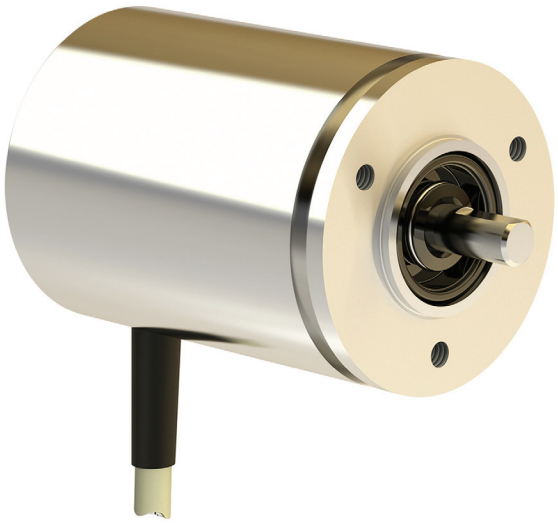
User manual – CANopen Safety (IEC61508) and CANopen for absolute rotary encoder TBN/TRN and TBSN/TRSN

SUPREME SENSORING
TWK

Document no.: TXN/TXSN 15469 GE
Date: 25.01.2024



SIL2 IEC 61508 FUNCTIONAL SAFETY SENSOR

SIL3 IEC 61508 FUNCTIONAL SAFETY SENSOR

CANopen

CANopen safety easy to use

Specification rotary encoder TBN/TRN and TBSN/TRSN

according to

CANopen Safety - SIL2, SIL3 acc. IEC 61508
and CANopen Standard

TWK-ELEKTRONIK GmbH
Bismarckstraße 108
40210 Düsseldorf
Tel.: +49 211 961170
info@twk.de
visit us at | twk.de

General

Scope

This specification is valid exclusively for the following absolute encoders TBN/TRN and TBSN/TRSN with CANopen and CANopen Safety (SIL2/SIL3) interface.

Documentation

The following documents must be observed (depending on device):

- The owner's system-specific operating instructions
- The specification (manual) TXN/TXSN 15469
- Data sheet TBN/TRN 14271 (SIL2)
- Data sheet TBSN/TRSN 16365 (SIL3)
- The connection assignment - TYxxxxx (depending on device, on request: enclosed with every device)
- EDS-File TYxxxxx_xx_xx.eds (on request)
- CRC Calculation Program and additional xml-Files for special versions (on request)
- TWK_CAN Error description (on request)
- Notes on the wiring & assembly of rotary encoder AN 16169
- Drawings (on request)

Proper use

Absolute encoders by TWK-ELEKTRONIK GmbH are used to acquire angular positions and provide their measured value in the form of an electrical output signal. As part of a system, they have to be connected to appriate downstream electronics concerning the use case and must only be used for this purpose.

Commissioning

- The relevant device may only be set up and operated in combination with this and the documentation specified above
- Protect the device against mechanical damage during installation and operation.
- Device commissioning and operation may only be undertaken by a specialist electrician.
- Do not operate the device outside of the limit values specified in the data sheet.
- Check all electrical connections before commissioning the system.

Table of contents

Table of contents

1 Intended use

The rotary encoders TB(S)N and TR(S)N are intended for use in CANopen and in CANopen safety-relevant systems.
The version TB(S)N is a singleturn and TR(S)N is a multiturn absolute encoder.
Thanks to its robust design, the encoder can be used in applications with harsh environmental conditions.

Both encoders (singleturn and multiturn) are designed for connection to a control system (PLC) which evaluates the measured values transmitted via the CAN bus and, in the event of error messages or the absence of process data, reacts according to the relevance of the error and prevents a hazardous state.

2 Design

Position sensing is carried out through angle measurement in case of TB(S)N. For TR(S)N position sensing is carried out through a combined angle and revolution measurement. The signals from the encoders are used to sense the position of the shaft and to scan a mechanical transmission for measuring the revolutions covered are combined to form a summarized position value. The position sensing system is equipped with a separate controller. Position and revolution sensing are designed in a fully redundant form. The position data are transferred to the evaluation module via two channels.

The evaluation module evaluates the position data supplied by the position registration system. The measured values are filtered, the two redundant channels are compared and the measurement signals are linearized. If the measured values are recognized as safe, the module makes the position data available to the downstream processes.

The absolute encoder module accepts the position data provided by the evaluation module, undertakes calibration depending on the set parameters and transfers the data to the CANopen interface. Depending on the setting, the CANopen interface can transfer both secure data via CANopen Safety (internal name: S4 for SIL2 / S41 for SIL3) and secure data via a normal, insecure CANopen interface to the application.

3 Documents

TWK data sheet	TBN/TRN 14271 for SIL2 devices TBSN/TRSN 16365 for SIL3 devices
Manual (Specification)	TXN/TXSN 15469 (this document)
SIL2 TÜV Certificate Registration No.	44 799 13 172915 (TWK: TXN15603)
SIL3 TÜV Certificate Registration No.	XX XXX XX XXXXXX (TWK: TXSNXXXXX)

4 Derived values

4.1 Velocity signal

The encoder provides a velocity signal. It is transmitted safely via SRDO2 and also as a non-safe transmission via PDO (for TBSN/TRSN SIL3 encoders only as per PDO transmission). The relationship between the value v and the value rotations per minute u is:

$$u\ [\text{rev./min}] = \frac{v\ [\text{digits}] \times 60000 \times S_D}{B\ [\text{digits/rev}] \times T\ [\text{ms}] \times S_M}$$

It means for safe speed (output at SRDO2):
u = Shaft speed in revolutions per minute
v = Digits read out from SRDO2 (Remark: The velocity signal is in the format Signed 16 Bits. At increasing position values you can use v directly. At decreasing position values you have to calculate: *FFFF - v* first, before inserting in the formula. Increasing or decreasing position values depend on the setting of *safety code sequence* - object 6101, sub 01). v_{max}: 15 Bits due to sign of velocity signal.
B = Basic resolution for velocity signal - depending on device
T = Safety speed integration time in milliseconds - see object 6101, sub 05
S_D = Safety speed divider - see object 6101, sub 07 (if existing, else = 1)
S_M = Safety speed multiplier - see object 6101, sub 06 (if existing, else = 1)
60000 = Compensation factor milliseconds ↔ minutes

5 CANopen functionality

The CANopen interface enables operation via CANopen Standard and secure operation via CANopen Safety. The CANopen profile definition for the encoder is executed once according to CiA 301 application layer and according to EN 50325-5 framework for safety-relevant communication.
The sensor system (position registration) is designed in redundant form. The sensor system's measured values are supplied to the self-monitoring controller on separate interfaces (SPI). This compares the position values of both sensors. If the measured values lie within a specified tolerance, the signals are transferred as validated for downstream evaluation. Otherwise, an error message is generated and the controller assumes a secure state (pre-operational).

5.1 Process data objects PDO's

Please see Obejcts 1800 and 1801 for user transmit configuration. For dynamic user configurable mapping, please see Objects 1A00 and 1A01.

The measured position and speed values are output in these objects as shown in standard configuration.

Byte 0								Byte 1								Byte 2							
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
LSB								Position															

Byte 3								Byte 4								Byte 5							
24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
						MSB		LSB						Speed									MSB

5.2 Safety-relevant data object SRDO1

The sensor depending on its resolution eg. 14/16 bit supplies 26/28 significant data bits accordingly in unsigned long format. Objects 6120 / 6121 are output.
Please see Object 1301 for user transmit configuration.

Byte 0								Byte 1								Byte 2								Byte 3							
0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
LSB								Position data																MSB				0	0	0	0

Byte 0								Byte 1								Byte 2								Byte 3							
0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
LSB								Position data inverted																MSB				1	1	1	1

5.3 Safety-relevant data object SRDO2

The sensor supplies 16 significant data in signed word format for the speed measurement value. Objects 6124 / 6125 are output on the SRDO2.
Please see Object 1302 for user transmit configuration.

Byte 0								Byte 1							
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LSB								Speed							MSB

Byte 0								Byte 1							
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LSB								Speed inverted							MSB

6 CANopen profile definition

6.1 CANopen profile overview

Table 1 of all objects contained in the encoder profile

Index	Data type	Designation	Data length	Memory type	m/o
General communication parameter					
1000	VAR	device_type	LONG	ro	m
1001	VAR	error_register	BYTE	ro	m
1003	ARRAY	pre_defined_error_field	-	ro	o
1005	VAR	COB-ID_SYNC	LONG	rw	o
1008	VAR	manufacturer_device_name	STRING	ro	o
1009	VAR	manufcturer_hardware_version	STRING	ro	o
100A	VAR	manufacturer_software_version	STRING	ro	o
1010	ARRAY	store_parameters	LONG	-	o
1011	ARRAY	restore_default_parameters	LONG	-	o
1014	VAR	COB-ID-EMCY	LONG	rw	o
1015	VAR	inhibit_time_EMCY	LONG	rw	o
1017	VAR	producer_heartbeat_time	WORD	rw	o
1018	RECORD	identity object		ro	m
1029	ARRAY	Error behavior	BYTE	rw	m
SRDO parameter set					
1301	RECORD	SRDO1 communication parameters		rw	m
1381	RECORD	SRDO_1_mapping_parameters			
1302	RECORD	SRDO2 speed parameters		rw	m
1382	RECORD	SRDO_2_mapping_parameters			
13FE	VAR	configuration valid	BYTE	rw	m
13FF	ARRAY	safety configuration checksum		rw	m
PDO parameter set					
1800	RECORD	PDO1_communication_parameters	-	rw	
1A00	RECORD	PDO1_mapping_objects		rw [t]	
1801	RECORD	PDO2_communication_parameters	-	rw	
1A01	RECORD	PDO2_mapping_objects		rw [t]	
General application parameter					
6000	VAR	operating parameters	WORD	rw*	m
6001	VAR	measuring units per revolution	LONG	rw*	m
6002	VAR	total measuring range in measuring units	LONG	rw*	m
6003	VAR	preset value	LONG	rw*	m
6004	VAR	position value	LONG	ro	m
6030	RECORD	speed_value	-	ro	o
6031	RECORD	speed parameters	-	rw*	o
6200	VAR	cyclic timer	WORD	rw	

* At monoturn (TBN) encoders with CANopen Safety Profile SIL2 (TBN...S4...), or SIL3 encoders TBSN/TRSN: read only / ro

[t] Only valid for Multiturn (TRN) CANopen Safety SIL2 (TRN...S4...) encoder, else: read only / ro

6 CANopen profile definition

Index	Data type	Designation	Data length	Memory type	m/o
Encoder safety objects					
6100	RECORD	safety_position_configuration_parameters	-	rw	m
6101	RECORD	safety_speed_configuration_parameters	-	rw	m
6120	ARRAY	safety_standard_position_value	BYTE	rw	m
6121	ARRAY	safety_inverted_standard_position_value	BYTE	rw	m
6124	ARRAY	safety_speed_value	BYTE	rw	o
6125	ARRAY	safety_inverted_speed_value	BYTE	rw	o
61FE	VAR	safety_configuration_valid	BYTE	rw	m
61FF	ARRAY	safety_configuration_signature		rw	m
Objects for achieving compatibility					
6200		cyclic_timer	WORD	rw	
Encoder diagnosis objects					
6500	VAR	Operating status	WORD	ro	m
6501	VAR	Single-turn resolution	LONG	ro	m
6502	VAR	Number of distinguishable revolutions	WORD	ro	m
6503	VAR	Alarms	WORD	ro	m
6504	VAR	Supported alarms	WORD	ro	m
6506	VAR	Supported warnings	WORD	ro	m
6507	VAR	Profile and software version	LONG	ro	m
6508	VAR	Operating time	LONG	ro	m
6509	VAR	Offset value	LONG	ro	m
650A	RECORD	Module identification		ro	m
650B	VAR	Serial number	LONG	ro	m
650D	VAR	Absolute accuracy	BYTE	ro	m
650E	VAR	Device cabability	LONG	ro	m
LMT objects					
2000	VAR	node ID	BYTE	rw	o
2001	VAR	bit_rate	BYTE	rw	o
Manufacturer specific Objects					
3100	VAR	safety_gear_configuration	WORD / LONG	rw	o
3102	VAR	gear_configuration	WORD / LONG	rw	o
31FE	VAR	safety_gear_data_valid	BYTE	rw	o
31FF	RECORD	safety_gear_CRC_checksum	WORD	rw	o
3300	VAR	safety_encoder_parameter	BYTE / LONG	wp	o

rw read/write

ro read only

o optional

m mandatory

wp Factory programming

6 CANopen profile definition

6.2 Communication and service data objects (SDO)

For every modification of the encoder via SDO communication the device must be set preoperational first.
It is crucial to wait for the response message of the encoder before sending another SDO message. Then you can be sure that the SDO message (e.g. parameterization) is executed properly.
Subsequently the appropriate save procedure (1010) should be executed to ensure that the parameterized objects keep their values after a power OFF/ON cycle or NMT reset.
Please don't make a reset of the device (power OFF/ON or NMT reset) before all values are transmitted and/or saved properly.
Wait for the response of the encoder after saving or wait at least 500 ms.

6.2.1 Object 1000 device_type

The encoder types are defined as follows:

Coding	Device type designation
1	Singleturn absolute rotary encoder
2	Multiturn absolute rotary encoder
3	Singleturn absolute rotary encoder with electronic turn count
4	Incremental rotary encoder
5	Incremental rotary encoder with electronic counting
6	Incremental linear encoder
7	Incremental linear encoder with electronic counting
8	Absolute linear encoder
9	Absolute linear encoder with cyclic coding
10	Multi-sensor encoder interface

Device_type structure:

	Byte 0	Byte 1	Byte 2	Byte 3
Device type	Device profile number		Encoder type	
TR(S)N	0x96	0x01	0x02	0x00
TB(S)N	0x96	0x01	0x01	0x00

device_type

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1000	0	Device_type	Long	ro	0x00020196	-	-

6 CANopen profile definition

6.2.2 Object 1001 error_register

Bit	m/o	Designation
0	m	generic error
1	o	current
2	o	voltage
3	o	temperature
4	o	communication error (overrun, error state)
5	o	device profile specific
6	o	Reserved (always 0)
7	o	manufacturer-specific

The error register is the global error register. It sums all errors in bit 0.
Generic, communication and manufacturer-specific errors are supported. In the event of an error, the generic error bit is always set. The error which has occurred can be read off in object 6503 alarms.

error_register

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1001	0	error_register	Byte	ro	0, 0x11, 0x81	-	-

6.2.3 Object 1003 pre_defined_error_field

All alarm messages transmitted via emergency messages are stored in this object. Sub-index 0 contains the error code of the last emergency message stored. Overwriting sub-index 0 with ZERO deletes the stored messages in all sub indixes. This object contains 20 entries at maximum. When this number is exceeded no further error will be stored.

pre-defined error field

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1003	0	Last_error_entry	Byte	rw	0, ... 20	-	-
	1	one_stored_error_msg.					
	2	two_stored_error_msg.					
		...					

Structure_pre_defined_error_field

Byte 0	Byte 1	Byte 2	Byte 3
Alarm code	Custom error code		

See also: CANopen emergency messages structure

6 CANopen profile definition

6.2.4 Object 1005 COB-ID-SYNC

Identifier of the sync message transmitted by the master.
Range and plausibility check takes place according to CiA 301 v 4.2.0, Restricted CAN ID's.

COB-ID-SYNC

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1005	0	COB-ID-SYNC	Long	rw	1...0x7FF	-	0x80

6.2.5 Object 1008 manufacturer_device_name

The name of the device is stored as a string and is output via SDO segment transfer.
e.g. "Encoder TRN Safety"

manufacturer_device_name

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1008	0	manufacturer_device_name	String	ro	Depends on device	-	-

6.2.6 Object 1009 manufacturer_hardware_version

Hardware version of the device. It is stored as a string e.g. "P-0824"
It is output via SDO segment transfer.

manufacturer_hardware_version

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1009	0	manufacturer_hardware_version	String	ro	Depends on device	-	-

6.2.7 Object 100A manufacturer_software_version

Software version of the device. It is stored as a string e.g. "Safety standard"
It is output via SDO segment transfer.

manufacturer_software_version

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
100A	0	manufacturer_software_version	String	ro	Depends on device	-	-

6 CANopen profile definition

6.2.8 Object 1010 store_parameters

By inputting "save" (0x65766173hex resp. 1702257011dez) as a password in the relevant sub-index, the writeable objects are saved to the memory.
The object cannot be changed on writing. Reading the object is possible.
1 (saving through command) is returned.

Which parameters are stored is defined by specifying the sub-index.

Sub-index 01 Storage of all parameters except the 0x2000 to 0x2FFF range.
Sub-index 02 Storage of communication parameters 0x1000 to 0x1FFF.
Sub-index 03 Storage of parameters 0x6000 to 0x9FFF defined in the profile.
Sub-index 04 Storage of the manufacturer-specific range 0x2000 to 0x2FFF (common).
Sub-index 05 Storage of the manufacturer-specific range 0x3000 to 0x3FFF

store_parameters

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1010	0	largest_supported_ sub-index	-	ro	5	-	-
	1	save_all_parameters (except node ID and baudrate)	Long	rd/(rw)	"save" =0x65766173	*	1
	2	save_communica-tion_parameters	Long		"save"	*	1
	3	save_application_ parameters	Long		"save"	*	1
	4	save_LMT_parame-ters (only node ID, baudrate)	Long		"save"	*	1
	5	Save_manufac-turer_parameters	Long		"save"	*	1

* Parameters are backed-up in the memory on inputting the correct password (save = 65 76 61 73).

→ Wait for response of the encoder after saving before resetting the device

6.2.9 Object 1011 restore_default_parameters

On inputting "load" (0x64616F6Chex resp. 1684107116dez) as the password in the relevant sub-index, the default parameters are loaded into the objects of the number group selected with the sub-index in the RAM. Reading the object is possible.
The device has to be set “preoperational” at first.

1 (device restores parameters) is returned.

Which parameters are loaded is defined by specifying the sub-index.

Sub-index 01 Loading of all parameters except the 0x2000 to 0x2FFF range.
Sub-index 02 Loading of communication parameters 0x1000 to 0x1FFF.
Sub-index 03 Loading of parameters 0x6000 to 0x9FFF defined in the profile.
Sub-index 04 Loading of the manufacturer-specific range 0x2000 to 0x2FFF.
Sub-index 05 Loading of the manufacturer-specific range 0x3000 to 0x3FFF.

6 CANopen profile definition

restore_default_parameters

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1011	0	largest_supported_ sub-index	-	ro	5	-	-
	1	load_all_default_ parameters (except node ID and baudrate)	Long	rd/(rw)	"load" =0x64616F6C	*	1
	2	load_communica-tion_parameters	Long		"load"	*	1
	3	load_application_ parameters	Long		"load"	*	1
	4 **	load_LMT_parame-ters (only node ID, baudrate)	Long		"load"	*	1
	5	load_manufacturer_ parameters	Long		"load"	*	1

* On inputting the correct password (load=6C 6F 61 64), the default parameters are loaded from the ROM. In case for a complete load default (COB IDs included): subindex 1 and 4 have to be executed.

** When loading the default values: For all default node IDs which exceed 32 dez (33 ….) the related COB ID 1 and COB ID 2 have to be entered manually in objects 1301, sub 05/sub 06 and 1302, sub 05/sub 06. Otherwise the SRDOs are disabled (e.g. 0x8000 0141/0x8000 0142). For all node IDs up to 32 dez the COB IDs are adopted by the encoder automatically. (See as well objects 1301, 1302 and 2000)

6.2.10 Object 1014 COB-ID-EMCY

Identifier for the emergency message which the encoder transmits on occurrence of an alarm.
After "Load default", the identifier is COB-ID-EMCY + node ID.
If the user changes the COB ID, the node address is no longer added.
Range and plausibility check takes place according to CiA 301 v 4.2.0, Restricted CAN ID's.

COB-ID-EMCY

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1014	0	COB-ID-EMCY	Long	rw	-	*	0x80+node ID

* Default state evaluation, then addition of the node address.

6.2.11 Object 1015 inhibit_time_EMCY

Blocking time to limit the bus load in the event of EMCY messages in quick succession. The resolution is 100 µs per digit.

inhibit_time_EMCY

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1015	0	inhibit_time_EMCY	Word	rw	0...0xFFFF	-	1000

6 CANopen profile definition

6.2.12 Object 1017 producer_heartbeat_time

If a value greater than zero is entered here, the heartbeat message is transmitted on the identifier GUARD COB ID + node ID in the producer_heartbeat_time interval in ms. (GUARD COB-ID = 0x700)

producer_heartbeat_time

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1017	0	producer_heartbeat_ time	Word	rw	0...0xFFFF	-	0

The format of the heartbeat message:

Bit No.	7	6	5	4	3	2	1	0
Content	0	Subscriber status						

Subscriber status:
0 (0x0): BOOT-UP
4 (0x4): STOPPED
5 (0x5): OPERATIONAL
127 (0x7F): PRE-OPERATIONAL

6.2.13 Object 1018 identity_object

This object contains data assigned to the individual encoder. The object is the address for the Layer Setting Service (LSS).
The following data must be entered:

1. Manufacturer ID 0x0000 010D
2. Product code TWK-internal
3. Revision number TWK software revision number
4. Serial number x xxx xxx

The serial number can be written via LSS in factory programming state.

An example for parameterization via LSS for the node ID can be obtained at the end of this document: Node ID Parameterization via LSS Service.

identity_object

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1018	0	largest_supported_ sub-index	-	ro	4	-	-
	1	vendor ID	Long	ro	0x0000 010D	-	-
	2	product_code	Long	ro	0x0000 xxxx*	-	-
	3	revision_number	Long	ro	0x0001 0001	-	-
	4	serial_number	Long	ro(rw)	0	**	-

* depends on device
** Written in factory programming state (wp)

6 CANopen profile definition

6.2.14 Object 1029 error behaviour_object

This object defines the behaviour in the event of an error. The sub-indices are assigned to the error types as follows:

Sub-index	Error type	Bit in error register 1001	Bit in alarm object 6503
1	Communication error	4	-
2	CRC error	7	5
	Supply out of range	7	6
	Sensor error	7	7
4	Redundancy error	7	3

The following settings in object 1029 are possible for the behaviour in the event of an error:

Value	Description
0x00	Switch from OPERATIONAL to PRE-OPERATIONAL
0x01	No NMT status switch
0x02	Switch to STOP state
0x03	Reserve
::::::::	
0x7F	Reserve
0x80	Manufacturer-specific
::::::::	
0xFF	Manufacturer-specific

error_behaviour

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1029	0	largest_supported_ sub-index	-	ro	3	-	-
	1	communication_error	Byte	rw	0, 1, 2	-	0x00
	2	internal_device_error	Byte	ro	0	-	-
*	4	redundancy_error	Byte	rw	0, 1, 2	-	0x01

* Valid for full redundant encoder systems

6 CANopen profile definition

6.3 CANopen Safety process data Objects (SRDO)

The process data are output via two Safety Relevant Data Objects (SRDOs).

Attention:

The SRDO is only active when the object configuration_valid (object 13FE) is set (written with the value 0xA5). The configuration_valid object is stored in the E²PROM. The flag is deleted if one of the Safety Relevant Data Objects is changed.

COB ID structure

MSB										LSB
EN	x	x	x	x	x	x	x	0	COB ID high	COB ID low

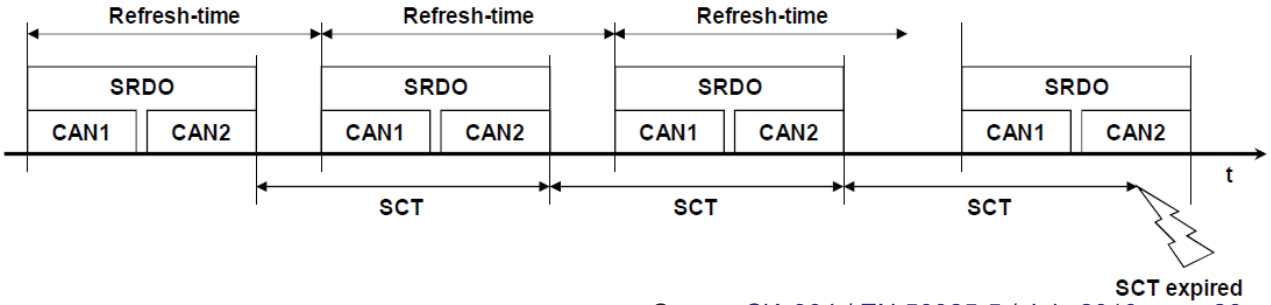
The MSB represents the enable bit.
Bit 31 = 0 (EN) SRDO enabled
Bit 31 = 1 (EN) SRDO disabled (=0x80000”FF+2xID” resp. 0x80000”100+2xID”)

The plausibility of the other bits is not checked.

Only transmission type 254 is supported (Cyclic output with the defined refresh time, see object 1301/1302).

6.3.1 Timing requirements - SCT

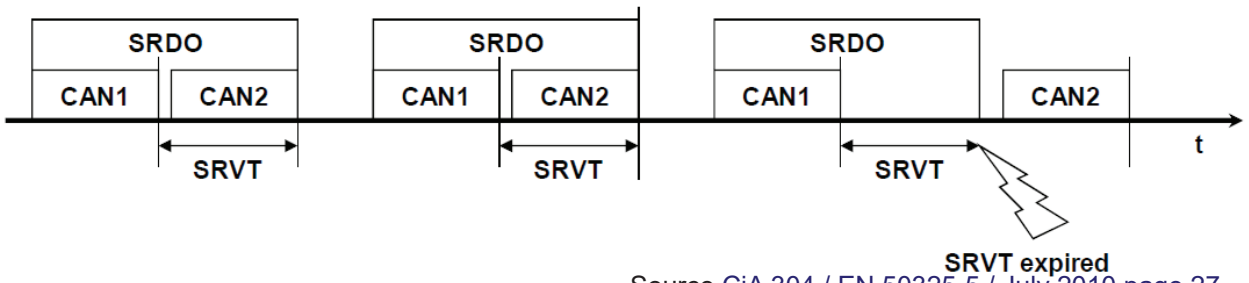
The SRDO is transmitted as defined above and the reception is monitored by the safety controller. The cyclic transmission rate is defined by the refresh-time and monitored with the safety cycle-time (SCT) by the safety controller. If the SCT is elapsed before the corresponding SRDO is received the safety controller shall switch into the safe state. The SCT time used by the safety controller has to be chosen slightly higher than the refresh time to account for sync problems and SW runtime. Further the bit rate effects the SCT time as well and has to be taken into account. Other plausibility checks are not performed. The designer of the overall system is responsible for the meaningfulness of the set parameters.



Source CiA 304 / EN 50325-5 / July 2010 page 26

6.3.2 Timing requirements - SRVT

The SRDO consists of two CAN frames that are transmitted subsequently and the reception is monitored by the safety controller. The reception is monitored with the SR validation time (SRVT). If the SRVT is elapsed before the second CAN data frame (inverted dataframe) is received the safety controller shall switch into the safe state. The SRVT must be selected in such a way that an interruption of the SRDO on the CAN bus does not lead to an error at the safety controller. **SRVT time has to be smaller than SCT time.** Further the bit rate effects the SRVT time as well and has to be taken into account. The designer of the overall system is responsible for the meaningfulness of the set parameters.



Source CiA 304 / EN 50325-5 / July 2010 page 27

6 CANopen profile definition

6.3.3 Object 1301 SRDO1_communication_parameters

The object under this index is only writeable in PRE-OPERATIONAL state.
Modifications will cause 13FE = ‘0’ (must be set ‘0xA5’ after modification). New CRC Checksum 13FF/01 must be calculated and transmitted to the encoder.

The configuration_valid byte is reset after each write access.

SRDO_communication_parameters

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1301	0	largest_supported_ sub-index	-	ro	6	-	-
	1	information_direction	Byte	ro	1	-	-
	2	refresh_time (SCT)	Word	rw	1..65,535 [ms]	-	25
	3	validation_time (SRVT)	Byte	ro	20 [ms]	-	-
	4	transmission_type	Byte	ro	254	-	-
	5 **	COB-ID_1	Long	rw	257..384 0x8000... SRDO deactivated	*	0xFF + (2x node ID)
	6 **	COB-ID_2	Long	rw	257..384 0x8000... SRDO deactivated	*	0x100 + (2x node ID)

* Default state evaluation, then addition of the node address. After overwriting, addition is no longer executed. The configuration_valid byte 13FE is reset. No plausibility check is undertaken for the COB IDs.

** For all node IDs which exceed 32 dez (33 ….) the related COB ID 1 and COB ID 2 have to be entered manually in objects 1301, sub 05/sub 06 and 1302, sub 05/sub 06. Otherwise the SRDOs are disabled (e.g. 0x8000 0141 /0x8000 0142). For all node IDs up to 32 dez the COB IDs are adopted by the encoder automatically. This has to be considered as well when the default value of the node ID exceeds 32 dez and the default values are loaded by 1011, sub 04. (See as well objects 1302, 1011 and 2000).

Only transmission type 254 is supported (Cyclic output with the defined refresh time, see object 1301, sub 02).

To shut off the SRDO, both COB IDs must be disabled. If only one of the two COB IDs is disabled, setting the configuration_valid flag is not possible.

6 CANopen profile definition

6.3.4 Object 1302 SRDO2_speed_parameters

The object under this index is only writeable in PRE-OPERATIONAL state. Modifications will cause 13FE = '0' (must be set '0xA5' after modification). New CRC Checksum 13FF/02 must be calculated and transmitted.

The configuration_valid byte is reset after each write access.

SRDO_speed_parameters

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1302	0	largest_supported_ sub-index	-	ro	6	-	-
	1	information_direction	Byte	ro	1	-	-
	2	refresh_time (SCT)	Word	rw	1..65,535	-	25 [ms]
	3	validation_time (SRVT)	Byte	ro	20 [ms]	-	-
	4	transmission_type	Byte	ro	254	-	-
	5 **	COB-ID_1	Long	rw	257..384 0x8000... SRDO deactivated	*	0x13F + (2x node ID)
	6 **	COB-ID_2	Long	rw	257..384 0x8000... SRDO deactivated	*	0x140 + (2x node ID)

* Default state evaluation, then addition of the node address. After overwriting, addition is no longer executed. The configuration_valid byte is reset. No plausibility check is undertaken for the COB IDs.

** For all node IDs which exceed 32 dez (33 ….) the related COB ID 1 and COB ID 2 have to be entered manually in objects 1301, sub 05/sub 06 and 1302, sub 05/sub 06. Otherwise the SRDOs are disabled (e.g. 0x8000 0181/0x8000 0182. For all node IDs up to 32 dez the COB IDs are adopted by the encoder automatically. This has to be considered as well when the default value of the node ID exceeds 32 dez and the default values are loaded by 1011, sub 04. (See as well objects 1301, 1011 and 2000).

Only transmission type 254 is supported (Cyclic output with the defined refresh time, see 1302, sub 02).

To shut off the SRDO, both COB IDs must be disabled. If only one of the two COB IDs is disabled, setting the configuration_valid flag is not possible.

6 CANopen profile definition

6.4 CAN open process data objects (PDO)

6.4.1 Object 1800 Transmit PDO 1

Range and plausibility check takes place according to CiA 301 v 4.2.0, Restricted CAN ID's. The inhibit time has a resolution of 100 µs. The event timer has a resolution of 1 ms.

Transmit_PDO_1

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1800	0	largest_supported_ sub-index	-	ro	5	-	-
	1	COB ID	Long	rw	-	*	0x180+ID
	2	transmission type	Byte	rw	0...240**, 252, 253, 254	-	253
	3	inhibit time	Word	rw	0..65,535 [ms]	-	0
	4	reserved	-	-	-	-	-
	5	event_timer	Word	rw	0..65,535 [ms]	-	0

* On reading (upload), the node address is added to the selected identifier.
** 0 ≤ n ≤ 240: Every nth sync-message the PDO2 is transmitted.

252: Value is picked-up with sync and transmitted with remote frame (cyclic transmission via object 6200)
253: Value is picked-up and transmitted with remote frame (cyclic transmission via object 6200)
254: Value is picked-up and transmitted with every change of value
Disable PDO1 with 0x80000180+ID as COB ID

6.4.2 Object 1801 Transmit PDO 2

Range and plausibility check takes place according to CiA 301 v 4.2.0, Restricted CAN ID's.
The inhibit time has a resolution of 100 µs. The event timer has a resolution of 1 ms.

Transmit_PDO_2

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1801	0	largest_supported_ sub-index	-	ro	2	-	-
	1	COB ID	Long	rw	-	*	0x280+ID
	2	Transmission type	Byte	rw	0...240**, 252***, 253***, 254***	-	1
	3***	inhibit time	Word	rw	0..65,535 [ms]	-	0
	4***	reserved	-	-	-	-	-
	5***	event_timer	Word	rw	0..65,535 [ms]	-	0

* On reading (upload), the node address is added to the selected identifier.
** 0 ≤ n ≤ 240: Every nth sync-message the PDO2 is transmitted.

*** valid for Multiturn Encoder TR(S)N (SIL2/3) with CANopen safety (TRN...S4/S41...) and Singleturn TBSN (SIL3) with CANopen safety (TBSN...S41..) only, for other encoders: please contact TWK for this functionality

252: Value is picked-up with sync and transmitted with remote frame
253: Value is picked-up and transmitted with remote frame
254: Value is picked-up and transmitted with every change of value

Disable PDO2 with 0x80000280+ID as COB ID

6 CANopen profile definition

6.5 Mapping objects

6.5.1 Object 1381 SRDO1_mapping_parameters

The parameter contains the following coding for each "mapping" object:

Byte 0	Byte 1	Byte 2	Byte 3
Index		Sub-index	Length

The length is specified as the number of bits in hex coded form.

SRDO_1_mapping_parameters

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1381	0	largest_supported_ sub_index	-	ro	8	-	-
	1	first_SRDO_map- ping_object	Long	ro	0x6120 0108	-	-
	2	second_SRDO_ mapping_object	Long	ro	0x6121 0108	-	-
	3	third_SRDO_map- ping_object	Long	ro	0x6120 0208	-	-
	4	fourth_SRDO_map- ping_object	Long	ro	0x6121 0208	-	-
	5	fifth_SRDO_map- ping_object	Long	ro	0x6120 0308	-	-
	6	sixth_SRDO_map- ping_object	Long	ro	0x6121 0308	-	-
	7	seventh_SRDO_ mapping_object	Long	ro	0x6120 0408	-	-
	8	eighth_SRDO_map- ping_object	Long	ro	0x6121 0408	-	-

6 CANopen profile definition

6.5.2 Object 1382 SRDO2_mapping_parameters

The parameter contains the following coding for each "mapping" object:

Byte 0	Byte 1	Byte 2	Byte 3
Index		Sub-index	Length

The length is specified as the number of bits in hex coded form.

SRDO_2_mapping_parameters

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1382	0	largest_supported_ sub_index	-	ro	4	-	-
	1	first_SRDO_map- ping_object	Long	ro	0x6124 0108	-	-
	2	second_SRDO_ mapping_object	Long	ro	0x6125 0108	-	-
	3	third_SRDO_map- ping_object	Long	ro	0x6124 0208	-	-
	4	fourth_SRDO_map- ping_object	Long	ro	0x6125 0208	-	-

6 CANopen profile definition

6.5.3 Object 1A00 transmit_PDO_1_mapping

The parameter contains the following coding for each "mapping" object:

Byte 0	Byte 1	Byte 2	Byte 3
Index		Sub-index	Length

The length is specified as the number of bits in hex coded form.

mappable objects are:

0x6004 sub 0x00 length 0x04
0x600C sub 0x00 length 0x04
0x6030 sub 0x01 length 0x02
0x6120 sub 0x01 length 0x01
0x6120 sub 0x02 length 0x01
0x6120 sub 0x03 length 0x01
0x6120 sub 0x04 length 0x01
0x6124 sub 0x01 length 0x01
0x6124 sub 0x02 length 0x01

Else the SDO transfer is aborted with abort code 0x 0604 0041 - Object cannot be mapped to the PDO.
If the PDO length is exceeded abort code 0x 0604 0042 is provided by SDO transfer - The number and length of the objects to be mapped exceed PDO length.

transmit_PDO_1_mapping

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1A00	0	largest_supported_ sub_index	-	rw*	0x00 (mapping disabled) to 0x08 (sub index 0x01...0x08 valid)	- [t]	0x02
	1	PDO1_mapping_ object1	Long	rw*	0x6004 0020 (see above for mappable objects)	-	0x6004 0020
	2	PDO1_mapping_ object2	Long	rw*	0x6030 0110 (see above for mappable objects)	-	0x6030 0110

* Only valid for Multiturn (TRN) CANopen Safety SIL2 (TRN...S4...),
else e.g.: TBSN/TRSN, TBN...C3/S4 or TRN...C3... these objects are: read only / ro
[t] upon changing this parameter the value defines which subs become valid for mapping

Note:

The following procedure shall be used for re-mapping, which may take place during the NMT state pre-operational and during the NMT state operational:

1. Destroy TPDO by setting 1800/01 to 0x 8000 0180 + Node ID as COB ID
2. Disable mapping by setting sub-index 0x00 to 0x00
3. Modify mapping by changing the values of the corresponding sub-indices
4. Enable mapping by setting sub-index 0x00 to the number mapped objects
5. Create TPDO by setting 1800/01 to 0x180+Node ID as COB ID

6 CANopen profile definition

6.5.4 Object 1A01 transmit_PDO_2_mapping

The parameter contains the following coding for each "mapping" object:

Byte 0	Byte 1	Byte 2	Byte 3
Index		Sub-index	Length

The length is specified as the number of bits in hex coded form.

mappable objects are:

0x6004 sub 0x00 length 0x04
0x600C sub 0x00 length 0x04
0x6030 sub 0x01 length 0x02
0x6120 sub 0x01 length 0x01
0x6120 sub 0x02 length 0x01
0x6120 sub 0x03 length 0x01
0x6120 sub 0x04 length 0x01
0x6124 sub 0x01 length 0x01
0x6124 sub 0x02 length 0x01

Else the SDO transfer is aborted with abort code 0x 0604 0041 - Object cannot be mapped to the PDO.
If the PDO length is exceeded abort code 0x 0604 0042 is provided by SDO transfer - The number and length of the objects to be mapped exceed PDO length.

transmit_PDO_2_mapping

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
1A01	0	largest_supported_ sub_index	-	rw*	0x00 (mapping disabled) to 0x08 (sub index 0x01...0x08 valid)	- [t]	0x02
	1	PDO2_mapping_ object1	Long	rw*	0x6004 0020 (see above for mappable objects)	-	0x6004 0020
	2	PDO2_mapping_ object2	Long	rw*	0x6030 0110 (see above for mappable objects)	-	0x6030 0110

* Only valid for Multiturn (TRN) CANopen Safety SIL2 (TRN...S4...),
else e.g.: TBSN/TRSN, TBN...C3/S4 or TRN...C3... these objects are read only ro
[t] upon changing this parameter the values defines, which subs become valid for mapping

Note:

The following procedure shall be used for re-mapping, which may take place during the NMT state pre-operational and during the NMT state operational:

1. Destroy TPDO by setting 1801/01 to 0x 8000 0280 + Node ID as COB ID
2. Disable mapping by setting sub-index 0x00 to 0x00
3. Modify mapping by changing the values of the corresponding sub-indices
4. Enable mapping by setting sub-index 0x00 to the number mapped objects
5. Create TPDO by setting 1801/01 to 0x280 + Node ID as COB ID

6 CANopen profile definition

6.6 Safety CAN objects

6.6.1 Object 13FE configuration_valid

This parameter is reset (= 0) each time a "safety relevant parameter" is accessed. Entering 0xA5 switches the configuration to valid. In the case of an invalid value (not 0 or 0xA5) in the configuration_valid flag or incorrect setting of the Safety parameters, write access is rejected and the SRDOs are not transmitted in OPERATIONAL mode.

Attention:

The flag is automatically reset by writing to the safety position data in area $61xx_h$.
The flag can only be activated if the data valid flag of the safety position parameter safety_configuration_valid, object $61FE_h$, is activated.

The parameter is stored in the E²PROM.

configuration_valid

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
13FE	0	configuration_valid	Byte	rw	0 / 0xA5	-	0x0

6.6.2 Object 13FF safety_configuration_checksum

This parameter contains the checksum crosswise through the safety CAN parameters in objects 1301 and 1302 according to the table below. The checksum is written by the master. The checksum is checked on setting the configuration_valid flag. If no correspondence with the checksum stored in this object is ascertained, setting the flag is blocked (configuration_valid remains ZERO). To calculate the CRC checksum you can use the special TWK program.

Link to the program on www.twk.de: Look for the desired device → “Download” and “Software Safety CRC”.
You can also find it on the website at "SUPPORT AND SERVICE" → “DOCUMENTATION” → “Software" and then "SafetyCRC Software”.
Or use the following link:
https://www.twk.de/files/001_CANopen/001_CRC/001_Programm/SafetyCRC_V2.0.9.zip
Description file: CRC 14076. Please ask our technicians for xml files for special versions of encoders or look at www.twk.de.

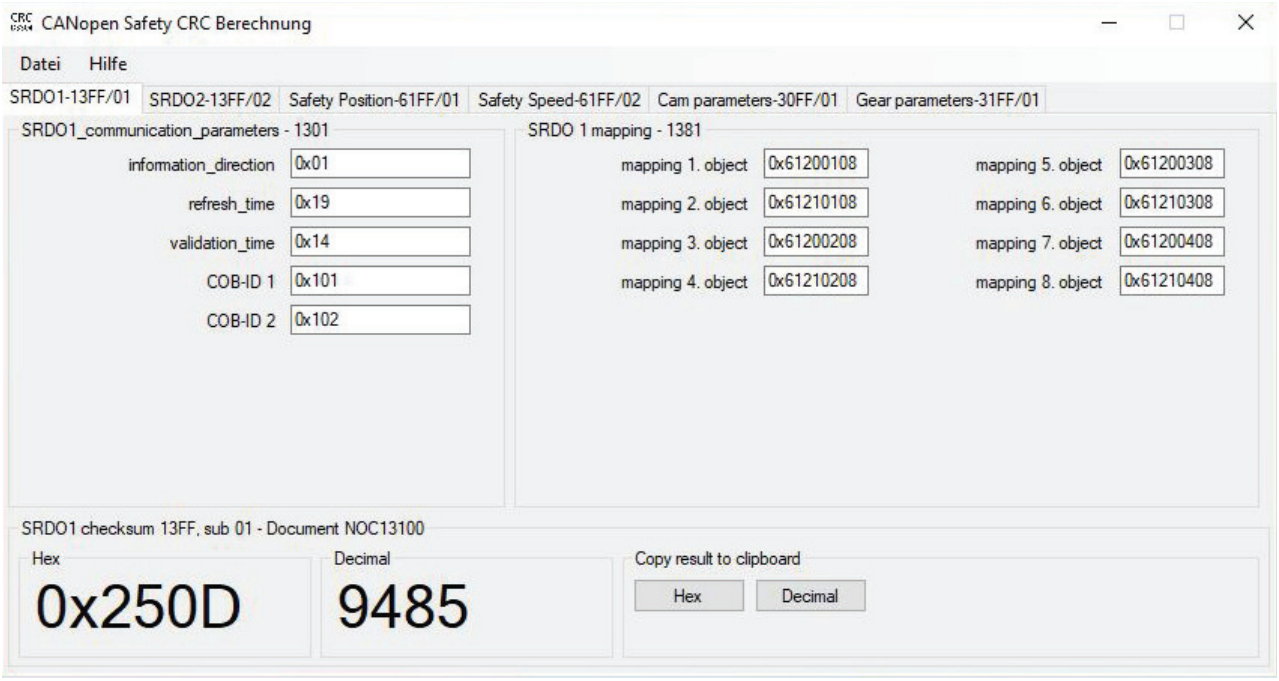
Attention: The parameter can only be written if the data valid flag of the safety position parameter safety_configuration_valid, object 61FE, is activated. See example for parameterization at the end of this document.

safety_configuration_checksum

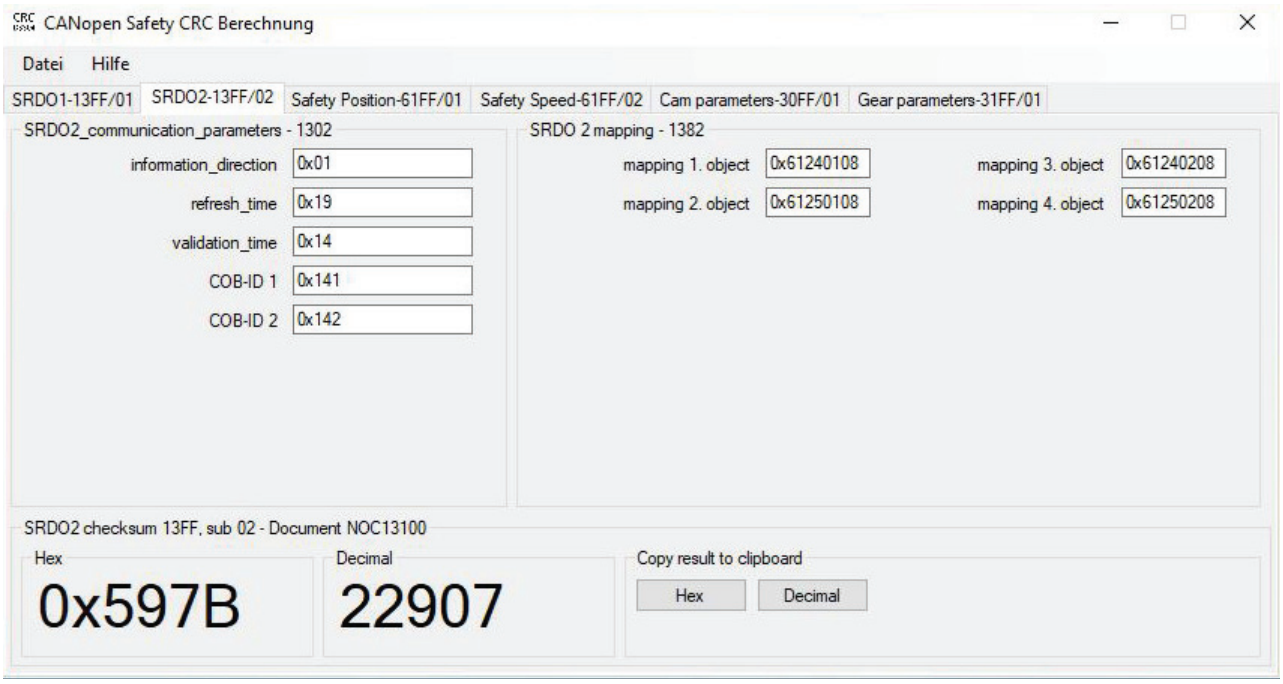
Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
13FF	0	largest_supported_ sub-index	Byte	ro	2	-	2
	1	SRDO1_checksum (for changes in object 1301)	Word	rw	0 ... 0xFFFF	-	0x250D
	2	SRDO2_checksum (for changes in object 1302)	Word	rw	0 ... 0xFFFF	-	0x597B

6.6.3 TWK Program for calculation of the checksum

Generator polynomial: $2^{16} + 2^{12} + 2^5 + 1 = 0x11021$
Initial value: 0x0000
Final XOR: no



6 CANopen profile definition



6 CANopen profile definition

6.7 LMT objects

6.7.1 Object 2000 node ID

The node address of the encoder.

The parameter only becomes effective after saving with object 1010, sub 04 and a power on/NMT reset.

Remark: It is not possible to save this parameter with object 1010, sub 01 (save all parameters).

The parameter is not reset to the default value with load_default 1011, sub 01 but with 1011, sub 04.

For all node IDs which exceed 32 dez (33 ….) the related COB ID 1 and COB ID 2 have to be entered manually in objects 1301, sub 05/sub 06 and 1302, sub 05/sub 06. Otherwise the SRDOs are disabled (i.e. 0x8000 0122 /0x8000 0123, chap. 9.6). For all node IDs up to 32 dez the COB IDs are adopted by the encoder automatically. This has to be considered as well when the default value of the node ID exceeds 32 dez and the default values are loaded by 1011, sub 04. (See as well objects 1301, 1302 and 1011).

node ID

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
2000	0	node ID	Byte	rw	1 ...127 (0x01 ...0x7F) 255 (0xFF)*	-	0x01

* Encoder is in spare part mode and can only be reached by LSS.

6.7.2 Object 2001 bit_rate

Baud rate of the CAN bus.

This object can also be changed using the Layer Setting Service.

The bit rate index is set according to the following table:

Index	Baud rate [kBaud/s]
0	1000
1	800
2	500
3	250
4	125
5	100
6	50
7	20

The parameter only becomes effective after saving with object 1010, sub 04 and a power on reset.
Remark: It is not possible to save this parameter with object 1010, sub 1 (save all parameters).

The parameter is not reset to the default value with load_default 1011, sub 01 but with 1011, sub 04.

bit_rate

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
2001	0	bit_rate	Byte	rw	0 ...7	-	3 (250 kBaud)

6.8 Objects according to profile definition

6.8.1 Object 6000 operating_parameters

Operating mode byte for the sensor.

Note: The code direction for SRDO (safety communication) has to be set in object 6100 sub 1.

The state table:

Bit	m/o	Designation	0	1
0	m	Code direction	CW	CCW
1	o	Set firmly to zero		
2	o	scaling_function_control	Inactive	Active
3-11	o	Set firmly to zero		
12	o	slewing ring function	Inactive	Active

operating_parameters

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6000	0	operating_parameters	Word	rw*	0x0, 0x01, 0x04, 0x05 0x1004, 0x1005	Sen	0x0,4

* At monoturn (TBN) encoders with CANopen Safety Profile SIL2 (TBN...S4...), or SIL3 encoders TBSN/TRSN: read only ro
It is mirrored from the safety Object 6100

6 CANopen profile definition

6.8.2 Object 6001 measuring_units_per_revolution

Number of steps per revolution.

measuring_units_per_revolution

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6001	0	measuring_units_per_ revolution	Long	rw*	depending on model and perfo-mance e.g.: 4096	-	-

* At monoturn (TBN) encoders with CANopen Safety Profile SIL2 (TBN...S4...), or SIL3 encoders TBSN/TRSN: read only ro

6.8.3 Object 6002 total_measuring_range_in_measuring_units

Total measuring range in measuring units.
This value is only relevant for TRN Multiturn devices. For any TBN devices it is automatically set according to object 6001.
For TRN...S4... (SIL2) or TBSN/TRSN...S41... (SIL3) safe measuring range for SRDO output please refer to object 3300.

total_measuring_range_in_measuring_units

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6002	0	total_measuring_ran-ge_in_measuring_units	Long	rw*	depending on model and per-formance e.g.: 16777216	-	-

* At monoturn (TBN) encoders with CANopen Safety Profile SIL2 (TBN...S4...), or SIL3 encoders TBSN/TRSN: read only ro

6.8.4 Object 6003 preset_value

Presets the current PDO position output to the value specified in object 6003.

Note: The preset value for SRDO (safety communication) has to be set in object 6100 sub 2 safety_preset_value.

preset_value

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6003	0	preset_value	Long	rw *	0…(obj. 6002)-1	-	-

* At monoturn (TBN) encoders with CANopen Safety Profile SIL2 (TBN...S4...), or SIL3 encoders TBSN/TRSN: read only ro
It is mirrored from the safety object 6100 sub 2

6 CANopen profile definition

6.8.5 Object 6004 position_value

Position value. This value is the measured value.
The parameter is provided as a position value by the sensor. This object is updated cyclically.

The following is only valid for Singleturn (TBN...S4...) CAN open Safety (SIL2):
The parameter is taken over from the safety area and contains the secure measured value.

position_value

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6004	0	position_value	Long	ro	0..(obj 6002) -1	-	-

Remark: Object 600C contains the position raw data with a resolution depending on device. 600C can be used for calculating the speed signal with max. resolution (see object 6101).

6.8.6 Object 600C raw_position_value

Index	Sub	Description	Length-COM	Memory Type	Range/value	Action	Default
600C	0	raw_position_value	Long	ro	0...0xFFFF	-	-

6.8.7 Object 6030 speed_value

Speed value. The dimension is digits per object 6031 sub-index 2 in ms with a resolution depending on speed_ source_selector, speed_multiplier and speed_divider. The format is 16bit signed. This object is updated cyclically.

speed_value

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6030	0	largest_supported_ sub_index	Byte	ro	1	-	-
	1	speed_value_channel1	Word	ro	0...0xFFFF	-	-

6.8.8 Object 6031 speed_parameter

speed_parameter

Index	Sub	Description	Length-COM	Memory Type	Range/value	Action	Default
6031	0	largest_supported_sub_in-dex	Byte	ro	4	-	-
	1	speed_source_selector [t]	Byte	rw*	1, 2	-	2
	2	speed_ integration _time	Word	rw*	1 to 1000 [ms]	-	100
	3	speed_multiplier	Word	rw*	1 to 65535	-	100
	4	speed_divider	Word	rw*	1 to 65535		10

* At monoturn (TBN) encoders with CANopen Safety Profile SIL2 (TBN...S4...), or SIL3 encoders TBSN/TRSN: read only ro

They are mirrored from the safety Object 6101

[t] 1 = scaled position from object 6004. 2 = Raw data of position from object 600C, no scaling factor enabled. The related resolution of setting 1 or 2 for the speed signal depends on the device

6 CANopen profile definition

6.9 Safety Objects according to profile definition

6.9.1 Object 6100 safety_position_configuration_parameters

The object defines the behaviour of the position registration system in the safety area.
Modifications will cause 61FE = ‘0’ (must be set ‘0xA5’ after modification). New CRC Checksum 61FF/01 must be calculated and transmitted to the encoder.

The parameter can only be changed in PRE-OPERATIONAL state.

safety_position_configuration_parameters

Index	Sub	Description	Length Com	Memory Type	Range/value	Action	Default
6100	0	largest_supported_ sub_index	Byte	ro	3	-	-
	1	safety_code_sequence	Word	rw	0 to 1 (0=cw, 1=ccw)	-	0
	2	safety_preset_value	Long	rw	0...(obj.3300/02)-1 0...(obj. 6002)-1 [t]		0
	3	Safety_HR*_preset_ value		ro	0x7FFFFFFFFF FFFFFF		

* High resolution
[t] for TBN...S4... (with CANopen safety Profile SIL2) encoder

Note for encoders TBN/TRN-S4 bearing the product codes 0x6000, 0x6001 and 0x6002 (see Object 1018 sub2):

If objects 6100, sub01, sub02 and sub03 are modified, objects 6101, sub01, sub02 and sub03 are modified automatically and vice versa.
Therfore upon modifying values in object 6100, cheksums 61FF sub 1 and 61FF sub 2 have to be modified with the appropriate values.
For all other products with product codes not mentioned above Objects 6100 and 6101 are completly independent from each other.

6.9.2 Object 6101 safety_speed_configuration_parameters

The object defines the behaviour of speed measurement in the safety area.
Modifications will cause 61FE = '0' (must be set '0xA5' after modification). New CRC Checksum 61FF/02 must be calculated and transmitted to the encoder.
The parameter can only be changed in PRE-OPERATIONAL state.

If 6100, sub01, sub02 and sub03 is modified, 6101, sub01, sub02 and sub03 is modified automatically and vice versa.

safety_speed_configuration_parameters

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6101	0	largest_supported_ sub_index	Byte	ro	7	-	-
	1	safety_code_sequence	Word	rw	0, 1 (0=cw, 1=ccw)	-	0
	2	safety_preset_value	Long	rw	0...(obj.3300/02)-1 0...(obj. 6002)-1 [t]	-	0
	3	safety_preset_HR_value	Long64	ro	0x7FFFFFFFFF FFFFFF	-	-
	4	safety_speed_source *	Byte	rw/ro	1, 2	-	2
	5	safety_speed_ inte-gration _time	Word	rw	1…1000 [ms]	-	100
	6	safety_speed_multi-plier	Word	rw	1 … 65535	-	100
	7	safety_speed_divider	Word	rw	1 … 65535	-	10

* 1 = scaled position from object 6004 for TBN...S4... (SIL2)
1 = scaled position from object 6120 for TRN...S4... (SIL2) or TBSN/TRSN...S41... (SIL3) encoders
2 = Raw data of position from object 600C, no scaling factor enabled.
The related resolution of setting 1 or 2 for the speed signal depends on the device

[t] for TBN...S4... (with CANopen safety Profile SIL2) encoder

Note for encoders TBN/TRN-S4 bearing the product codes 0x6000, 0x6001 and 0x6002 (see Object 1018 sub2):

If objects 6100, sub01, sub02 and sub03 are modified, objects 6101, sub01, sub02 and sub03 are modified automatically and vice versa.
Therfore upon modifying values in object 6101, cheksums 61FF sub 1 and 61FF sub 2 have to be modified with the appropriate values.
For all other products with product codes not mentioned above Objects 6100 and 6101 are completly independent from each other.

6 CANopen profile definition

6.9.3 Object 6120 safety_standard_position_value

The object contains the current position. This object is used in the mapping structure for data output within the SRDO. In the event of access to individual objects, it must be noted that the consistency of the measured value is not ensured.

safety_standard_position_value

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6120	0	largest_supported_sub_index	Byte	ro	4	-	-
	1	safety_position_value_byte1	Byte	ro	0...0xFF	-	-
	2	safety_position_value_byte2	Byte	ro	0...0xFF	-	-
	3	safety_position_value_byte3	Byte	ro	0...0xFF	-	-
	4	safety_position_value_byte4	Byte	ro	0...0xFF	-	-

6.9.4 Object 6121 safety_inverted_position_value

The object contains the current bit-inverted position. This object is used in the mapping structure for data output within the SRDO. In the event of access to individual objects, it must be noted that the consistency of the measured value is not ensured.

safety_inverted_standard_position_value

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6121	0	largest_supported_sub_index	Byte	ro	4	-	-
	1	safety_inverted_position_value_byte1	Byte	ro	0...0xFF	-	-
	2	safety_inverted_position_value_byte2	Byte	ro	0...0xFF	-	-
	3	safety_inverted_position_value_byte3	Byte	ro	0...0xFF	-	-
	4	safety_inverted_position_value_byte4	Byte	ro	0...0xFF	-	-

6.9.5 Object 6124 safety_speed_value

The object contains the actual calculated speed value. This object is used in the mapping structure for data output within the SRDO. In the event of access to individual objects, it must be noted that the consistency of the measured value is not ensured.

safety_speed_value

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6124	0	largest_supported_sub_index	Byte	ro	2	-	-
	1	safety_speed_value_byte1	Byte	ro	0 ..0xFF	-	-
	2	safety_speed_value_byte2	Byte	ro	0 ..0xFF	-	-

6 CANopen profile definition

6.9.6 Object 6125 safety_inverted_speed_value

The object contains the current bit-inverted measured speed value. This object is used in the mapping structure for data output within the SRDO. In the event of access to individual objects, it must be noted that the consistency of the measured value is not ensured.

safety_inverted_speed_value

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6125	0	largest_supported_sub_index	Byte	ro	2	-	-
	1	safety_inverted_speed_value_byte1	BYTE	ro	0 ..0xFF	-	-
	2	safety_inverted_speed_value_byte2	BYTE	ro	0 ..0xFF	-	-

6.9.7 Object 61FE safety_configuration_valid

The object is the confirmation flag stating that the configuration is valid. On setting the flag, the consistency of the CRC checksum and the set parameters is checked.

This parameter can only be changed in PRE-OPERATIONAL state.

When the CRC checksum is not correct (i.e. in case of changes of parameters without changing the CRC checksum or a wrong checksum is transmitted to the encoder) 61FE can not be set to 0xA5. An error message comes: (e.g. 80 fe 61 00 22 00 00 08).

Attention:

safety_configuration_valid

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
61FE	0	safety_configuration_valid	Byte	rw	0/0xA5	-	0xA5

6.9.8 Object 61FF safety_configuration_signature

The object contains the checksum across the safety encoder parameters in objects 6100 and 6101. The checksum must be rewritten in the event of changes by the user. The checksum is checked on setting the safety_configuration_valid to 0xA5. If no correspondence with the checksum stored in this object is ascertained, setting the flag is blocked (safety_configuration_valid remains ZERO). To calculate the CRC checksum you can use the special TWK program.

Link to the program on www.twk.de: Look for the desired device → “Download” and “Software Safety CRC”. You can also find it on the website at "SUPPORT AND SERVICE" → “DOCUMENTATION” → “Software" and then "SafetyCRC Software”.
Or use the following link:
https://www.twk.de/files/001_CANopen/001_CRC/001_Programm/SafetyCRC_V2.0.9.zip
Description file: CRC 14076. Please ask our technicians for xml files for special versions of encoders or look at www.twk.de.

The parameter can only be changed in PRE-OPERATIONAL state.
safety_configuration_signature

Index	Sub	Description	Length' COM	Memory Type	Range/value	Action	Default
61FF	0	largest_supported_ sub_index	Byte	ro	2	-	2
	1	SRDO1_signature (for changes in object 6100)	Word	rw	0 ..0xFFFF	-	0x....*
	2	SRDO2_signature (for changes in object 6101)	Word	rw	0 ..0xFFFF	-	0x....*

* Depends on default values in objects 6100 and 6101. Pay attention to the different values which have to be tagged in the TWK program for the CRC checksums for sub 1 and sub 2.

6.10 Objects for achieving compatability

6.10.1 Object 6200 Cyclic Timer

In the case of values > 0, the object position value 6004 is transmitted cyclically with the value of the cyclic timer in ms on PDO 1. This object is logically set equal to event_timer of PDO1 (Object 1800 sub5).

Cyclic Timer

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6200	0	cyclic_timer	Word	rw	0...0xFFFF	-	0

6 CANopen profile definition

6.11 Diagnosis objects

6.11.1 Object 6500 operating_status

Current sensor status. This is a representation of object 6000.
The parameter is provided by the sensor.

operating_status

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6500	0	operating_status	Word	ro	Object 6000*	-	-

* 0x00=CW and inactive, 0x01=CCW and inactive, 0x04=CW and active (standard), 0x05= CCW and active, for slewing ring applications: 0x1004 (CW), 0x1005 (CCW) (bit 12 activated, see Object 6000 for further details)

6.11.2 Object 6501 singleturn_resolution

Maximum single turn resolution

singleturn_resolution

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6501	0	singleturn_resolution	Long	ro	max. 65536	-	-

6.11.3 Object 6502 number_of_distinguishable_revolutions

Maximum number of distinguishable revolutions.

number_of_distinguishable_revolutions

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6502	0	number_of_distinguis-hable_revolutions	WORD	ro	4096	-	-

6.11.4 Object 6503 alarms

Internally, there is only one error byte. If an alarm occurs, an emergency message is transmitted. During SDO upload, the error byte is loaded into the object's MSB.

The following errors are evaluated:

Bit	Error type
0	Not used
1	Not used
2	Not used
3	Device error
4	Not used
5	CRC parameter error
6	Supply out of range
7	Sensor error

Extended error coding is available for certain errors in this byte in the emergency message and in object 1003 pre_defined_error_field.

CRC error: The alignment parameters and the CAN interface parameters are monitored by CRC. In the case of an error in one of the areas, this flag is set.

Sensor error: Position sensor error or the measured values of the sensors read-in in redundant form lie too far apart. Or the position controller discovers a scanning error.

Device error: In case of full redundant systems each systems controls the other system. If one system recognizes that the other system doesn't work anymore this error will be generated and transmitted via CAN. A hardware error (look at fatal errors) is not transmitted via CAN.

Alarms

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6503	0	alarms	Word	ro	-	See above	-

6.11.5 Object 6504 supported_alarms

Supported alarm messages.
This is a representation of the error displays possible in the case of index 6503.

supported_alarms

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6504	0	supported_alarms	Word	ro	0xE8	-	-

6.11.6 Object 6506 supported_warnings

Supported warning messages.
No warnings are supported. Object 6505 can therefore be omitted.

supported_warnings

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6506	0	supported_warnings	Word	ro	0	-	-

6.11.7 Object 6507 profile_and_software_version

The profile and software version of the encoder.

profile_and_software_version

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6507	0	profile_and_software_version	Long	ro	0x400100102	-	-

6 CANopen profile definition

6.11.8 Object 6508 operating_time

Not supported.

operating_time

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6508	0	operating_time	Long	ro	0xFFFF FFFF	-	-

6.11.9 Object 6509 offset_value

Output offset. In our encoders, this is the zero point cell.
The parameter is evaluated by the sensor (only used internally).

offset_value

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
6509	0	offset_value	Long	ro	0..(obj 6002) -1	SEN	-

6.11.10 Object 650A module_identification

The manufacturer offset is used as the zero point parameter for synchronising the two nodes' position data. This parameter is written via the factory programming during system alignment.

module_identification

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
650A	0	largest_supported_ sub-index	-	ro	1	-	-
	1	manufacturer_off- set_value	Long	ro	0..(obj 6001) -1	-	*

* Written in factory programming state.

6.11.11 Object 650B serial_number

The serial number is written with the factory programming.

serial_number

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
650B	0	serial_number	Long	wp	0....	*	-

* Written in factory programming state.

6 CANopen profile definition

6.11.12 Object 650D absolute_accuracy

Displays the accuracy of the measuring value.

absolute_accuracy

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
650D	0	absolute_accuracy	Byte	ro	.....	SEN	10 *

* Depending on device

6.11.13 Object 650E device_capability

Defines the operating capability of the encoder. The entry is carried out according to the following table:

Name	Bit	Value	Definition
ec: Encoder class	2:0	0b000 (b=binary)	Reserved
		0b001	Class 1
		0b010	Class 2
		0b011	Class 3
		0b100 to 0b111	Reserved
rsl: Resolution	3	0	Low
		1	High
r: Reserved	4	Reserved (always 0)	
saf: Safety	5	0	Safety not supported
		1	Safety supported
st: Safety type	7:6	0b00	CANopen Safety
		0b01 to 0b11	Reserved
r: Reserved	11:8	Reserved (always 0)	
msc 1: Manufacturer-specific capability 1: slewing ring functionality	12	0	slewing ring functionality not supported
		1	slewing ring functionality supported see Objects 3100 (safety) and 3102 (non safety) for programming of slewing ring functionality.
msc 2 to 4: Manufacturer-specific capability 1 to 4	15:13	0	Manufacturer-specific capability 2 to 4 disabled
		1	Manufacturer-specific capability 2 to 4 enabled

Bit assignment parameter definition

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
msc4	msc3	msc2	msc1	r				st		saf	r	rsl	ec		

device_capability

Index	Sub	Description	Length COM	MEM Type	Range/value	Action	Default
650E	0	device_capability	Long	ROM	0x23 safety / 0x1023 (with safety slewing ring functionality)		-
					0x03 standard / 0x1003 (with slewing ring functionality)		

Class 3 encoder with CANopen Safety.

7 Manufacturer Specific Objects

7.1 Object 3100 safety_gear_configuration

This Object is for setting the safety gear parameters for safety SRDO transmission.

Only valid for TRN S4 version with slewing ring function (Version "S" in order code number between ST and MT resolution). If you have a device not marked with S4 on the order designation, please refer to Object 1302. Output of slewing ring position modifications will cause 31FE = "0" (must be set "A5" after modification). New CRC Checksum 31FF/01 must be calculated and transmitted to TRN.

Note: Maximum gear ratio i =1024 [dez], Minimum gear ratio i =1, i = slew_gear / measure_gear
Bigger or smaller values are not allowed. A plausibilty check ensures that the valid flag 31FE cannot be set. Error code 0800 0022 - data not stored to device is transmitted via SDO transfer if the plausibility check fails.

Adjustment of gear parameters

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
3100	0	largest_supported_ sub-index	-	ro	0x03	-	-
	1	safety_slew_gear [teeth]	LONG	rw	0x0 ... 0x FFFF FFFF		0x0
	2	safety_measure_ gear [teeth]	LONG	rw	0x0 ... 0x FFFF FFFF		0x0
	3	safety_measuring_ range	LONG	rw	0x0 ... 0x FFFF FFFF and ≤ [3300/01] x i (i= slew_gear / measure_gear)		0x0

safety_slew_gear	Number of teeth driving gear (slewing)
safety_measure_gear	Number of teeth of encoder tooth gear
safety_measuring_range	Number of steps output for 1 turn of the slewing ring

Example for 3100 sub3:

Condition: Object 3300 sub 01 = 0x4000 (16384[dez])

If i = 0x03 (3[dez]) (for example 30 / 10) the maximum of the measuring range is 0xC000 (49,152[dez]).
If i = 0x01 (1[dez]) → measuring range is ≤ 0x4000 (16384[dez]).

Depending on ratio i, 3100 sub3 has to be matched. This is reasonable for providing every single step of resolution.

The gear parameters in object 3100 can be set easily after disabling of 31FE and transmission of the related checksum via 31FF and then setting 31FE to 0xA5.

In Order to completely deactivate the slewing ring function and achieve an encoder behavior exactly as a normal binary encoder, all parameters of 3100 (sub1, sub 2 and sub 3) have to be set to zero.

Checksum 31FF has to be set accordingly and valid flag 31FE has to be set 0xA5.
After modifying the gear parameters it has to be ensured that a "safe all" command via object 1010 sub1 is executed. Otherwise after switching the supply power off and back on, an error is reported (The encoder is in a safe state, no position signal via SRDO):
0xFF 0xFF 0x81 0x00 0x20 0x01 0x02 0x02
The last saved parameter set is loaded. The parameter set that was not saved and the recorded position is lost. The error must be eliminated by setting the Data Valid Flag to 0x0 and setting it to 0xA5 again and a subsequent reset.

After modifying any safety_gear_configuration parameter, the output position is undefined within the code range of the output resolution (object 3100/3). By setting a preset value (object 6100/2) a defined initial position is set. This parameter is verified by the checksum of the profile specific parameters.

7 Manufacturer Specific Objects

7.1.1 Object 31FE safety_gear_data_valid

This parameter is reset (= 0x0) each time a "safety relevant parameter" in object 3100 is accessed. Entering 0xA5 switches the configuration to valid. In the case of an invalid value (not 0 or 0xA5) in the safety_gear_ data_valid or incorrect setting of the checksum, write access is rejected and the SRDOs are not transmitted in OPERATIONAL mode.
Attention:
The flag is automatically reset by writing to the safety position data in safety_gear_configuration object 3100.

safety_gear_data_valid

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
31FE		safety_gear_data_valid	Byte	rw	0 / 0xA5		0xA5

7.1.2 Object 31FF safety_gear_CRC_checksum

This parameter contains the checksum crosswise through the safety CAN parameters in objects 3100 according to the table below. The checksum is written by the master. The checksum is checked on setting the safety_gear_data_valid. If no correspondence with the checksum stored in this object is ascertained, setting the flag is blocked (safety_gear_data_valid remains ZERO). To calculate the CRC checksum you can use the special TWK program.

Link to the program on www.twk.de: Look for the desired device → “Download” and “Software Safety CRC”.
You can also find it on the website at "SUPPORT AND SERVICE" → “DOCUMENTATION” → “Software" and then "SafetyCRC Software”.
Or use the following link:
https://www.twk.de/files/001_CANopen/001_CRC/001_Programm/SafetyCRC_V2.0.9.zip
Description file: CRC 14076. Please ask our technicians for xml files for special versions of encoders or look at www.twk.de.

Attention: The parameter can only be written if the data valid flag of the safety position parameter safety_configuration_valid, object 61FE, is activated. See example for parameterization at the end of this document.

safety_gear_CRC_checksum

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
31FF	0	largest_supported_sub_index	Byte	ro	0x1		0x1
	1	safety_gear_configuration_checksum	WORD	rw	0 ... 0xFFFF		0x88FC *

* Depends on values in Object 3100.

After resetting the gear parameters, the output position is undefined within the code range of the output resolution (object 3100/3). By setting a preset value (object 6100/2) a defined initial position is set.

7 Manufacturer Specific Objects

7.1.3 Safety Notes to slewing ring functionality for safety_gear_configuration

The encoder can in the range of 4096 turns determine an absolute position of the shaft. The working range of the slewing ring function is split in two areas, positive and negative relative to its zero position.

Without losing the correct position value of the slewing ring the allowed number of encoder shaft-turns r_{safety} in a situation when the encoder is not supplied with power can be calculated according to the following formula:

r_{safety}	allowed safety encoder revolutions in each direction without power supply to the encoder
i	gear ratio (i= slew gear / measure gear)
s_{mult}	multiturn encoder steps $s_{multi} = \frac{\text{Object 3300/02}}{\text{Object 3300/01}}$ (S_{multi}=4096 for most applications)

$$r_{safety} = \frac{s_{multi}}{2} - i$$

r_{safety} is rounded down to "full revolutions"

Safety Note:
It has to be ensured in the application, that the shaft is never turned x revolutions, where x > r_{safety} in a state where the encoder is not connected to supply power.
If in a non power state the encoder shaft is turned for x revolutions, where x > r_{safety} the output position is not valid, this is not detected by the encoder upon reconnecting it to supply power.

The smallest possible value for r_{safety} = 1024 encoder shaft turns in the case of i (gear ratio) = 1024. Bigger values for the gear ratio are not possible since the plausibility check of the gear ratio rejects the parameterization and valid flag 31FE cannot be set.
This means the safety area will always be in the range of 2047 ≥ r_{safety} ≥ 1024 revolutions of the encoder shaft depending on the gear ratio.

Example 1:
s_{multi} = 4096 [dez]
i = 3 [dez]
r_{safety} = 2045 [encoder shaft revolutions]

Example 2:
s_{multi} = 4096 [dez]
i = 3.2 [dez]
r_{safety} = 2044 [encoder shaft revolutions]

7 Manufacturer Specific Objects

7.2 Object 3102 gear_configuration

This Object is for setting the gear parameters for non safety PDO transmission.

Only valid for TRN version with slewing ring function (Version "S" in order code number between ST and MT resolution).

Note: Maximum gear ratio i =1024 [dez], Minimum gear ratio i =1, i = slew_gear / measure_gear
Bigger values are not allowed. A plausibilty check ensures that bit 12 in object 6000 cannot be set to "1". Error code 0x 0800 0022 - data not stored to device is transmitted via SDO transfer if the plausibility check fails.

Setting of the Preset value is possible via object 6003.

Adjustment of gear parameters.

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
3102	0	largest_supported_ sub-index	-	ro	0x03	-	-
	1	slew_gear [teeth]	LONG	rw	0x0 ... 0x FFFF FFFF		0x0
	2	measure_gear [teeth]	LONG	rw	0x0 ... 0x FFFF FFFF		0x0
	3	measuring_range	LONG	rw	0x0 ... 0x FFFF FFFF and ≤ [6001] x i (i= slew_gear / measure_gear)		0x0

slew_gear	Number of teeth driving gear (slewing)
measure_gear	Number of teeth of encoder tooth gear
measuring_range	Number of steps output for 1 turn of the slewing ring

Example for 3102 sub 3:
Condition: Object 6001 = 0x4000 (16384[dez])
If i = 3 (for example 30 / 10) the maximum of the measuring range is 49,152 (0xC000) (Object 6001 x i).
If i = 1 → measuring range is ≤ 16,384 (0x4000). Depending on ratio i, 3102 sub 3 has to be matched. This is reasonable for providing every single step of resolution.

Note: Before setting values in Object in 3102 it is mandatory to set the following:

Object	Value [hex]	Value [dez]
6000	0x04 (for CW) or 0x05 (for CCW)	4 (for CW) or 5 (for CCW)
6001	0x4000	16,384
6002	0x400,0000	67,108,864

To activate / deactivate the standard slewing ring function please refer to Object 6000 bit 12 (slewing ring function).

If a gear parameter is changed before the Data Valid flag is reset, the Data Valid is automatically deleted (set "0"). The attempt to set the flag fails without an error message. The flag can only be set after deleting the deleted flag.
After that, the error message appears at startup:
0xFF 0xFF 0x81 0x00 0x20 0x00 0x02 0x02
The error message can only be deleted by a reset.
So the changed values have to be saved in order to activate the new setting.

After modifying any gear parameter, the output position is undefined within the code range of the output resolution (object 3102 sub 3). By setting a preset value (object 6003) a defined initial position is set.

7 Manufacturer Specific Objects

Mandatory steps in order to set Object 3102 gear_configuration

1. Set Objects 6000, 6001 and 6002 as described above
2. Set Object 3102 sub 1, 2 and 3 as desired
3. Set Object 6000 to 0x1004 for CW or 0x1005 for CCW (Activate slewing ring functionality)
4. Set Object 1010 sub 1 to 0x65766173 in order to save_all_parameters

If step 4 was not fulfilled the encoder upon reset will give out the error code: 0x FF FF 81 00 20 00 02 02.
To solve this issue it is mandatory to do a load default (Object 1011 sub1) and a Reset. After that the recommendet steps shall be followed to parameterize the slewing gear functionality.

7.2.1 Safety notes to slewing ring function of standard slewing ring position

The encoder can in the range of 4096 turns determine an absolute position of the shaft. The working range of the slewing ring function is split in two areas, positive and negative relative to its zero position.

The allowed number of encoder shaft-turns r_{safety} in a situation when the encoder is not supplied with power can be calculated according to the following formula:

r_{safety}	allowed safety encoder revolutions in each direction without power supply to the encoder
i	gear ratio (i= slew gear / measure gear)
S_{mult}	multiturn encoder steps $s_{multi} = \frac{\text{Object 6002}}{\text{Object 6001}}$ (S_{multi}=4096 for most applications)

$$r_{safety} = \frac{s_{multi}}{2} - i$$

r_{safety} is rounded down to "full revolutions"

Safety Note:
It has to be ensured in the application, that the shaft is never turned x revolutions, where x > r_{safety} in a state where the encoder is not connected to supply power.
If in a non power state the encoder shaft is turned for x revolutions, where x > r_{safety} the output position is not valid, this is not detected by the encoder upon reconnecting it to supply power.

The smallest possible value for r_{safety} = 1024 encoder shaft turns in the case of i (gear ratio) = 1024. Bigger values for the gear ratio are not possible since the plausibility check of the gear ratio rejects the parameterization and operating byte 12 in Object 6000 cannot be set.
This means the safety area will always be in the range of 2047 ≥ r_{safety} ≥ 1024 revolutions of the encoder shaft depending on the gear ratio.

Example 1:
s_{multi} = 4096 [dez]
i = 3 [dez]
r_{safety} = 2045 [encoder shaft revolutions]

Example 2:
s_{multi} = 4096 [dez]
i = 3.2 [dez]
r_{safety} = 2044 [encoder shaft revolutions]

7 Manufacturer Specific Objects

7.3 Object 3300 safety_encoder_parameter extension of Safety Parameters for 610X

Note:Only valid for Multiturn Encoder (TRN) with CANopen Safety SIL2 (TRN...S4...) and TBSN/TRSN with CANopen Safety SIL3 (TBSN/TRSN...S41...)

This parameter contains an extension for safety parameters not defined in the scope of standard encoder Objects.
These parameters define the capability of a safety encoder with respect to model and performance for safety tasks, which are otherwise not communicated through CANopen safety protocoll.
These parameters cannot be changed and are set ex works (wp).

safety_encoder_parameter

Index	Sub	Description	Length COM	Memory Type	Range/value	Action	Default
3300	0	largest_supported_sub_index	Byte	ro	0x03		0x03
	1	safety_measuring_units_per_ revolution	LONG	wp	depending on model and perfomance e.g.: 4096		
	2	safety_ total_measuring_ range_in_measuring_units	LONG	wp	depending on model and performance e.g.: 16777216		
	3	safety_factory_offset	LONG	wp	0...(Obj. 3300/2)-1		

* Written in factory programming state.

8 Object listing

Name	Index	Sub	Len	Value dez	Value hex	Remark
device_type	1000		4	131478	0x00020196	
error_register	1001		1	0	0x00	
PredefinedErrorField	1003					
NumberofErrors		0	1	0	0xX	
StandardErrorField_1 to StandardErrorField_13		1 to 0B	4			
COB-ID_SYNC	1005		4	128	0x00000080	
manufacturer_device_name	1008		18	-	-	Encoder TRN/TBN/ TBSN/ TRSN Safety
manufacturer_hardware_version	1009		13	-	-	P-0xxx
manufacturer_software_version	100a		15	-	-	Safety Standard
store_parameters	1010					
largest_supported_sub-index		0	1	5	0x05	
save_all_parameters		1	4	1	0x00000001	
SaveCommunicationParameters		2	4	1	0x00000001	
SaveApplicationParameters		3	4	1	0x00000001	
SaveLMTDefinedParameters		4	4	1	0x00000001	
SaveManufacturerDefinedParameters		5	4	1	0x00000001	
restore_default_parameters	1011					
largest_supported_sub-index		0	1	5	0x05	
restore_all_default_parameters		1	4	1	0x00000001	
RestoreCommunicationDefaultParameters		2	4	1	0x00000001	
RestoreApplicationDefaultParameters		3	4	1	0x00000001	
RestoreLMTDefinedDefaultParameters		4	4	1	0x00000001	
RestoreManufacturerDefinedDefaultParameters		5	4	1	0x00000001	
COB-ID_EMCY	1014		4	141	0x0000008D	
EMCY_inhibit_time	1015		2	1000	0x03E8	
producer_heartbeat_time	1017		2	0	0x0	
identity_object	1018					
largest_supported_sub-index		0	1	4	0x04	
vendor-ID		1	4	269	0x0000010D	
product_code		2	4	25376	0x0000xxxx	
revision_number		3	4	65537	0x00020002	
serial_number		4	4	x	0xxxxxxxxx	
error_behaviour	1029					
NrofErrorClasses		0	1	3	0x03	
CommunicationError		1	1	0	0x00	
InternalDeviceError		2	1	0	0x00	
		3	1	1	0x00	
RedundancyError (only for full redundant encoders)		4	1	1	0x01	

8 Object listing

Name	Index	Sub	Len	Value dez	Value hex	Remark
SRDO1_communication_parameter	1301					
largest_supported_sub-index		0	1	6	0x06	
information_direction		1	1	1	0x01	
SRDO_refresh_time		2	2	25	0x0019	
validation_time		3	1	20	0x14	
transmission_type		4	1	254	0xFE	
SRDO_compar_COB_ID_1		5	4	281	0x00000119	0: active 8: inactive
SRDO_compar_COB_ID_2		6	4	282	0x0000011A	0: active 8: inactive
SRDO2_communication_parameter	1302					
largest_supported_sub-index		0	1	6	0x06	
information_direction		1	1	1	0x01	
SRDO_refresh_time		2	2	25	0x0019	
validation_time		3	1	20	0x14	
transmission_type		4	1	254	0xFE	
SRDO_compar_COB_ID_1		5	4	345	0x00000159	0: active 8: inactive
SRDO_compar_COB_ID_2		6	4	346	0x0000015A	0: active 8: inactive
SRDO1_mapping_parameter	1381					
largest_supported_sub-index		0	1	8	0x08	
first_SRDO_mapping_object		1	4	1629487368	0x61200108	
second_SRDO_mapping_object		2	4	1629552904	0x61210108	
third_SRDO_mapping_object		3	4	1629487624	0x61200208	
fourth_SRDO_mapping_object		4	4	1629553160	0x61210208	
fifth_SRDO_mapping_object		5	4	1629487880	0x61200308	
sixth_SRDO_mapping_object		6	4	1629553416	0x61210308	
seventh_SRDO_mapping_object		7	4	1629488136	0x61200408	
eighth_SRDO_mapping_object		8	4	1629553672	0x61210408	
SRDO2_mapping_parameter	1382					
largest_supported_sub-index		0	1	4	0x04	
first_SRDO_mapping_object		1	4	1629749512	0x61240108	
second_SRDO_mapping_object		2	4	1629815048	0x61250108	
third_SRDO_mapping_object		3	4	1629749768	0x61240208	
fourth_SRDO_mapping_object		4	4	1629815304	0x61250208	
Configuration_valid	13fe		1	0	0x00	
safety_configuration_signature	13ff					
Safety_signature_Number_of_entries		0	1	4	0x04	
SRDO1_signature		1	2	..... *	0x.... *	
SRDO2_signature		2	2	..... *	0x.... *	

8 Object listing

Name	Index	Sub	Len	Value dez	Value hex	Remark
first_transmit_PDO_parameter	1800					
largest_supported_sub-index		0	1	5	0x05	
COB-ID_used_by_PDO		1	4	385	0x8000018D	0: active 8: inactive
transmission_type		2	1	253	0xFD	
inhibit_time		3	2	0	0x0	
reserved		-	-	-	-	
event_timer		5	2	0	0x0	
second_transmit_PDO_parameter	1801					
largest_supported_sub-index		0	1	5	0x05	
COB-ID_used_by_PDO		1	4	641	0x8000028D	0: active 8: inactive
transmission_type		2	1	1	0x01	
inhibit_time		3	2	0	0x0	
reserved		-	-	-	-	
event_timer		5	2	0	0x0	
first_transmit_PDO_mapping	1a00					
largest_supported_sub-index		0	1	2	0x02	0x0: mapping disabled
PDO_mapping_for_the_first_object		1	4	1610874912	0x60040020	
PDOMappingEntry_2		2	4	1613758736	0x60300110	
second_transmit_PDO_mapping	1a01					
largest_supported_sub-index		0	1	2	0x02	0x0: mapping disabled
PDO_mapping_for_the_first_object		1	4	1610874912	0x60040020	
PDOMappingEntry_2		2	4	1613758736	0x60300110	
operating_parameters	6000		2	4	0x04	
measuring_units_per_revolution	6001		4	16384*	0x00004000*	
total_measuring_range_in_measuring_units	6002		4	67108864*	0x400 0000*	
preset_value	6003		4	0	0x0	
position_value	6004		4	x	0xX	
raw_position_value	600c		4	x	0xX	
Speed value	6030					
NrOfObjects		0	1	1	0x1	
Speed value channel 1		1	2	x	0xX	
speed_parameter	6031					
NrOfObjects		0	1	4	0x04	
speed_source_selector		1	1	2	0x02	
speed_integration_time		2	2	100	0x0064	
speed_multiplier		3	2	100	0x0064	
speed_divider		4	2	10	0x000A	

* Depends on settings ex works

8 Object listing

Name	Index	Sub	Len	Value dez	Value hex	Remark
safety_position_configuration_parameters	6100					
NrOfObjects		0	1	3	0x03	
safety_code_sequence		1	2	0	0x0	
safety_preset_value		2	4	0	0x0	
safety_preset_value_high_resolution		3	8		0x7FFFFFFFFF FFFFFF	
safety_speed_configuration_parameters	6101					
NrOfObjects		0	1	7	0x07	
safety_code_sequence		1	2	0	0x0	
safety_preset_value		2	4	0	0x0	
safety_preset_value_high_resolution		3	8	0	0x7FFFFFFFFF FFFFFF	
safety_speed_source_selector		4	1	2	0x02	
safety_speed_integration_time		5	2	100	0x0064	
safety_speed_multiplier		6	2	100	0x0064	
safety_speed_divider		7	2	10	0x00A	
safety_position_value	6120					
NrOfObjects		0	1	4	0x04	
safety_position_value_byte1		1	1	x	0xX	
safety_position_value_byte2		2	1	x	0xX	
safety_position_value_byte3		3	1	x	0xX	
safety_position_value_byte4		4	1	x	0xX	
safety_inverted_position_value	6121					
NrOfObjects		0	1	4	0x04	
safety_inverted_position_value_byte1		1	1	x	0xX	
safety_inverted_position_value_byte2		2	1	x	0xX	
safety_inverted_position_value_byte3		3	1	x	0xX	
safety_inverted_position_value_byte4		4	1	x	0xX	
safety_speed_value	6124					
NrOfObjects		0	1	2	0x02	
safety_speed_value_byte1		1	1	x	0xX	
safety_speed_value_byte2		2	1	x	0xX	
safety_inverted_speed_value	6125					
NrOfObjects		0	1	2	0x02	
safety_inverted_speed_value_byte1		1	1	x	0xX	
safety_inverted_speed_value_byte2		2	1	x	0xX	
safety_application_configuration_valid	61fe		1	165	0xA5	
safety_application_configuration_signature	61ff					
NrOfObjects		0	1	2	0x02	
SRDO1_signature		1	2	..... *	0x.... *	
SRDO2_signature		2	2	..... *	0x.... *	
cyclic_timer	6200		2	0	0x0	
operating_status	6500		2			
singleturn_resolution	6501		4	4096	0x1000	
number_of_distinguishable_revolutions	6502		2	4096	0x1000	

* Depends on settings ex works.

8 Object listing

Name	Index	Sub	Len	Value dez	Value hex	Remark
alarms	6503		2	0	0x0	
supported_alarms	6504		2	63488	0xF800	
supported_warnings	6506		2	0	0x0	
profile_and_software_version	6507		4	xxxxxxx	0xXXXXXXX	
operating_time	6508		4	-1	0xFFFFFFFF	
offset_value	6509		4	0	0x0	
module_identification						
largest_supported_sub-index	650a	0	1	1	0x01	
manufacturer_offset_value		1	4	0	0x0	
serial_number	650b		4	x	0xX	
absolute_accuracy	650d		1	10	0xA	
Device_capability	650e		4	35	0x23	
node-ID	2000		1	13	0x0D	
bit_rate	2001		1	2	0x03	
safety_gear_configuration						
largest supported sub index		0	1	3	0x03	
safety_slew_gear	3100	1	4	0 *	0x0 *	
safety_measure_gear		2	4	0 *	0x0 *	
safety_measuring_range		3	4	0 *	0x0 *	
gear_configuration						
largest supported sub index		0	1	3	0x03	
slew_gear	3102	1	4	0 *	0x0 *	
measure_gear		2	4	0 *	0x0 *	
measuring_range		3	4	0 *	0x0 *	
safety_gear_configuration_valid	31FE		1	165	0xA5	
safety_gear_configuration_signature						
largest supported sub index	31FF	0	1	1	0x01	
gear_configuration_signature		1	2	0x88FC *	35068 *	
safety_encoder_parameter	3300					
largest_supported_sub_index		0	1	3	0x03	
safety_measuring_units_per_revolution		1	2	16384*	0x00004000*	
safety_total_measuring_range_in_measuring_units		2	2	67108864*	0x400 0000*	
safety_factory_offset		3	2	..... *	0x..... *	

* Depends on settings ex work.

9 Example parameterization of encoder TBN/TRN and TBSN/TRSN with CANopen Safety

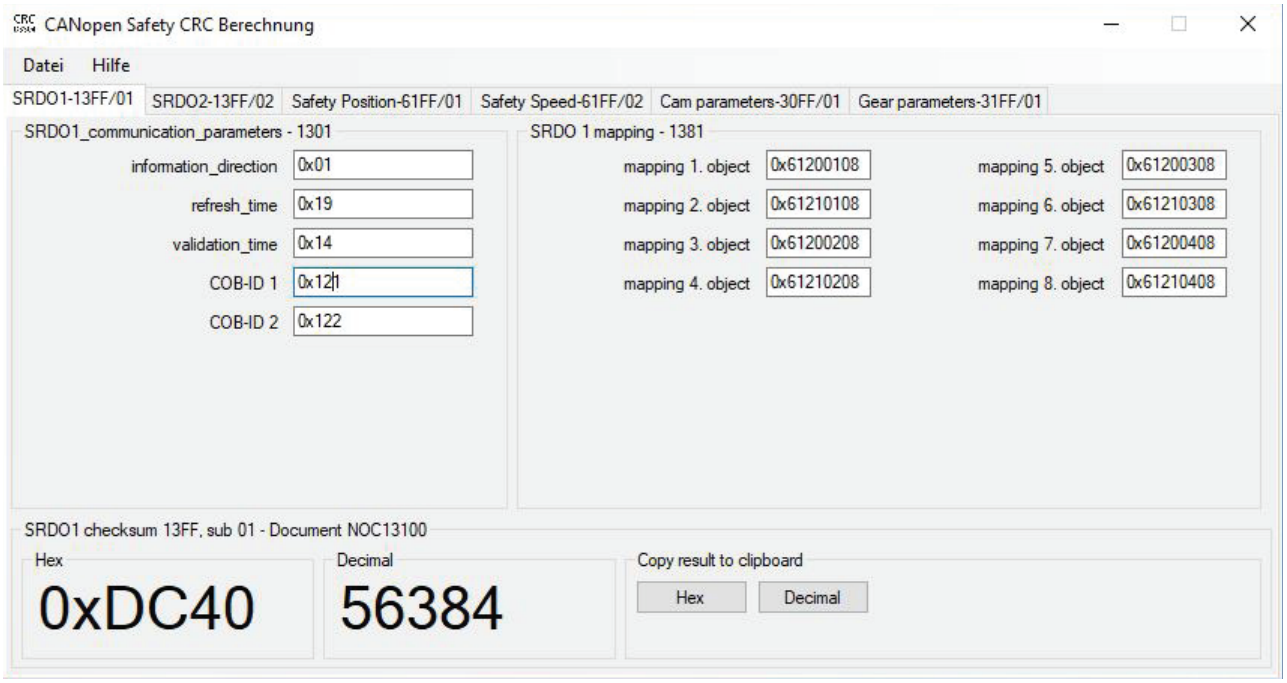
9.1 Parameter setting Object 2000 safety_node ID: 0x1 -> 0x11 (= 17dez), SW Version "R" binary

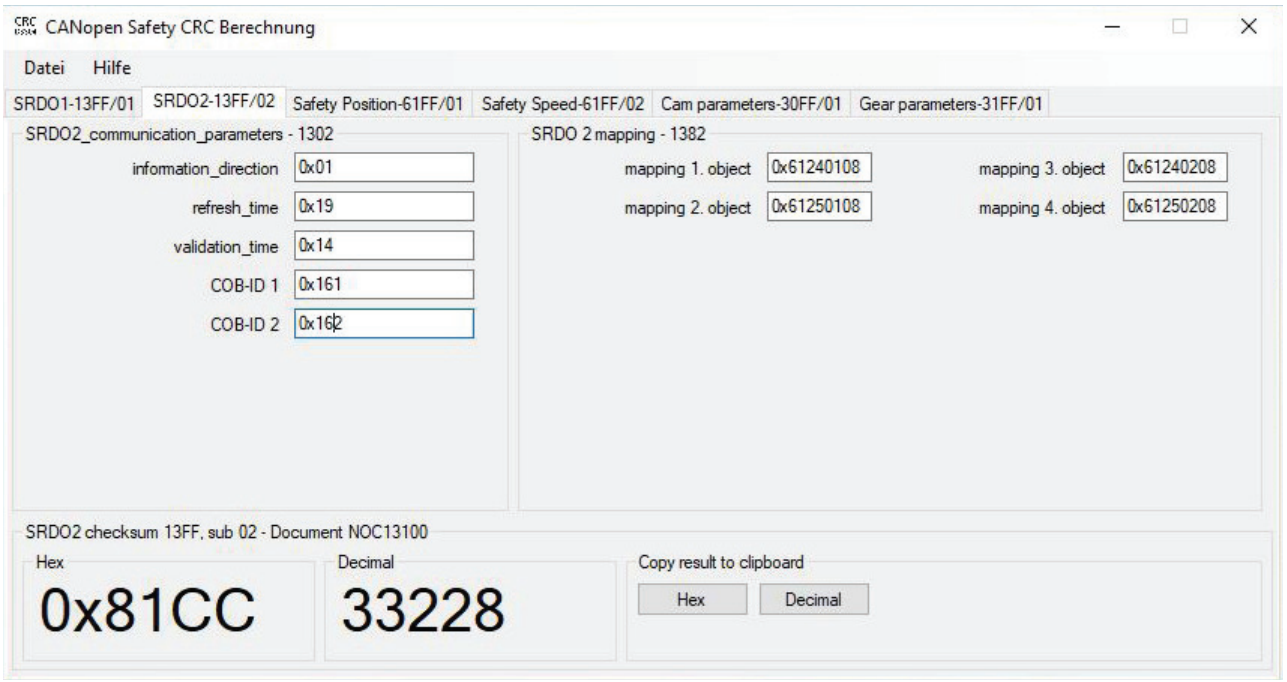
Step	Description	ID	DLC	Byte								Comment
				1	2	3	4	5	6	7	8	
				cs	Index		Sub-Index	Value				
1	Set preoperational	0	2	80	01							
2	Deactivating 13fe/00: configuration_valid	601	8	2f	fe	13	00	00	00	00	00	13fe/00: 00
		581	8	60	fe	13	00	00	00	00	00	
3	Set 2000/00 node ID: 0x11	601	8	2f	00	20	00	11	00	00	00	2000/00: 0x11
		581	8	60	00	20	00	00	00	00	00	
4	Save_LMT_parameters 1010/04	601	8	23	10	10	04	73	61	76	65	1010/04: 0x65766173 (save)
		581	8	60	10	10	04	00	00	00	00	
5	Set 1301/05: COB-ID1	601	8	23	01	13	05	21	01	00	00	1301/05: 121 COB_ID_1 = FF + 2 * Node ID
		581	8	60	01	13	05	00	00	00	00	
6	Set 1301/06: COB-ID2	601	8	23	01	13	06	22	01	00	00	1301/06: 122 COB_ID_2 = 100 + 2 * Node ID
		581	8	60	01	13	06	00	00	00	00	
7	Set 1302/05: COB-ID1	601	8	23	02	13	05	61	01	00	00	1302/05: 161 COB_ID_1 = 13F + 2 * Node ID
		581	8	60	02	13	05	00	00	00	00	
8	Set 1302/06: COB-ID2	601	8	23	02	13	06	62	01	00	00	1302/06: 162 COB_ID_2 = 140 + 2 * Node ID
		581	8	60	02	13	06	00	00	00	00	
9	Calculate CRC checksum (TWK program)											
10	Set 13ff/01: safety_configuration_checksum SRDO1	601	8	2b	ff	13	01	40	DC	00	00	13ff/01: 0xDC40 COB-ID1: 0x121 COB-ID2: 0x122
		581	8	60	ff	13	01	00	00	00	00	
11	Set 13ff/02: safety_configuration_checksum SRDO2	601	8	2b	ff	13	02	CC	81	00	00	13ff/02: 0x81CC COB-ID1: 0x161 COB-ID2: 0x162
		581	8	60	ff	13	02	00	00	00	00	
12	Activating 13fe/00: configuration_valid	601	8	2f	fe	13	00	a5	00	00	00	13fe/00: 0xa5
		581	8	60	fe	13	00	00	00	00	00	
13	Save _all_parameters 1010/01	601	8	23	10	10	01	73	61	76	65	1010/01: 0x65766173 (save) See object 1010 for detailed information concerning sub-indices!
		581	8	60	10	10	01	00	00	00	00	
14	Power off/Power on											
15	Set operational	0	2	11								
16	Position	121	4	12	23	01	00					
17	Position inverted	122	4	ED	DC	FE	FF					
18	Velocity	161	2	23	01							
19	Velocity inverted	162	2	DC	FE							

9 Example parameterization of encoder TBN/TRN and TBSN/TRSN with CANopen Safety

9.2 Screenshots: CRC checksum calculation program

Checksum for 13FF, sub 01 and 02 for Node-ID:.0x11





9 Example parameterization of encoder TBN/TRN and TBSN/TRSN with CANopen Safety

9.3 Node ID Parameterization via LSS Service

In the case of LSS, either all CANopen subscribers are addressed via a global command or an individual subscriber is addressed via its LSS address, which is comprised of the manufacturer name, the product name, the revision number and the serial number.
In the following example, the sensor is addressed via its LSS address (i.e. is switched from LSS-Operation-Mode to LSS-Configuration-Mode), node address 2 is programmed and saved. LSS-Operation-Mode is subsequently reset.
The sensor then reboots and logs on (without voltage off/on) with its boot-up protocol. It is now ready to operate with its new address.
In order ro do this, a switch first has to be made to stop status and the heartbeat timer has to be deactivated, i.e. heartbeat time= 0 (default status).

Attention: During LSS-programming the Heartbeat-Time (Index 1017h) has to be zero (default status).
After programming with LSS service, new checksums have to be calculated accordingly, as mentioned in 11.1 and 11.2 .

Aktion	Id	Rx/Tx	DLC	Databytes								Comment
				00	01	02	03	04	05	06	07	
Stop Node	0	Tx	2	02	00							Stop node for all nodes
LSS-Switch Mode Selective	7E5	Tx	8	40	0D	01	00	00	00	00	00	1. Transmission of the manufacturer name (010D; Object 1018 sub 1)
LSS-Switch Mode Selective	7E5	Tx	8	41	02	63	00	00	00	00	00	2. Transmission of the product number (in this case: 6302; Object: 1018 sub 2)
LSS-Switch Mode Selective	7E5	Tx	8	42	01	00	01	00	00	00	00	3. Transmission of the revision number (in this case: 10001; Object 1018 sub 3)
LSS-Switch Mode Selective	7E5	Tx	8	43	66	BE	02	00	00	00	00	4. Transmission of the serial number (in this case: 179814 [dez]; Object 1018 sub 4)
	7E4	Rx	8	44	00	00	00	00	00	00	00	Success message from the sensor, which is now in LSS Configuration-Mode
LSS-Configure Modul ID	7E5	Tx	8	11	02	00	00	00	00	00	00	Node address 2 programming
	7E4	Rx	8	11	00	00	00	00	00	00	00	Success message from the sensor
LSS-Store Configuration	7E5	Tx	8	17	00	00	00	00	00	00	00	Zero-voltage-protected saving
	7E4	Rx	8	17	00	00	00	00	00	00	00	Success message from the sensor
LSS-Switch Mode Global: Operation Mode	7E5	Tx	8	04	00	00	00	00	00	00	00	Sensor is reset to LSS-Operation-Mode
	702	Rx	1	00								Boot-up node with new node address

9.4 Example parameterization of safety slewing ring function

Step	Description	ID	DLC	Byte								Comment
				1	2	3	4	5	6	7	8	
				cs	Index		Sub-Index	Value				
1	Set preoperational	0	2	80	01							
2	Set 3100/01: safety_slew_gear	601	8	23	00	31	01	83	00	00	00	3100/01: 0x83 Note: valid flag 31FE is automatically deactivated (set 0) by writing to this object
		581	8	60	00	31	01	00	00	00	00	
3	Set 3100/02: safety_measure_gear	601	8	23	00	31	02	0b	00	00	00	3100/02: 0xb
		581	8	60	00	31	02	00	00	00	00	
4	Set 3100/3: safety_measuring_range	601	8	23	00	31	03	a0	8c	00	00	3100/03: 0x8ca0
		581	8	60	00	31	03	00	00	00	00	
5	Calculate CRC checksum (TWK program)											
6	Write checksum 31ff: safety_gear_configuration_checksum	601	8	2b	ff	31	01	c2	0c	00	00	31ff: 0x0cc2
		581	8	60	ff	31	01	00	00	00	00	
7	Activating 31fe/00: safety_gear_data_valid	601	8	2f	fe	31	00	a5	00	00	00	31fe/00: 0xa5
		581	8	60	fe	31	00	00	00	00	00	
8	Activating 61fe/00: safety_application_configuration_valid	601	8	2f	fe	61	00	a5	00	00	00	61fe/00: 0xa5
		581	8	60	fe	61	00	00	00	00	00	
9	Activating 13fe/00: configuration_valid	601	8	2f	fe	13	00	a5	00	00	00	13fe/00: 00
		581	8	60	fe	13	00	00	00	00	00	
10	Save _all_parameters 1010/01	601	8	23	10	10	01	73	61	76	65	1010/01: 0x65766173 (save) See object 1010 for detailed information concerning sub-indices!
		581	8	60	10	10	01	00	00	00	00	
11	Set operational	0	2	11								

10 Error behaviour of the encoder

10.1 General hints to the error behaviour

All errors that occur are displayed and stored in a hierarchical structure. In the error register (object 1001 and 1003) is the error type of the whole system coded. In object 6503 the error type is coded. The error code gives a detailed error description depending on the error type. The data output of the CANopen Safety system is no longer able to work in case a sensor error occurs.

Through the error_behaviour object 1029 the CAN interface can be set up to be operated independently from the status of the sensor.

10.1.1 Fatal errors

Errors which place the functional capability of the controller in doubt - ROM or RAM CRC errors, oscillator drift and watchdog triggering - are not answered with a CAN message. Immediately after detecting the error the controller assumes a secure state (endless loop without actions). This error must be detected and processed by the control system (e.g. hardware device error).

10.1.2 CANopen emergency messages

If the encoder has discovered an error, an emergency message is transmitted unless the node is set to STOP state. The error code is additionally entered in the error register and in object 6503. The behaviour in the event of an error is defined in object 1029 error behaviour. In the event of an error, the sensor switches to the NMT state PRE-OPERATIONAL. If an error disappears (CAN channel error), an EMC message is transmitted again with a deleted error bit. The time interval between emergency messages is determined by object 1015 Inhibit Time EMCY. The absolute encoder's error states remain set until reset or power on occurs.

The structure of the emergency messages can be seen at CANopen emergency messages structure.

11 CANopen emergency messages structure

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
EMC (emergency) error code		Error register	Object 6503 (Alarms)		Specific error code		

EMC (emergency) error codes (Byte 0-1)	
0xFFFF	Customer-specific error; error in the sensor system
0x8120	Passive state error
0x8140	Return from bus off state
0x8110	Overrun error, not all messages can be transmitted by the sensor

Error register (Byte 2): Object 1001		
Bit	M/O	Designation
0	M	generic error
1	O	current
2	O	voltage
3	O	temperature
4	O	communication error (overrun, error state)
5	O	device profile specific
6	O	Reserved (always 0)
7	O	manufacturer-specific error

Object 6503 (Alarms), (Byte 3-4):	
Bit	Error type
0 - 2	Not used
3	Device error
4	Not used
5	CRC parameter error
6	Supply out of range
7	Sensor error

Specific error code (Byte 5-7):
This part is very differentiaded. Examples of typical error messages shown below.

11 CANopen emergency messages structure

11.1 CANopen emergency messages

Emergency messages (sent on ID 80 + node-ID)							
B0	B1	B2	B3	B4	B5	B6	B7
FF	FF	81	00	80	01	02	42
error in the sensor system		generic error, manufacturer-specific error		sensor error		sensor error	error parameter
FF	FF	81	00	20	00	00	00
				CRC parameter error			
FF	FF	81	00	40	00	00	01
				supply voltage out of range			supply voltage too high
FF	FF	81	00	40	00	00	02
				supply voltage out of range			supply voltage too low or short voltage dips
FF	FF	81	00	80	03	03	00
				sensor error	overall	speed overflow	
FF	FF	81	00	80	03	04	00
						synchronisation fault	
20	81	11	00	00	00	00	00
passive error state		generic error, communication error					
40	81	11	00	00	00	00	00
return from bus off state							
Expiry of the inhibit time is followed by the message "correct operation"							
00	00	00	00	00	00	00	00
Expiry of the inhibit time is followed by the message "correct operation"							

The data are output on the bus in Intel format.
A distinction is made between two types of error:

1. Errors in the sensor system (error code 0xFFFF)
   All errors which render proper sensor operation impossible.
2. Communication errors (error code 0x81xx)
   Errors due to the bus system; these are not usually caused by the sensor but indicate a malfunction in the bus system.

All sensor errors are critical errors.

The user of the overall system must assess the errors in the bus system and define the reaction to them.

12 Appendix

12.1 Appendix

12.2 Command specifier for SDO Messages

Command specifier describes the type of SDO Message (see examples for SDOs, CiA 301 Work Draft - CANopen application layer and communication profile)		
Command specifier in hex	Type	Function
22	SDO (rx)	Transmit parameter to encoder Initiate download request (data length max. 4 bytes)
23	SDO (rx)	Transmit parameter to encoder Initiate download request (data length max. 4 bytes)
2B	SDO (rx)	Transmit parameter to encoderInitiate download request (data length = 2 bytes)
2F	SDO (rx)	Transmit parameter to encoder Initiate download request (data length = 1 byte)
60	SDO (tx)	Confirmation of take-over to master Initiate download response
40	SDO (rx)	Request parameter from encoder Initiate upload request
43	SDO (rx) Initiate upload response	Parameter to master with data length = 4 bytes Initiate upload response (Unsigned 32)
4B	SDO (rx) Initiate upload response	Parameter to master with data length = 2 bytes Initiate upload response (Unsigned 16)
4F	SDO (rx) Initiate upload response	Parameter to master with data length = 1 byte Initiate upload response (Unsigned 8)
80	SDO (rx)	Encoder reports error code to master Abort domain transfer

12.3 Configuration valid of the safety node (SRDO Parameter set)

Switch the safety node 1 in the state configuration valid										
ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Function
		cs	SDO		Sub-Index	Data				
701	1	7F								Node 1 in preoperational mode
601	8	2F	FE	13	00	A5	00	00	00	configuration valid
581	8	60	FE	13	00	00	00	00	00	answer sensor

12 Appendix

12.4 NMT State transitions

ID	DLC	Byte 0 Command Specifier	Byte 1 Node ID	Byte 2 - 7	Function
0	2	01	01	00	change to NMT state operational (node-Id = 1)
0	2	02	01	00	change to prepared (stop)
0	2	80	01	00	change to NMT state pre-operational
0	2	81	01	00	reset node
0	2	82	01	00	reset communication

12.5 NMT States

ID	DLC	Byte 0	Function
700 + Node-Id	1	0	Boot up
700 + Node-Id	1	04	Stop
700 + Node-Id	1	05	operational mode
700 + Node-Id	1	7F	pre-operational

12.6 CANopen features of the encoder

- NMT Master: no
- NMT-Slave: yes
- Maximum Boot up: no
- Minimum Boot up: yes
- COB ID Distribution: Default, SDO
- Node ID Distribution: via Index 2000 or LSS (CiA 305)
- No of PDOs: 2 Tx/Rx
- PDO-Modes: sync, async, cyclic, acyclic, safety specific regarding standard
- Variable PDO-Mapping: yes
- Emergency Message: yes
- Heartbeat: yes
- No. of SRDOs: 2 (Position and Velocity - normaly and inverted)
- Device Profile: CiA DS 406 Version 4.1.0 (CANopen and CANopen Safety - SIL2/SIL3)

12 Appendix

12.7 Contact addresses

TWK-ELEKTRONIK GmbH

TWK-ELEKTRONIK GmbH
Bismarckstraße 108
DE-40210 Düsseldorf

Phone: +49 211 961170
e-mail: info@twk.de
www.twk.de

CAN in Automation Organisation

CAN in Automation (CiA)
Kontumazgarten 3
DE-90429 Nuremberg

Phone: +49-911-928819-0
e-mail: headquarters@can-cia.org
www.can-cia.org

13 Changelog

Revision	date	edited by	reviewed by	reason for change
TXN 15469 AE	13.01.2020	U. Stark	A. Albertini	first creation of 15469
TXN 15469 BE	25.03.2020	U. Stark	A. Albertini	addition of TÜV certificate
TXN 15469 CE	28.09.2021	U. Stark	R. Steinebach	general corrections
TXN 15469 DE	25.11.2021	R. Steinebach	U. Stark	addition of slewing ring capability
TXN 15469 EE	10.03.2022	R. Steinebach	A. Albertini	general corrections
TXN/TXSN 15469 FE	11.09.2023	R. Steinebach	A. Albertini	addition of SIL3 encoder TXSN
TXN/TXSN 15469 GE	25.01.2024	R.Steinebach	J. Kühn	Correction of 9.3 and 12.4