	Not used

Faults and errors are signaled by an emergency message at the moment of their occurrence.

4.6.1.3 1002h: Manufacturer Status Register

Object 1002h outputs the counter readings of the tab "Transmit Error Counter" and "Receive error counter". The contents of this register provide information about transmission faults at the mounting location of the encoder.

Sub-index	00h			
Description	Transmit Error Counter and Receive Error Counter			
Access	ro			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	0h			
EEPROM	No			
Data content	Byte 0	Byte 1	Byte 2	Byte 3
	Receive Error Counter	Transmit Error Counter		

4.6.1.4 1003h: Pre-defined Error Field

The 8 most recent error states are archived in object 1003h (cf. Chapter 4.5.1): Emergency service (EMCY)).

The entry under Sub-index 0 specifies the number of the stored errors.

The most recent error state is always stored in Sub-index 01h. Previous error messages move in the position by one Sub-index further.

The entire error list is deleted when you write the value 0h at Sub-index 00h.

The entries in the error list have the format described in Chapter 4.5.1: Emergency service (EMCY).

Sub-index	00h
Description	Number of stored error messages
Access	rw
PDO mapping	No
Data type	UNSIGNED 8
Default	0h
EEPROM	Yes
Sub-index	01h-08h
Description	Error messages that occurred
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	0h
EEPROM	Yes

4.6.1.5 1005h: COB-ID SYNC Message

The COB ID of the SYNC object is set by object 1005h.

Sub-index	00h	
Description	Defines the COB ID of the synchronization object (SYNC)	
Access	rw (can only be described in the “Pre-Operational” state; cf. Chapter 0)	
PDO mapping	No	
Data type	UNSIGNED 32	
Default	80h	
EEPROM	Yes	
Data content	Bit 31	Not specified
	Bit 30	0: Encoder does not generate any SYNC message
	Bit 29	0: 11 bit identifier (CAN 2.0A) 1: 29 bit identifier (CAN 2.0B)
	Bit 28 … 11	0: if Bit 29 = 0 X: Bits 28 - 11 of the SYNC COB ID if bit 29 = 1
	Bit 10 … 0	X: Bits 10 - 0 of the SYNC COB ID

4.6.1.6 1008h: Manufacturer Device Name

Object 1008h specifies the device name. Because this contains 6 data bytes, the SDO Normal Transfer is required for reading (cf. Chapter 4.4.1.2): Normal Request and Confirmation Procedure).

Sub-index	00h						
Description	Encoder name in ASCII characters						
Access	Const						
PDO mapping	No						
Data type	Visible String						
Default	MSK5000CAN						
EEPROM	No						
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
	4Dh	53h	4Bh	35h	30h	30h	30h
	("M")	("S")	("K")	("5")	("0")	("0")	("0")

4.6.1.7 1009h: Manufacturer Hardware Version

Object 1009h specifies the hardware version.

Sub-index	00h			
Description	Hardware version in ASCII characters			
Access	Const			
PDO mapping	No			
Data type	Visible_String			
Default	V100			
EEPROM	No			
Data content	Byte 0	Byte 1	Byte 2	Byte 3
	56h ("V")	31h ("1")	30h ("0")	30h ("0")

4.6.1.8 100Ah: Manufacturer Software Version

Object 100Ah specifies the software version of the encoder.

Sub-index	00h			
Description	Software version in ASCII characters			
Access	Const			
PDO mapping	No			
Data type	Visible_String			
Default	V001			
EEPROM	No			
Data content	Byte 0	Byte 1	Byte 2	Byte 3
	56h ("V")	31h ("0")	30h ("0")	30h ("1")

4.6.1.9 100Ch: Guard Time

Object 100Ch specifies the cycle time set in the master for node guarding (cf. Chapter 5.5.2:). Node Guarding). The cycle time width is entered in milliseconds. The value "0h" means that Node Guarding is deactivated.

Sub-index	00h
Description	Guard Time
Access	rw
PDO mapping	No
Data type	UNSIGNED 16
Default	0h
EEPROM	Yes

4.6.1.10 100Dh: Life Time Factor

Object 100Dh specifies the Life Time Factor set in the master for node guarding (cf. Chapter 5.5.2: Node Guarding). The value "0h" means that Node Guarding is deactivated.

Sub-index	00h
Description	Life Time Factor
Access	rw
PDO mapping	No
Data type	UNSIGNED 8
Default	0h
EEPROM	Yes

4.6.1.11 1010h: Store Parameter

With this object, parameters are transmitted into the EEPROM, so that they are safe from power outages. Depending on the selection of which Sub-index is accessed, different parameter groups are saved. The string "save" must also be transmitted as data content.

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	4h
EEPROM	No

Sub-index	01h			
Description	Save all parameters			
Access	rw			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	1h			
EEPROM	No			
Data content	Writing:			
	Byte 0	Byte 1	Byte 2	Byte 3
	73h ("s")	61h ("a")	76h ("v")	65h ("e")
	Read:			
	Bit 31 ... 2	0, reserved		
	Bit 1	0: Encoder does not store parameters independently		
	Bit 0	1: Encoder stores parameters on command		

Sub-index	02h			
Description	Only save communication parameters (1000h-1FFFh, CiA TF -301)			
Access	rw			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	1h			
EEPROM	No			
Data content	Writing:			
	Byte 0	Byte 1	Byte 2	Byte 3
	73h ("s")	61h ("a")	76h ("v")	65h ("e")
	Read:			
	Bit 31 ... 2	0, reserved		
	Bit 1	0: Encoder does not store parameters independently		
	Bit 0	1: Encoder stores parameters on command		

Sub-index	03h			
Description	Only store application parameters (6000h-9FFFh, CiA TF -406)			
Access	rw			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	1h			
EEPROM	No			
Data content	Writing:			
	Byte 0	Byte 1	Byte 2	Byte 3
	73h ("s")	61h ("a")	76h ("v")	65h ("e")
	Read:			
	Bit 31 ... 2	0, reserved		
	Bit 1	0: Encoder does not store parameters independently		
	Bit 0	1: Encoder stores parameters on command		

Sub-index	04h			
Description	Only store manufacturer-specific parameters (2000h-5FFFh)			
Access	rw			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	1h			
EEPROM	No			
Data content	Writing:			
	Byte 0	Byte 1	Byte 2	Byte 3
	73h ("s")	61h ("a")	76h ("v")	65h ("e")
	Read:			
	Bit 31 ... 2	0, reserved		
	Bit 1	0: Encoder does not store parameters independently		
	Bit 0	1: Encoder stores parameters on command		

4.6.1.12 1011h: Restore Parameter

Object 1011h restores the factory settings of the device depending on the selection. The string "load" must be sent as data content and then a reset performed.

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	4h
EEPROM	No

Sub-index	01h			
Description	Set all parameters to factory settings			
Access	rw			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	1h			
EEPROM	No			
Data content	Writing:			
	Byte 0	Byte 1	Byte 2	Byte 3
	6Ch ("l")	6Fh ("o")	61h ("a")	64h ("d")
	Read:			
	Bit 31 ... 1	0, reserved		
	Bit 0	1: Encoder allows the loading of default parameters.		

Sub-index	02h			
Description	Only set communication parameters to the factory setting (1000h-1FFFh, CiA TF -301)			
Access	rw			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	1h			
EEPROM	No			
Data content	Writing:			
	Byte 0	Byte 1	Byte 2	Byte 3
	6Ch ("l")	6Fh ("o")	61h ("a")	64h ("d")
	Read:			
	Bit 31 ... 1	0, reserved		
	Bit 0	1: Encoder allows the loading of default parameters.		

Sub-index	03h			
Description	Only set application parameters to the factory setting (6000h-9FFFh, CiA TF -406)			
Access	rw			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	1h			
EEPROM	No			
Data content	Writing:			
	Byte 0	Byte 1	Byte 2	Byte 3
	6Ch ("l")	6Fh ("o")	61h ("a")	64h ("d")
	Read:			
	Bit 31 ... 1	0, reserved		
	Bit 0	1: Encoder allows the loading of default parameters.		

Sub-index	04h			
Description	Only set manufacturer-specific parameters to the factory setting (2000h-5FFFh)			
Access	rw			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	1h			
EEPROM	No			
Data content	Writing:			
	Byte 0	Byte 1	Byte 2	Byte 3
	6Ch ("l")	6Fh ("o")	61h ("a")	64h ("d")
	Read:			
	Bit 31 ... 1	0, reserved		
	Bit 0	1: Encoder allows the loading of default parameters.		

4.6.1.13 1014h: COB ID emergency message

The COB ID of the emergency object is set by object 1014h (cf. Chapter 4.5.1. Emergency service (EMCY)).

Sub-index	00h	
Description	Defines the COB ID of the emergency object (EMCY)	
Access	rw (can only be described in the “Pre-Operational” state; cf. Chapter 4.1: Telegram structure)	
PDO mapping	No	
Data type	UNSIGNED 32	
Default	80h + Node ID	
EEPROM	Yes	
Data content	Bit 31	0: EMCY object exists/is valid 1: EMCY object does not exist/is invalid
	Bit 30	Always 0b
	Bit 29	0: 11 bit identifier (CAN 2.0A) 1: 29 bit identifier (CAN 2.0B)
	Bit 28 ... 11	0: if Bit 29 = 0b X: Bits 28 - 11 of the EMCYCOB ID if bit 29 = 1b
	Bit 10 ... 0	X: Bits 10 - 0 of the EMCY COB ID

4.6.1.14 1015h: Inhibit time EMCY

Object 1015h specifies the delay time of the emergency message in 100 µs.

Sub-index	00h
Description	Defines the delay time of the emergency message
Access	rw
PDO mapping	No
Data type	UNSIGNED 16
Default	0h
EEPROM	Yes
Data content	0d ... 65535d (0h ... FFFFh); the numerical value corresponds to a multiple of 100 µs. Value 0 deactivates the service.

4.6.1.15 1017h: Producer Heartbeat Time

The cycle time "Heartbeat Time" is set for the Heartbeat protocol by object 1017h. The cycle time width is entered in milliseconds.

Sub-index	00h
Description	Defines the cycle time of the Heartbeat monitoring service
Access	rw
PDO mapping	No
Data type	UNSIGNED 16
Default	0
EEPROM	Yes
Data content	0d, 10d ... 65535d (0h, Ah ... FFFFh); the numerical value corresponds to a multiple of 1 ms. The value 0h deactivates the delay.

4.6.1.16 1018h: Identity Object

The manufacturer identification number (Vendor ID) is indicated by object 1018h.

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	4h
EEPROM	No

Sub-index	01h
Description	Manufacturer identification number (Vendor ID) assigned to SIKO GmbH by CiA
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	195h
EEPROM	No

Sub-index	02h
Description	Product Code (function is not supported; only compatibility entry for various configurators)
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	FFFFFFFFh
EEPROM	No

Sub-index	03h
Description	Revision Number (function is not supported; only compatibility entry for various configurators)
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	FFFFFFFFh
EEPROM	No

Sub-index	04h
Description	Serial Number
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	0h
EEPROM	Yes

4.6.1.17 1200h: Server SDO Parameter

The COB IDs are specified for the server SDOs by object 1200h. The COB IDs cannot be changed.

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	2h
EEPROM	No

Sub-index	01h
Description	COB ID Client -> Server (rx)
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	00000600h + Node ID
EEPROM	No

Sub-index	02h
Description	COB ID Server -> Client (tx)
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	00000580h + Node ID
EEPROM	No

4.6.1.18 1800h: 1. Transmit PDO Parameter

According to CiA DS-406, TPDO1 is used for asynchronous PDO transmission.
The communication parameters for TPDO1 are set by object 1800h.

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	5h
EEPROM	No

Sub-index	01h
Description	COB ID of PDO1
Access	rw (can only be described in the “Pre-Operational” state; cf. Chapter 0)
PDO mapping	No
Data type	UNSIGNED 32
Default	180h + Node ID
EEPROM	Yes

Sub-index	02h	
Description	Transmission Type	
Access	rw	
PDO mapping	No	
Data type	UNSIGNED 8	
Default	FEh (254d)	
EEPROM	Yes	
Data content	FEh (254d) FFh (255d)	PDO has asynchronous characteristics (PDO is sent depending on the "Event Timer").
	FDh (253d)	Encoder only responds to an RTR request if RTR Bit 30 is released in the COB ID.

Sub-index	03h
Description	Inhibit time (function is not supported; only compatibility entry for various configurators)
Access	Const
PDO mapping	No
Data type	UNSIGNED 16
Default	0h
EEPROM	No

Sub-index	04h (is not used; access generates an error message)

Sub-index	05h
Description	Event timer for TPDO1 hard-wired (CiA -406) with cyclic timer 6200h
Access	rw
PDO mapping	No
Data type	UNSIGNED 16
Default	0h
EEPROM	Yes
Data content	Writing the value 0h switches off the service. The content of this object is identical with object 6200h. If the value is changed when the timer is running, the change only takes effect after the next run of the timer.

4.6.1.19 1801h: 2. Transmit PDO Parameter

According to CiA DS-406, TPDO2 is used for synchronous PDO transmission.
The communication parameters for TPDO2 are set by object 1801h.

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	5h
EEPROM	No

Sub-index	01h
Description	COB ID of PDO2
Access	rw (can only be described in the “Pre-Operational” state; cf. Chapter 0)
PDO mapping	No
Data type	UNSIGNED 32
Default	280h + Node ID
EEPROM	Yes

Sub-index	02h	
Description	Transmission Type	
Access	rw	
PDO mapping	No	
Data type	UNSIGNED 8	
Default	1h	
EEPROM	Yes	
Data content	01h (1d) F0h (240d)	PDO is sent after received 1d ... 240d SYNC messages.
	FCh (252d)	Encoder only responds to an RTR request if RTR Bit 30 is released in the COB ID.

Sub-index	03h
Description	Inhibit time (function is not supported; only compatibility entry for various configurators)
Access	ro
PDO mapping	No
Data type	UNSIGNED 16
Default	0h
EEPROM	No

Sub-index	04h (is not used; access generates an error message)

Sub-index	05h
Description	Event timer (function is not supported; only compatibility entry for various configurators)
Access	ro
PDO mapping	No
Data type	UNSIGNED 16
Default	0h
EEPROM	No

4.6.1.20 1A00h: 1. Transmit PDO Mapping Parameter

The objects are specified by object 1A00h that are depicted in the first Transmit PDO (TPDO1) (cf. Chapter 4.3.1.2:. Variable TPDO Mapping).

Sub-index	00h
Description	Number of mapped objects
Access	rw
PDO mapping	No
Data type	UNSIGNED 8
Default	2h
EEPROM	Yes

Sub-index	01h
Description	1. Object of the PDO1 message (data bytes 0 to 3)
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	60040020h (position value of object 6004h, Sub-index 00h, 32 bit)
EEPROM	Yes

Sub-index	02h
Description	2. Object of the PDO1 message (data bytes 4 to 5)
Access	rw
PDO mapping	No
Data type	UNSIGNED 16
Default	60300110h (Speed Object 6030h, Sub-index 01h, 16 bit)
EEPROM	Yes

4.6.1.21 1A01h: 2. Transmit PDO Mapping Parameter

The objects are specified by object 1A01h that are depicted in the second Transmit PDO (TPDO2) (cf. Chapter 4.3.1.2:. Variable TPDO Mapping).

Sub-index	00h
Description	Number of mapped objects
Access	rw
PDO mapping	No
Data type	UNSIGNED 8
Default	2h
EEPROM	Yes

Sub-index	01h
Description	1. Object of the PDO2 message (data byte 0+1)
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	60040020h (position value of object 6004h, Sub-index 00h, 32 bit)
EEPROM	Yes

Sub-index	02h
Description	2. Object of the PDO2 message (data bytes 2 to 5)
Access	rw
PDO mapping	No
Data type	UNSIGNED 32
Default	60300110h (Speed Object 6030h, Sub-index 01h, 16 bit)
EEPROM	Yes

4.6.1.22 5115h: Referencing encoders

With object 5115h, referencing can be carried out or information can be provided about whether referencing is carried out. (Referencing = setting the position value to the calibration value after the startup of the encoder)

Sub-index	00h	
Description	It is possible to reference the measured value with this object. This sets the position value to the calibration value. Position value = measurement value + calibration value	
Access	rw	
PDO mapping	No	
Data type	UNSIGNED 8	
Default	0h	
EEPROM	Yes	
Data content	Object 5115h Read:	
	0, 1	In the case of a read access, a 0h is reported back if no referencing has yet been made after starting the encoder or the reference position has been lost due to a sensor error. If the encoder is referenced, a 1h is reported back.
	Write object 5115h:	
	1	Writing the value 1h sets the position value to the calibration value.

4.6.1.23 5122h: Register

Object 5122h can be used to read individual registers.

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	4h
EEPROM	No

Sub-index	01h	
Description	Read Sys register	
Access	ro	
PDO mapping	No	
Data type	UNSIGNED 8	
Default	0h	
EEPROM	No	
Data content	Bit 7	Not used
	Bit 6	Checksum error occurred
	Bit 5	Verify error (sensor is not aligned)
	Bit 4	Encoder/tape alignment in progress
	Bit 3	Not used
	Bit 2	Speed check (v > 5 m/s)
	Bit 1	Encoder not referenced
	Bit 0	Encoder/tape read distance error

Sub-index	02h	
Description	Flag 0 register	
Access	ro	
PDO mapping	No	
Data type	UNSIGNED 8	
Default	00h	
EEPROM	No	
Data content	Bit 7, 6	Not used
	Bit 5	Not used
	Bit 4...2	Not used
	Bit 1	Counting direction 0 = increasing in the direction of the cable outlet 1 = decreasing in the direction of the cable outlet
	Bit 0	Resolution: 0 = 10 µm; 1 = 5 µm

In the flag 1 register, the flags are reset after reading.

Sub-index	03h	
Description	Flag 1 register	
Access	ro	
PDO mapping	No	
Data type	UNSIGNED 8	
Default	83h	
EEPROM	No	
Data content	Bit 7	External reset (/MCLR) pin bit
	Bit 6	Software reset (instruction) Flag bit
	Bit 5	Software enable/disable or WDT bit
	Bit 4	Watchdog Timer Timeout Flag bit
	Bit 3	Wake-up from Sleep Flag bit
	Bit 2	Wake-up from Idle Flag bit
	Bit 1	Brown-out Reset Flag bit
	Bit 0	Power-on Reset Flag bit

In the Flag 2 register, the flags are reset after reading.

Sub-index	04h	
Description	Flag 2 register	
Access	ro	
PDO mapping	No	
Data type	UNSIGNED 8	
Default	0h	
EEPROM	No	
Data content	Bit 7	Trap Reset Flag bit
	Bit 6	Illegal Opcode or Uninitialized W Access Reset Flag bit
	Bit 5...2	Not used
	Bit 1	Configuration Mismatch Flag bit
	Bit 0	Voltage Regulator Standby During Sleep bit

4.6.1.24 5F09h: Bus terminal.

Internal bus termination resistance can be read by object 5F09h..
There are two different versions, which are delivered once with and once without a bus termination resistor. The correct version must be determined before ordering.

Sub-index	00h
Description	State of the internal bus termination
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	0h
EEPROM	Yes
Data content	0 = Terminating resistor not present 1 = Terminating resistor present

4.6.1.25 5F0Ah: Node ID and baud rate Bus CAN

Node ID and the baud rate bus can be set by object 5F0Ah. The transfer of a newly set Node ID or Bus baud rate only takes place at a renewed initialization (cf. Chapter Network Management Services (NMT)).

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	2h
EEPROM	No

Sub-index	01h
Description	Node ID
Access	rw
PDO mapping	No
Data type	UNSIGNED 8
Default	1h
EEPROM	Yes
Data content	01h ... 7Fh

Sub-index	02h
Description	Baud rate of the CAN bus
Access	rw
PDO mapping	No
Data type	UNSIGNED 8
Default	4h (500 kBaud)
EEPROM	Yes
Data content	1: 50 kBaud 2: 100 kBaud 3: 125 kBaud 4: 250 kBaud 5: 500 kBaud 6: 1000 kBaud

4.6.1.26 5FFAh: Speed alarm

This entry can be used to set the threshold for the overspeed alarm. The speed increment is set in Object 6005.2h.

Only active in Operational Mode

Sub-index	00h
Description	Maximum speed in mm/s. If 0: Alarm deactivated
Access	rw
PDO mapping	No
Data type	SIGNED 16
Default	0h
EEPROM	Yes
Data content	0: Off Resolution 5 um: 50 – 32767 (in mm/s) Resolution 10 um: 100 – 32767 (in mm/s)

4.6.1.27 6000h: Operating parameters

Operating parameters can be set via object 6000h.

Sub-index	00h	
Description	Operating parameters	
Access	rw	
PDO mapping	No	
Data type	UNSIGNED 16	
Default	4h	
EEPROM	Yes	
Data content	Bit 15 ... 5	Not used
	Bit 4	0: 5 µm / 1: 10 µm (resolution)
	Bit 3	0 = increasing position values when moving the encoder away from the encoder cable 1: increasing position values when moving the encoder to the encoder cable
	Bit 2	1: Release scaling
	Bit 1	Not used
	Bit 0	Not used

Scaling: The encoder works with its set resolution, which can be parameterized via object 6005h. It is not possible to switch off the scaling function.

4.6.1.28 6003h: Preset value (calibration value)

The position value of the encoder for calibration to a calibration value can be set via object 6003h. Position value = measurement value + calibration value

Sub-index	00h
Description	Calibration value
Access	rw
PDO mapping	No
Data type	SIGNED 32
Default	0h
EEPROM	Yes
Data content	-2147483648d ... 2147483647d (80000000h...7FFFFFFFh)

4.6.1.29 6004h: Position value

NOTICE	The encoder does not provide position values if it was not referenced at startup. (cf. Object 0x5115h)

Object 6004h indicates the current position value of the encoder.

Sub-index	00h
Description	Position value
Access	ro
PDO mapping	Yes
Data type	SIGNED 32
Default	0h
EEPROM	No

Position value = measurement value + calibration value

4.6.1.30 6005h: Resolution and speed increment

The resolution and the speed increment are specified by object 6005h.

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	2h
EEPROM	No

Sub-index	01h
Description	Resolution of the linear encoder. According to CiA -406, the parameter must be specified in multiples of nm.
Access	rw
PDO mapping	No
Data type	UNSIGNED 32
Default	*See Order key/Encoder nameplate*
EEPROM	Yes
Data content	5000d (1388h) or 10000d (2710h)

4.6.1.31 30h: Speed value

The speed can be read via object 6030h. The speed increment is set in Object 6005.2h.

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	1h
EEPROM	No

Sub-index	01h
Description	Speed value in increments per ms (Inc/ms)
Access	ro
PDO mapping	Yes
Data type	SIGNED 16
Default	0h
EEPROM	No

4.6.1.32 6200h: Cycle Timer

Object 6200h sets a cycle time, with which the PDO1 is to be output. This value is firmly linked to the object 1800h: 1. Transmit PDO Parameter Sub-index 05h. The timer-controlled output is active as soon as a valid cycle time is entered and the encoder is operated in operational mode. The value 0h deactivates the function.

Sub-index	00h
Description	Cycle Timer
Access	rw
PDO mapping	No
Data type	UNSIGNED 16
Default	0h
EEPROM	Yes
Data content	0d ... 65535d (0h...FFFFh)
Sub-index	02h
Description	Increment of the speed of the linear encoder. According to CiA -406, the parameter must be specified in multiples of 0.01 mm/s.
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	100d (64h)
EEPROM	Yes
Sub-index	02h
Description	Increment of the speed of the linear encoder. According to CiA -406, the parameter must be specified in multiples of 0.01 mm/s.
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	100d (64h)
EEPROM	Yes

4.6.1.33 6500h: Operating Status

The object 6500h indicates the settings programmed with object 6000h.

Sub-index	00h	
Description	Operating Status	
Access	ro	
PDO mapping	No	
Data type	UNSIGNED 16	
Default	4h	
EEPROM	No	
Data content	Bit 15 ... 4	Not used
	Bit 3	0 = Counting direction increasing in the direction of the cable outlet 1: Counting direction decreasing in the direction of the cable outlet
	Bit 2	0: Scaling locked 1: Release scaling
	Bit 1	Not used
	Bit 0	Not used

4.6.1.34 6501h: Measuring step

Object 6501h specifies the physical number of measuring steps.

Sub-index	00h
Description	Physical resolution
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	5000d (1388h)
EEPROM	No

4.6.1.35 6502h: Number of distinguishable revolutions

Object 6502h specifies the number of revolutions that the encoder can record.

Sub-index	00h
Description	Total number of detectable revolutions
Access	ro
PDO mapping	No
Data type	UNSIGNED 16
Default	1h
EEPROM	No

4.6.1.36 6507h: Profile and Software Version

Object 6507h displays the used encoder profile (CANopen device profiles for encoders) and the firmware version number.

Sub-index	00h			
Description	Profile and Software Version			
Access	ro			
PDO mapping	No			
Data type	UNSIGNED 32			
Default	01000302h			
EEPROM	No			
	Profile version		Software version	
	Byte 0 (LSB)	Byte 1	Byte 2	Byte 3 MSB
	02h	03h	01h	00h

4.6.1.37 6508h: Operating Time

The operating hours can be displayed via object 6508h. This function is not supported.

Sub-index	00h
Description	Operation hours counter
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	FFFFFFFFh
EEPROM	No

4.6.1.38 6509h: Offset value

Object 6509h outputs the position at which the encoder was referenced.

Sub-index	00h
Description	Position value at the time of calibration
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	0h
EEPROM	No

4.6.1.39 650Ah: Module identification

Object 650Ah specifies the manufacturer-specific offset value as well as the smallest and largest transferable position value

Sub-index	00h
Description	Displays the largest supported Sub-index
Access	ro
PDO mapping	No
Data type	UNSIGNED 8
Default	3h
EEPROM	No

Sub-index	01h
Description	Manufacturer-specific offset value
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	0h
EEPROM	No

Sub-index	02h
Description	Smallest transferable position value
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	-2147483648d (80000000h)
EEPROM	No

Sub-index	03h
Description	Largest transferable position value
Access	ro
PDO mapping	No
Data type	SIGNED 32
Default	2147483647d (7FFFFFFFh)
EEPROM	No

4.6.1.40 650Bh: Serial number

Object 650Bh supplies the serial number of the encoder.

Sub-index	00h
Description	Serial number
Access	ro
PDO mapping	No
Data type	UNSIGNED 32
Default	0h
EEPROM	Yes

SIKO GmbH

Weihermattenweg 2
79256 Buchenbach

Phone

+ 49 7661 394-0

Fax

+ 49 7661 394-388

E-Mail

info@siko-global.com

Internet

www.siko-global.com

Service

support@siko-global.com