

Manual

**Absolute encoder EB360
with SAE J1939 interface**

Contents

1	Version overview	4
2	Safety and operating instructions	5
3	Product Assignment.....	6
4	System Overview	7
4.1	General.....	7
4.2	Supported J1939 Services.....	7
4.3	Function Principle	8
4.3.1	Overview.....	8
4.3.2	Position values / Scaling.....	9
4.3.3	Speed values.....	11
4.3.4	Acceleration value	12
4.3.5	Electronic gear function	13
5	CAN Frame	15
5.1	Identifier.....	15
5.2	Parameter Group Number (PGN)	16
5.2.1	Identifier Examples	16
6	Network Management Service	17
6.1	NAME Field (PGN 60159)	17
6.2	Address Claiming.....	17
7	PGN65450 - cyclic message (PDU2 Format).....	18
7.1	General.....	18
7.2	Frame Format.....	18
7.3	Mapping.....	19
7.3.1	Mappable objects.....	19
7.3.2	Default mapping of absolute encoder	19
7.4	Timing.....	19
7.5	Exceptions of accurate calculation of process data	19
8	PGN61184 - Encoder Parameters and values (PDU1 Format).....	20
8.1	Request frame format for PDU1	20
8.2	CMD Codes	21
8.3	Response frame format for PDU1.....	21
8.4	Response Codes	21
8.5	Error-Codes	22
9	Object descriptions and examples.....	23
9.1	Store parameters (1010h).....	23
9.2	Restore of default values (1011h).....	23
9.3	Reading of the product code (1018h-2)	24
9.4	Reading of the software version (6507h)	24
9.5	Baudrate (2100h).....	25
9.6	ECU default address (2102h)	25
9.7	Electronic gear (2101h)	26
9.8	Measuring units per revolution (6001h)	27
9.9	Total measuring range (6002h).....	27
9.10	Preset value encoder (6003h).....	27
9.11	Group Extension Offset (2103h)	28



9.12	Transmission rate for PGN65450 (2104h)	28
10	Alarms, warnings, errors, encoder diagnostic.....	29
10.1	Surveillance mechanisms	29
10.2	Error injection.....	29
10.3	Encoder diagnostic sources	30
11	Object directory	31
11.1	Communication Profile Area	31
11.2	Manufacturer Specific Profile Area	32
11.3	Standardized Device Profile Area	34
12	Appendix	36
12.1	Electronic gear function (gear legacy mode activated)	36

1 Version overview

This document is subject to changes. In order to have the most current version please download on www.baumer.com

Document index	Date	Firmware version	Revision Number (1018h-03h)	Author	Changes
0001	15.05.2023	From V01-09	0003.0000h	zest	Initial version replaces all draft documents

2 Safety and operating instructions

Intended use

- The encoder is a precision measuring device that is used to record positions and speeds. It provides measuring values as electronic output signals for the subsequently connected device. It must not be used for any other purpose. Unless this product is specially labeled, it may not be used for operation in potentially explosive environments.
- Make sure by appropriate safety measures, that in case of error or failure of the encoder, no danger to persons or damage to the system or operating facilities occurs.

Personnel qualification

- Installation and assembly of this product may be performed only by a person qualified in electronics and precision mechanics.

Maintenance

- The encoder is maintenance-free and must not be opened up nor mechanically or electronically modified. Opening up the encoder can lead to injury.

Disposal

- The encoder contains electronic components. At its disposal, local environmental guidelines must be followed.

Electrical commissioning

- Do not proceed any electrical modifications at the encoder.
- Do not proceed any wiring work while encoder is live.
- Do not remove or plug on connector whilst under power supply.
- Ensure that the entire system is installed in line with EMC/EMI requirements. Operating environment and wiring have an impact on the electromagnetic compatibility of the encoder. Install encoder and supply cables separately or far away from sources with high emitted interference (frequency converters, contactors, etc.).
- When working with consumers with high emitted interference provide separate encoder supply voltage.
- Completely shield encoder housing and connecting cables.
- Connect encoder to protective earth (PE) using shielded cables. The braided shield must be connected to the cable gland or connector. Ideally, aim at dual connection to protective earth (PE), i.e. housing by mechanical assembly and cable shield by the downstream devices.

Supplementary information

- The present manual is intended as a supplement to already existing documentation (e.g. catalogues, data sheets or mounting instructions).

3 Product Assignment

Product	Product code (1018h-2h)	Device name
Bearingless absolute multiturn encoder	0090h	EBxxx MT
Bearingless absolute singleturn encoder	0091h	EBxxx ST

4 System Overview

4.1 General

The encoder is a rotary measuring system with a J1939 interface and supports scaling and presetting. Galvanically isolated encoders are available on request.

4.2 Supported J1939 Services

Following J1939 services are supported:

- Network Management (PGN60928)
 - Address claiming with Name-Field
- PGN65450 + Group Extension (PDU1)
 - Cyclic process data
 - Following Objects can be mapped into this PGN
 - Position (32-Bit)
 - Speed (16-Bit, [rpm])
 - Encoder Diagnostic (16-Bit)
 - Alarm (16-Bit)
 - Warnings (16-Bit)
 - Time Stamp (16-Bit)
- PGN61184 (PDU1)
 - Encoder parameters
 - Direction
 - Scaling (Measuring units per revolution, Total measuring range)
 - Preset
 - PDU2 parameters
 - Object mapping
 - Cyclic time
 - Baudrate
 - Group extension
 - Customer EEPROM (4x 32-Bit)
 - Store/Restore commands
- BAM messages
- Transport protocol

4.3 Function Principle

4.3.1 Overview

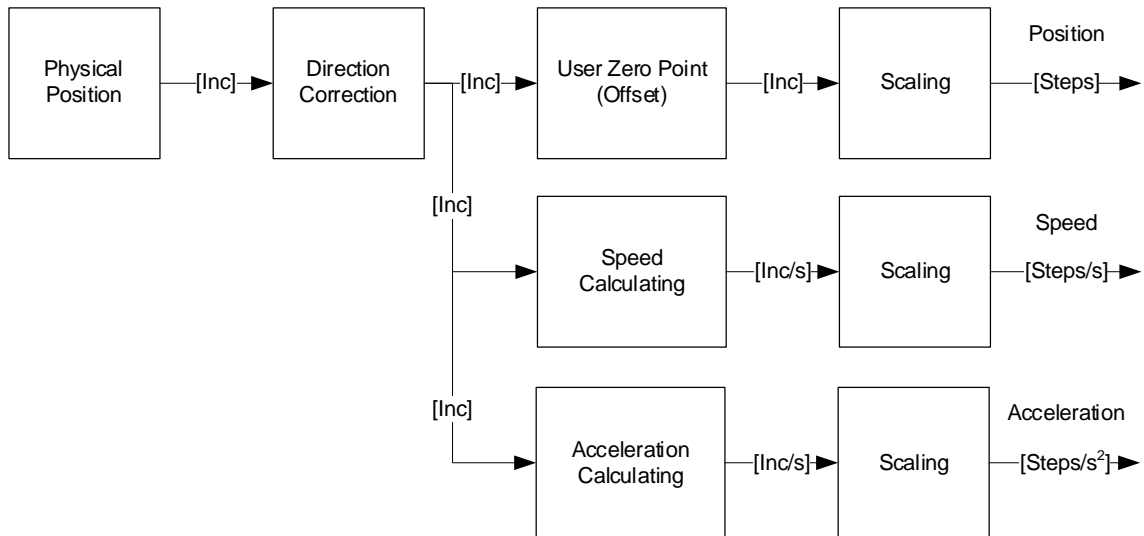


Figure 1: Function principle

4.3.2 Position values / Scaling

Setting of position resolution / scaling is used to adjust the encoder settings to the application.

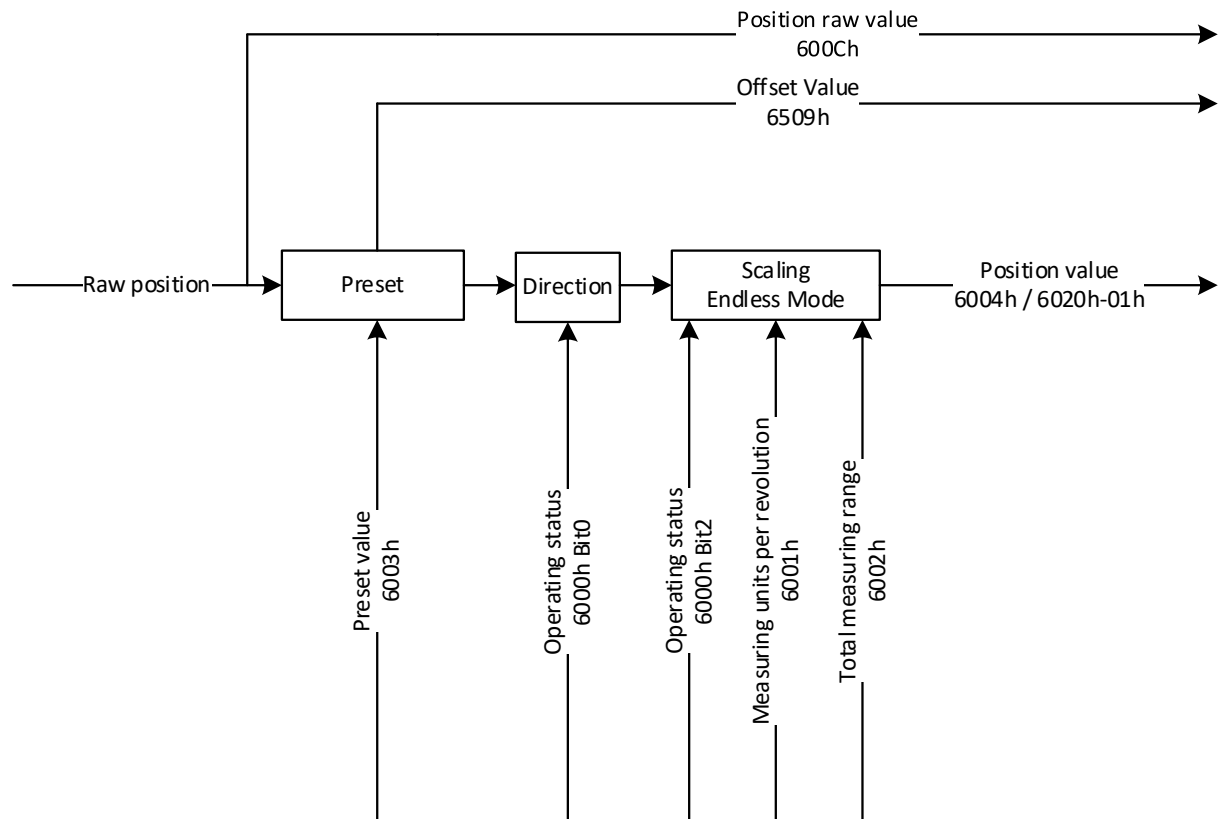


Figure 2: Position value / scaling

Position value (6004h)

The position value (6004h) depends on the position step settings (object 6001h and object 6002h). Scaling is enabled when operating status 6000h Bit2 = 1. The counting direction is affected by object operating status 6000h Bit0.

Measuring units per revolution (6001h)

The measuring units per revolution can be configured with object 6001h. The range is 0 or 2...65'536 steps per revolution.

Total measuring range (6002h)

The total measuring range in measuring units can be configured with object 6002h. (range 0... $2^{32}-1$).

Whereby value 0h means 2^{32} steps or 4'294'967'296dec or 1'0000'0000h steps.

For singleturn encoders the range is limited to 2...value of object 6001h.

Position raw value

The position raw value provides 16 bit singleturn resolution and in case of multiturn encoders 16 bit multiturn information in addition. The raw position value is not affected by scaling, preset or direction of rotation correction. It may be used as source for speed and acceleration calculation.

Limitation

Depending on the setting of measuring units per revolution (6001h), total measuring range (6002h) and the maximum possible revolutions of the encoder ($2^{30} = 1'073'741'824$ turns) the encoder automatically operates in endless mode.

Condition for endless mode operation:

$$2^{30} \frac{\text{measuring units per revolution (6001h)}}{\text{total measuring range (6002h)}} \neq 2^x$$

$$x \in \mathbb{N} (1, 2, 3, \dots)$$

In case of endless mode operation, the maximum encoder revolutions, while the encoder is unpowered or the power on time is shorter than 5 minutes, must be smaller than 2^{29} (536'870'912) turns. If this number of revolutions is exceeded, the encoder has to be referenced (see preset value, 6003h) after every power-up.

Example 1: Parameter combination where the encoder operates without limitation:

Maximum possible revolutions:	1'073'741'824 (30 bit multiturn)
Measuring units per revolution:	3'600
Total measuring range:	29'491'200
Calculation:	$1'073'741'824 \times 3'600 / 29'491'200 = 131'072 (= 2^{17})$

Example 2: Parameter combination where the encoder automatically operates in endless mode:

Maximum possible revolutions:	1'073'741'824 (30 bit multiturn)
Measuring units per revolution:	3'600
Total measuring range:	100'000
Calculation:	$1'073'741'824 \times 3'600 / 100'000 = 38'654'705.664 (\neq 2^x)$

Number of distinguishable revolutions (6502h)

The number of distinguishable revolutions are calculated by dividing the total measuring range by the measuring units per revolution (rounded down to integer) and can be read from Index 6502h:

$$\text{number of distinguishable revolutions (6502h)} = \left\lfloor \frac{\text{total measuring range (6002h)}}{\text{measuring units per revolution (6001h)}} \right\rfloor$$

For singleturn encoders the number of distinguishable revolutions (6502h) is 1.

Preset value (6003h)

To preset the position value (6004h), write the desired value to the object preset value (6003h). The preset becomes effective immediately. To ensure that the calculated offset value (6509h) is also effective after a device restart, a parameter store via object 1010h is required.

4.3.3 Speed values

The two objects 6030h-01h and 2118h provide speed value output. Since speed value and range highly depend on the application, the user should adapt scaling and filtering (6031h-02h) accordingly.

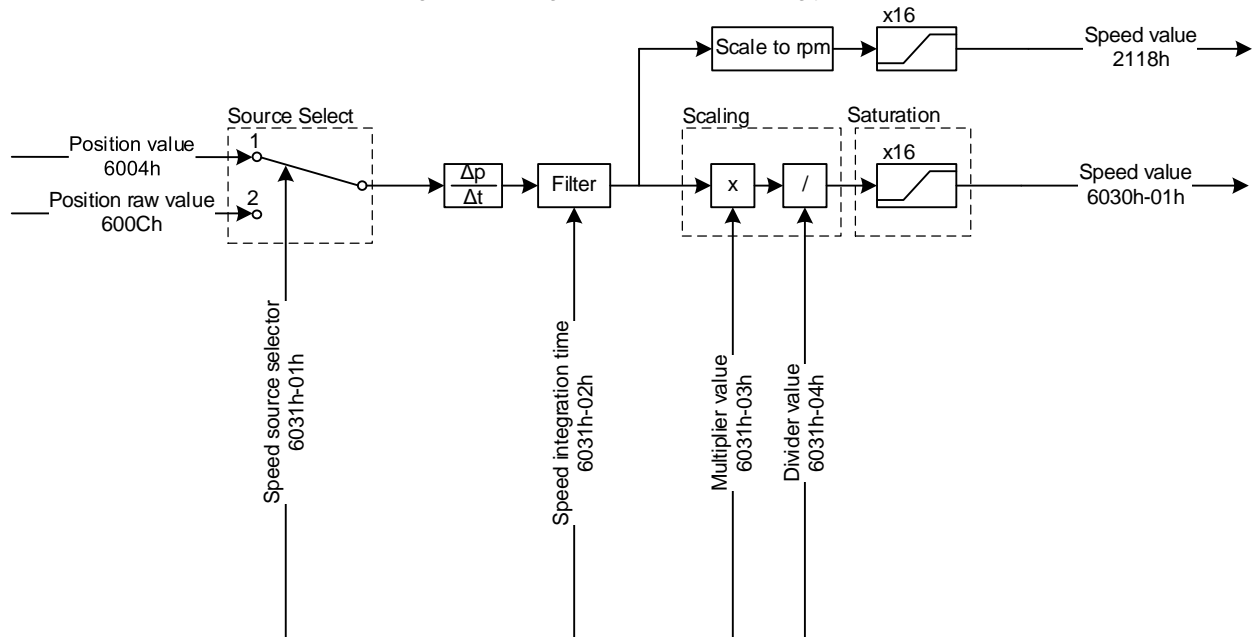


Figure 3: Speed values

NOTE: The encoder speed sign behavior depends on the selected position source (6031h-01h, see Figure 3). For position source position raw value, a possible change of rotation direction via the CW/CCW flag (in 6000h) has no influence on the sign of the speed value.

Speed value (6030h-01h)

The object 6030h-01h provides a 16 bit speed information with unit steps per second [steps/sec].

The range for object 6030h-01h is 8000h...7FFFh (-32768...32767dec or $-2^{15}...2^{15}-1$). If the scaled speed value exceeds the above range, the output is 8000h or 7FFFh (saturated logic).

Scaling can be used to adjust the speed range depending on the selected position source (6031h-01h), if applicable the measuring units per revolution (6001h) and the application speed range.

Speed scaling example:

Selected position source (6031h-01h):	Position value (6004h)
Measuring units per revolution:	16384 (2^{14}) steps
Application speed range in [rpm]:	+/- 3000 rpm (3000/60=50 1/s)
Desired range of speed value:	+/- 30'000 steps/s

Calculation of speed scaling factor:

$$\text{Speed scaling factor} = \frac{30000 \frac{\text{steps}}{\text{s}}}{\frac{3000}{60 \text{ s}} * 16384 \text{ steps}} = 0.0366 = \frac{1}{27.307} = \frac{\text{Multiplier value (6031h - 03h): } 1'000}{\text{Divider value (6031h - 04h): } 27'307}$$

Speed value (2118h)

The object 2118h provides a 16 bit speed value with a fixed unit of revolutions per minute [rpm].

The range for object 2118h is 8000h...7FFFh (-32768...32767dec or $-2^{15}...2^{15}-1$).

If the scaled speed value exceeds the above range, the output is 8000h or 7FFFh (saturated logic).

4.3.4 Acceleration value

In the rotary encoder device profile, the encoder supports acceleration output on object 6040h-01h. As the acceleration value is a highly dynamic value, the user should adapt scaling and filtering (6041h-02h) to his application.

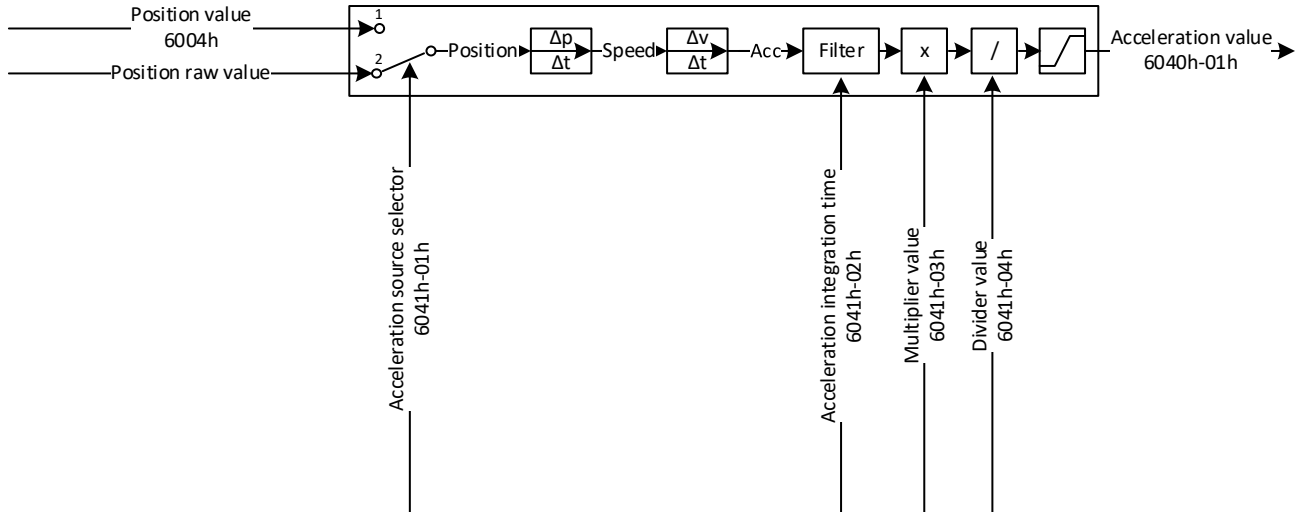


Figure 4: Acceleration value

NOTE: The encoder acceleration sign behavior depends on the selected position source (6031h-01h, see Figure 4). For position source position raw value, a possible change of rotation direction via the CW/CCW flag (in 6000h) has no influence on the sign of the acceleration value.

Acceleration value (6040h-01h)

The object 6040h-01h provides a 16 bit acceleration information with unit steps per second² [steps/sec²].

The range for object 6040h-01h is 8000h...7FFFh (-32768...32767dec or -2¹⁵...2¹⁵-1).

Scaling can be used to adjust the acceleration range depending on the selected position source (6041h-01h), if applicable the measuring units per revolution (6001h) and the application speed range.

Acceleration scaling example:

Selected position source (6031h-01h):	Position raw value
Measuring units per revolution:	65536 (2 ¹⁶) steps
Application acceleration range in [rpm/s]:	+/- 30'000 rpm/s (30000/60=500 1/s ²)
Desired range of acceleration value:	+/- 30'000 steps/s ²

Calculation of acceleration scaling factor:

$$\text{Acceleration scaling factor} = \frac{30'000 \frac{\text{steps}}{\text{s}^2}}{65536 \text{ steps} * \frac{30'000}{60\text{s}^2}} = 915.750 * 10^{-6} = \frac{1}{1092} = \frac{\text{Multiplier value (6041h - 03h): 1}}{\text{Divider value (6041h - 04h): 1'092}}$$

4.3.5 Electronic gear function

The electronic gear function (2001h) divides the encoder position value by the gear factor. Therefore it transforms the encoder position value into the view of the application:

$$\text{Application position} = \frac{\text{Encoder position}}{i}$$

The electronic gear function is also called numerator/denominator scaling or round axis function.

With electronic gear function enabled, the encoder can be mechanically mounted on the primary / input side (drive side) of a gearbox, while it outputs position data as if it was mounted on the secondary / output side (driven side) of the gearbox.

The parameter „total measuring range“ (6002h) always defines the number of requested steps for one revolution on the secondary side of the gear box. The parameter "measuring units per revolution" (6001h) is not taken into account when electronic gear function is enabled.

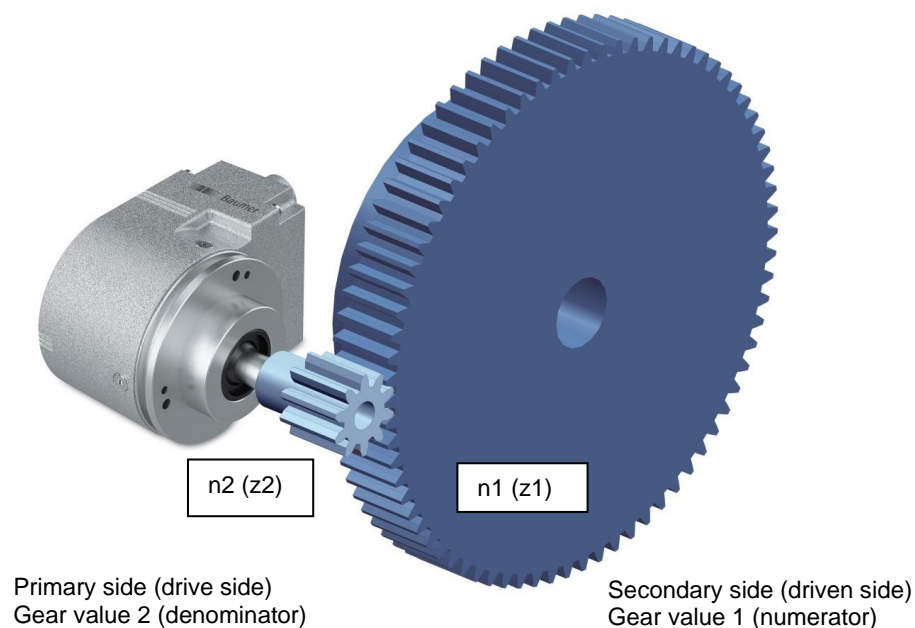


Figure 5: Electronic gear function (2001h)

The gear ratio or gear factor i is defined as:

$$\text{Gear factor } i = \frac{\text{Gear value 1}}{\text{Gear value 2}} = \frac{\text{Numerator}}{\text{Denominator}} = \frac{\text{Speed at drive side (n2)}}{\text{Speed at driven side (n1)}} = \frac{\text{Number of teeth at driven side (z1)}}{\text{Number of teeth at drive side (z2)}}$$

Gear value 1 (numerator, 2001h-02h) and gear value 2 (denominator, 2001h-03h) of the gear factor i directly result from the number of teeth / gear ratio in the application. In the above example, the number of teeth at the driven side is 75. At the drive side, the number of teeth is 10.

Example

The gear factor shall be 75:10 (means $i = 7,5$).

The resolution on the secondary side of the gearbox shall be "1 revolution = 10000 steps".

The numerator is 75 and the denominator is 10.

Only integer values are admissible for numerator and denominator. The total measuring range is 10000.

Electronic gear enable / disable (2001h-01h)

Set object 2001h-01h to value 1h to enable the electronic gear function, while value 0h disables it (further details see table 11.2).

NOTE: A change of object 2001h-01h Electronic gear enable / disable clears internal position offsets (if any). In this case, the current absolute multiturn position reference is lost and the encoder has to be referenced (see preset value, 6003h).

Gear value 1 (2001h-02h)

This object defines the numerator of the gear factor.

The range of this integer value is 1...32'767.

Gear value 2 (2001h-03h)

This object defines the denominator of the gear factor.

The range of this integer value is 1...32'767.

NOTE: After setting and storing the electronic gear function setting, the encoder has to be restarted to activate the electronic gear function.

Limitations

A useful gear factor is greater than 0.125, while a gear ratio smaller than 1 may result in higher signal noise.

Valid combinations of numerator, denominator and total measuring range fulfill following condition:

$$\text{total measuring range (6002h)} * \frac{\text{denominator (2001h - 03h)}}{\text{numerator (2001h - 02h)}} \leq 65536$$

The maximum encoder revolutions when the encoder is unpowered or power on time is shorter than 5 minutes must be smaller than 2^{29} (536'870'912) turns. If this number of revolutions is exceeded, the encoder has to be referenced (see preset value, 6003h) after every power-up.

The electronic gear function is useful for multiturn encoders. In case of singleturn encoders, the position value gets lost after a power cycle.

5 CAN Frame

A standard CAN-Frame with a 29-Bit identifier is being used for the J1939 bus. The data in the PDU fields will be interpreted differently, depending on chosen PDU1 or PDU2 format, which is defined by the Identifier.

		Protocol Data Unit (PDU)							
Description	Identifier	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Bit									

5.1 Identifier

The Identifier is defined as the CAN 29-Bit Identifier and can be configured by the user in two ways:

- Configuring of default ECU address (Object 2102h)
- Configuring of Group Extension (Object 2103h)

		Parameter Group Number (PGN)					
Description	Priority	Extended Data page	Data Page	PDU Format (PF)	Destination Address Group extension	Source address (ECU address)	
Bit	28 ... 26	25	24	23 ... 16	15 ... 8	7 ... 0	

Principally, there are two different message formats used in the J1939 protocol, which are defined by the data range of the PDU format field:

PF Values 0 ... 240 (called as PDU1 Format)

- Peer-to-Peer and broadcast communication
- Field Destination Address always contains the receivers address, or broadcast (25%)
- Used for Encoder parametrization

PF Values 240 ... 255 (called as PDU2 Format)

- Only broadcast communication
- Field Destination Address is used as Group Extension value
- Used for cyclic sending values of the Encoder

Priority:	As lower the value, as higher is the priority on the bus. This value is fixed to the value 6.
Extended Data page:	Only value 0 is supported
Data page:	Only value 0 is supported
PDU Format:	If Values < EFh -> PDU1 format is used If Values > FEh -> PDU2 format is used
PDU1: Destination Address: PDU2: Group Extension:	<ul style="list-style-type: none"> • PDU1: This is either the address of the Encoder, when requesting data or this will be the address of the ECU, which requested Data, when the Encoder responds to a message. • PDU2: Group Extension, can used to create an offset to the cyclic message PGN (65450 + Group Extension)
Source address (ECU address)	Containing always the own address (default-value: 172). This address is claimed with the NM Service (see address claiming)

5.2 Parameter Group Number (PGN)

The Parameter Group Number is an identifier for parameters.

The following PGN's are supported:

- PGN 60928 (Name-Field)
- PGN 61184 (Encoder Parameters)
- PGN 65450 (cyclic process data)

Calculation of Parameter Group Number (PGN):

PDU1 Format:

PGN = "PDU Format Field" * 256 + 0 -> Only every 256 PGN number exist.

PDU2 Format:

PGN = "PDU Format Field" * 256 + "Group Extension value"

5.2.1 Identifier Examples

Identifier for Cyclic PGN 65450 message:

PGN 65450 => in Hex => FFAAh

➔ PDU-Field = FFh (Value > 240 -> PDU2 format with Group Extension is used)

➔ Group Extension = AAh

ECU-Address = 172 = ACh

Description	Priority	Parameter Group Number				Destination Address Group extension	Source address (ECU address)
		Extended Data page	Data Page	PDU Format (PF)			
Value	6	0	0	FFh	AAh	ACh	
Bit	28 ... 26	25	24	23 ... 16	15 ... 8	7 ... 0	
Cob-ID [Hex]	18FFAACh						

Identifier for Acyclic PGN 61184 message, used for read command from another device to the encoder)

PGN 61184 => in Hex => EF00h

➔ PDU-Field = EFh (Value < 240 -> PDU1 format with Destination Address is used)

➔ Dest. Address = ACh (default address of Encoder, could be higher, according to address claim)

Description	Priority	Parameter Group Number				Destination Address Group extension	Source address (ECU address)
		Extended Data page	Data Page	PDU Format (PF)			
Value	6	0	0	EFh	ACh	00h	
Bit	28 ... 26	25	24	23 ... 16	15 ... 8	7 ... 0	
Cob-ID [Hex]	18EFAC00h						

6 Network Management Service

6.1 NAME Field (PGN 60159)

The name field will be sent by the device on every start up as a broadcast message. The name field is used to identify the device in the network, as well as for the address claiming. The name field can't be configured by the user.

Description	1 Bit	3 Bit	4 Bit	7 Bit	1 Bit	8 Bit	5 Bit	3 Bit	11 Bit	21 Bit		
Name field	Arbitrary Address capable	Industry Group	Vehicle Instance	Vehicle System	Reserved	Function	Function Instance	ECU Instance	Manufact. code	Serial-Nr		
Value (example)	1h	5h	0h	0h	0h	8Eh	0h	0h	157h	A7E77h		
PDU Byte	Byte 7			Byte 6		Byte 5	Byte 4		Byte 3	Byte 2	Byte 1	Byte 0
Value (example)	D0h			0h		8Eh	0h		2Ah	EAh	7Eh	77h

NOTE: The NAME field is transmitted LSB first:

			Protocol Data Unit (PDU)							
Description	Identifier	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	PGN60159	8								
Value (example)	18FFAAACH	8	77h	7Eh	EAh	2Ah	0h	8Eh	0h	D0h

6.2 Address Claiming

Every device on the J1939 network needs to observe name field messages on the bus in order to detect, if another device tries to use the same address as the own device.

When it is detected, that another device using the same address, the own name field is also broadcasted on the bus. The device with the lower priority (higher Identifier) needs to increment its own address by one and broadcast the name-field on the bus again.

7 PGN65450 - cyclic message (PDU2 Format)

7.1 General

The device supports a cyclic PDU2 message with PGN65450+GroupExtension, which can be used for cyclic transmitting of the encoder values.

The data/objects which are sent with the PDU2 can be ordered customer specific. The cycle time, can be configured.

7.2 Frame Format

			Protocol Data Unit (PDU)							
Description	Identifier	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Value (example)	18FFAACh	8	18h	91h	B7h	73h	21h	F0h	01h	00h

Identifier:

PGN: 0FFAAh = 65450d						
Identifier	18h = 11000b			FFAACh		
Description	Priority	Ext. DP	Data Page	PDU Format	PDU Specific	Source Address
Bits	3 Bit	1 Bit	1 Bit	8 Bit	8 Bit	8 Bit
Value (example)	6	0	0	FFh = 255d	AAh = 170d	ACh = 172d
Comment				PDU2: Broadcast	Group Extension	

PDU:

Data-Bytes are according to the defined mapping.

Default-mapping is:

Bytes	Description	Hex-Value from example	Interpretation
0 ... 3	Position value (14-Bit ST, 18Bit MT)	73B79118h	Multiturn-Value = 118494d Singleturn-Value = 4376d
4 ... 5	Speed value [rpm]	F021h	- 4063d rpm
6 ... 7	Encoder diagnostic	0001h	Frequency exceeded

7.3 Mapping

The encoder supports a customer specific mapping for PGN65450, which can be specified, when ordering an encoder (on request).

7.3.1 Mappable objects

The following objects are mappable:

Mapping content	Mapping entry	Description
Position value	6004'0020h	Object 6004h sub-index 00h, data length 32 bit
Speed value [steps/s]	6030'0110h	Object 6030h sub-index 01h, data length 16 bit
Alarms	6503'0010h	Object 6503h sub-index 00h, data length 16 bit
Warnings	6505'0010h	Object 6505h sub-index 00h, data length 16 bit
Encoder diagnostic	2117'0010h	Object 2117h sub-index 00h, data length 16 bit
Speed value [rpm]	2118'0010h	Object 2118h sub-index 00h, data length 16 bit
Time stamp [us]	2120'0010h	Object 2120h sub-index 00h, data length 16 bit
Acceleration value	6040'0110h	Object 6040h sub-index 01h, data length 16 bit

7.3.2 Default mapping of absolute encoder

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Position[0]	Position[1]	Position[2]	Position[3]	Speed[0]	Speed[1]	Diagnostic[0]	Diagnostic[1]

Byte 0 – 3 Position value (6004h):

Datatype: 4 Byte, U32

Resolution: According to scaling factors (see encoder parametrization in chapter 4.3.2)

Byte 4 – 5 Speed value [rpm] (2118h):

Datatype: 2 Byte, I16

Resolution: [rpm]

Byte 6 – 7 Encoder diagnostic (2117h):

Datatype: 2 Byte, U16

For more information refer to chapter 10.

7.4 Timing

The minimal cycle time for PGN65450 is 1 ms, although the J1939 suggest to use minimal cycle times of 50 ms. For more information on changing the cycle time for PGN65450, refer to chapter 9.11.

7.5 Exceptions of accurate calculation of process data

The following operations could interrupt the accurate calculation of process data such as position, speed, warnings and alarms:

- Changing the scaling parameters

8 PGN61184 - Encoder Parameters and values (PDU1 Format)

PDU1 Format is being used for reading and writing of the encoder parameters.

The following parameters can be accessed

- Encoder Values
 - Position value (Index 6004h)
 - Speed value (Index 6030h-01h)
 - Speed value [rpm] (Index 2118h)
 - Acceleration value (Index 6040h-01h)
 - Encoder diagnostic (Index 2017h)
- Encoder parameters
 - Direction (Index 6000h)
 - Scaling (Index 6001h / 6002h)
 - Preset (Index 6003h)
 - Electronic gear (Index 2001h)
- PGN65450 cyclic message parameters
 - Transmission rate
- Baudrate (Index 2100h)
- ECU default address (Index 2102h)
- Group extension (Index 2103h)
- Customer EEPROM (4x 32-Bit, Index 2300h)
- Store/Restore commands (Index 1010h / 1011h)

8.1 Request frame format for PDU1

			Protocol Data Unit (PDU)							
Description	Identifier	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Value (example)	18EFAC00h	8	CMD	Index		Sub-Index	Data bytes			

The PF-Value (PDU Format value) shall always be EFh.

The LSB of Index and Data bytes are always transferred first (see examples).

Identifier:

PGN: 0EFAC0h = 61356d						
Identifier	18h = 11000b			EFAC00h		
Description	Priority	Ext. DP	Data Page	PDU Format	PDU Specific	Source Address
Bits	3 Bit	1 Bit	1 Bit	8 Bit	8 Bit	8 Bit
Value (example)	6	0	0	EFh = 239d	ACh = 172d	00h = 0d
Comment				PDU1: Peer-to-peer	Destination Address	

8.2 CMD Codes

CMD-Codes	Description
22h	Writing request, unspecified length
23h	Writing request, 4 bytes
27h	Writing request, 3 bytes
2Bh	Writing request, 2 bytes
2Fh	Writing request, 1 bytes
40h	Read command

8.3 Response frame format for PDU1

			Protocol Data Unit (PDU)							
Description	Identifier	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Value (example)	18EF00ACh	8	CMD	Index		Sub-Index	Data bytes			

Be aware, that the LSB is always transmitted before MSB. For example Index = 2104h, Byte 1 = 04h, Byte 2 = 21h.

Identifier:

PGN: 0EF00h = 61184d						
Identifier	18h = 11000b			EF00ACh		
Description	Priority	Ext. DP	Data Page	PDU Format	PDU Specific	Source Address
Bits	3 Bit	1 Bit	1 Bit	8 Bit	8 Bit	8 Bit
Value (example)	6	0	0	EFh = 239d	00h = 0d	ACh = 172d
Comment				PDU1: Peer-to-peer	Destination Address	

8.4 Response Codes

Response-Code	Description
80h	Error occurred -> Check Error-Code
60h	Write command successfully
4Fh	Reading of 1 byte of data successfully
4Bh	Reading of 2 byte of data successfully
47h	Reading of 3 byte of data successfully
43h	Reading of 4 byte of data successfully

8.5 Error-Codes

Error-Code	Description
0504 0001h	Sent CMD is unknown
0601 0000h	Access to object unsupported
0601 0001h	Read-Access not supported to object
0601 0002h	Write-Access not supported to object
0602 0000h	Object doesn't exist
0609 0011h	Sub-Index doesn't exist
0609 0030h	Value out of Range
0609 0031h	Value too high
0609 0032h	Value too low

9 Object descriptions and examples

The frequently used objects are described in this chapter. More objects can be found in the object dictionary (chapter 10).

9.1 Store parameters (1010h)

With the object 1010h, the parameters can be stored to the non-volatile memory and applied at startup. The response is sent by the encoder after the parameters are saved to the non-volatile memory.

The Sub-Indexes are defined as follow:

Index - SubIndex	Description	Parameters which are saved
1010h-1	Storing of all parameter	Please see the object dictionary for all parameters, which are stored.
1010h-2	Not used for J1939	
1010h-3	Storing of application parameters	Standardized Device Profile Area (please refer to chapter 11.3)
1010h-4	Storing of manufacturer specific parameters	Manufacturer Specific Profile Area (please refer to chapter 11.2)

A magic number-sequence (to avoid accidental saving) is being used to trigger the save-command. The letters “save” need to be used as Byte 4 ... 7 (73h, 61h, 76h, 65h)

Example of saving all parameters:

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Send store all parameters	18EFAC00h	8	22	10	10	01	73	61	76	65
Successful response by encoder (60h)	18EF00ACh	8	60	10	10	01	00	00	00	00

9.2 Restore of default values (1011h)

With the object 1011h, the parameters can be restored to the factory preset values (factory values can be defined when ordering). The values are restored after a reset of the encoder.

The response is sent by the encoder after the command for restoring the parameters is save to the non-volatile memory.

Sub-Indexes are defined as follow:

Index-SubIndex	Description	Parameters which are restored
1011h-1	Restore of all parameter	Please see the object dictionary for all parameters, which are restored.
1011h-2	Not used for J1939	
1011h-3	Restore of application parameters	Standardized Device Profile Area (please refer to chapter 11.3)
1011h-4	Restore of manufacturer specific parameters	Manufacturer Specific Profile Area (please refer to chapter 11.2)

A magic number-sequence (to avoid accidental saving) is being used to trigger the save-command. The letters “load” need to be used as Byte 4 ... 7 (6C, 6Fh, 61h, 64h)

Example of restoring the application parameters:

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Send restore of all parameters	18EFAC00h	8	22	11	10	01	6C	6F	61	64
Successful response by encoder (60h)	18EF00ACh	8	60	11	10	01	00	00	00	00

9.3 Reading of the product code (1018h-2)

Product	Product-Code	Device Name
Bearingless multiturn encoder	0090h	EBxxx MT
Bearingless singleturn encoder	0091h	EBxxx ST

Byte 5: Major Product Code

Byte 4: Minor Product Code

Example of reading the product code

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Reading of the product code	18EFAC00h	8	40	18	10	02	00	00	00	00
Response by encoder: 0090h = Bearingless multiturn encoder	18EF00ACh	8	43	18	10	02	90	00	00	00

9.4 Reading of the software version (6507h)

Byte 7: Major FW-Version

Byte 6: Minor FW-Version

Example of reading the software version

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Reading of the software version	18EFAC00h	8	40	07	65	00	00	00	00	00
Response by encoder: 0109xxxxh => V01.09	18EF00ACh	8	43	07	65	00	xx	xx	09	01

9.5 Baudrate (2100h)

Value	Baudrate
0	10 kBit/s
1	20 kBit/s
2	50 kBit/s
3	100 kBit/s
4	125 kBit/s
5	250 kBit/s
6	500 kBit/s
7	800 kBit/s
8	1000 kBit/s

The baudrate is activated after a reset or power-on (if parameter is saved to non-volatile memory)

Example of changing baudrate to 500 kBit/s

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Changing baudrate to 500 kBit/s (value 6)	18EFAC00h	8	22	00	21	00	06	00	00	00
Successful encoder response (60h)	18EFAACh	8	60	00	21	00	00	00	00	00
Send saving of all parameters	18EFAC00	8	22	10	10	01	73	61	76	65
Successful encoder response (60h)	18EFAACh	8	60	10	10	01	00	00	00	00
Reboot of Encoder and changing of CAN-Adapter baudrate										

9.6 ECU default address (2102h)

The ECU default address (Source address) of the encoder can be set in a range of 0...253 with this object.

Example of setting the ECU default address to 173 (ADh)

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Write ECU default address (ADh)	18EFAC00h	8	22	02	21	00	AD	00	00	00
Successful response by encoder (60h)	18EF00ACh	8	60	02	21	00	00	00	00	00
Send saving of all parameters	18EFAC00h	8	22	10	10	01	73	61	76	65
Successful response by encoder (60h)	18EF00ACh	8	60	10	10	01	00	00	00	00
Reboot encoder to activate the new ECU default address										

9.7 Electronic gear (2101h)

The electronic gear function can be configured by this object. Please note that object functionality 2101h-01 depends on feature control (2110h) settings:

For gear legacy mode "off" (default):

0 = electronic gear disabled

1 = electronic gear enabled

For gear legacy mode "on":

1 = electronic gear disabled

2 = electronic gear enabled

For more information on electronic gear function, refer to chapter 4.3.5. or in case of gear legacy mode "on" to chapter 12.1

Example of enabling electronic gear function (ratio i=7.5)

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Enable electronic gear (enable = 1h)	18EFAC00h	8	22	01	20	01	01	00	00	00
Successful response by encoder (60h)	18EF00ACh	8	60	01	20	01	00	00	00	00
Set gear value 1 (numerator) to 75 (4Bh)	18EFAC00h	8	22	01	20	02	4B	00	00	00
Successful response by encoder (60h)	18EF00ACh	8	60	01	20	02	00	00	00	00
Set gear value 2 (denominator) to 10 (0Ah)	18EFAC00h	8	22	01	20	03	0A	00	00	00
Successful response by encoder (60h)	18EF00ACh	8	60	01	20	03	00	00	00	00
Send saving of all parameters	18EFAC00h	8	22	10	10	01	73	61	76	65
Successful response by encoder (60h)	18EF00ACh	8	60	10	10	01	00	00	00	00

Reboot encoder to activate the electronic gear function with the new settings.

Example of reading electronic gear settings

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Read electronic gear enable/disable	18EFAC00h	8	40	01	20	01	00	00	00	00
Response by encoder with 01h (= enabled)	18EF00ACh	8	4F	01	20	01	01	00	00	00
Read gear value 1 (numerator)	18EFAC00h	8	40	01	20	02	00	00	00	00
Response by encoder with 4Bh (= 75)	18EF00ACh	8	43	01	20	02	4B	00	00	00
Read gear value 2 (denominator)	18EFAC00h	8	40	01	20	03	00	00	00	00
Response by encoder with 0Ah (= 10)	18EF00ACh	8	43	01	20	03	0A	00	00	00

9.8 Measuring units per revolution (6001h)

The measuring units per revolution (Singleturn resolution) of the encoder can be set with this object. For more information on changing the scaling of position output, refer to chapter 4.3.2.

Example of setting the ST Resolution to 3600 (E10h)

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Write Measuring units per rev. (E10h)	18EFAC00h	8	22	01	60	00	10	0E	00	00
Successful response by encoder (60h)	18EF00ACh	8	60	01	60	00	00	00	00	00
<i>Optional send saving of all parameters</i>	<i>18EFAC00h</i>	<i>8</i>	<i>22</i>	<i>10</i>	<i>10</i>	<i>01</i>	<i>73</i>	<i>61</i>	<i>76</i>	<i>65</i>
<i>Optional successful response (60h)</i>	<i>18EF00ACh</i>	<i>8</i>	<i>60</i>	<i>10</i>	<i>10</i>	<i>01</i>	<i>00</i>	<i>00</i>	<i>00</i>	<i>00</i>

Example of reading the ST resolution

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Read Measuring units per revolution	18EFAC00h	8	40	01	60	00	00	00	00	00
Response by encoder with 0E10h (=3600)	18EF00ACh	8	43	01	60	00	10	0E	00	00

9.9 Total measuring range (6002h)

The total measuring range of the encoder can be set with this object. For more information on changing the scaling of position output, refer to chapter 4.3.2.

Example of setting the Total measuring range to 360000 (57E40h)

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Write Measuring units per rev. (57E40h)	18EFAC00h	8	22	02	60	00	40	7E	05	00
Successful response by encoder (60h)	18EF00ACh	8	60	02	60	00	00	00	00	00
<i>Optional send saving of all parameters</i>	<i>18EFAC00h</i>	<i>8</i>	<i>22</i>	<i>10</i>	<i>10</i>	<i>01</i>	<i>73</i>	<i>61</i>	<i>76</i>	<i>65</i>
<i>Optional successful response (60h)</i>	<i>18EF00ACh</i>	<i>8</i>	<i>60</i>	<i>10</i>	<i>10</i>	<i>01</i>	<i>00</i>	<i>00</i>	<i>00</i>	<i>00</i>

Example of reading the Total measuring range

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Read Measuring units per revolution	18EFAC00h	8	40	02	60	00	00	00	00	00
Response by encoder 57E40h (=360000)	18EF00ACh	8	43	02	60	00	40	7E	05	00

9.10 Preset value encoder (6003h)

The position value of the encoder can be preset by writing the desired position to this object. The preset position value is a 32-Bit value. For more information on presetting of position output, refer to chapter 4.3.2.

Example of setting the position to 180000 (2BF20h)

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Write preset Position to Encoder	18EFAC00h	8	22	03	60	00	20	BF	02	00
Successful response by encoder (60h)	18EF00ACh	8	60	03	60	00	00	00	00	00
<i>Optional send saving of all parameters</i>	<i>18EFAC00h</i>	<i>8</i>	<i>22</i>	<i>10</i>	<i>10</i>	<i>01</i>	<i>73</i>	<i>61</i>	<i>76</i>	<i>65</i>
<i>Optional successful response (60h)</i>	<i>18EF00ACh</i>	<i>8</i>	<i>60</i>	<i>10</i>	<i>10</i>	<i>01</i>	<i>00</i>	<i>00</i>	<i>00</i>	<i>00</i>
Cyclic PGN65450 with new Position Value	18FFDDACh	8	20	BF	02	00	00	00	00	00

9.11 Group Extension Offset (2103h)

The group extension offset is used to modify the group extension by adding it to the default group extension (AAh). The range is limited to 0...85 (0...55h) which will result in effective group extension of AA...FFh).

Example of changing the Group Extension to DDh

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Cyclic PGN65450 with Group Ext. AAh	18FFAACh	8	20	BF	02	00	00	00	00	00
Changing Group Extension Offset to 33h	18EFAC0h	8	22	03	21	00	33	00	00	00
Successful response by encoder (60h)	18EF00ACh	8	60	03	21	00	00	00	00	00
<i>Optional Send saving of all parameters</i>	<i>18EFAC00h</i>	<i>8</i>	<i>22</i>	<i>10</i>	<i>10</i>	<i>01</i>	<i>73</i>	<i>61</i>	<i>76</i>	<i>65</i>
<i>Optional Successful response (60h)</i>	<i>18EF00ACh</i>	<i>8</i>	<i>60</i>	<i>10</i>	<i>10</i>	<i>01</i>	<i>00</i>	<i>00</i>	<i>00</i>	<i>00</i>
Cyclic PGN65450 with Group Ext. DDh	18FFDDACh	8	20	BF	02	00	00	00	00	00

Example of Reading Group Extension Offset

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Read Group Extension Offset of encoder	18EFAC0h	8	40	03	21	00	00	00	00	00
Response by encoder with 33h	18EF00ACh	8	4F	03	21	00	33	00	00	00

9.12 Transmission rate for PGN65450 (2104h)

The transmission time is a 16-Bit value, which is in Byte 4 and Byte 5 and has the unit [ms].

Example of changing the transmission rate to 500 ms (1F4h)

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Changing of transmission time to 500ms	18EFAC0h	8	22	04	21	00	F4	01	00	00
Successful response by encoder (60h)	18EF00ACh	8	60	04	21	00	00	00	00	00
<i>Optional Send saving of all parameters</i>	<i>18EFAC00h</i>	<i>8</i>	<i>22</i>	<i>10</i>	<i>10</i>	<i>01</i>	<i>73</i>	<i>61</i>	<i>76</i>	<i>65</i>
<i>Optional Successful response (60h)</i>	<i>18EF00ACh</i>	<i>8</i>	<i>60</i>	<i>10</i>	<i>10</i>	<i>01</i>	<i>00</i>	<i>00</i>	<i>00</i>	<i>00</i>
Cyclic PGN65450 is sent every 500ms										

Example of Reading the transmission rate

Description	Identifier	DLC	Protocol Data Unit (PDU)							
			Byte 0 ... 7 [hex]							
Read transmission time of encoder	18EFAC0h	8	40	04	21	00	00	00	00	00
Response by encoder with 1F4h	18EF00ACh	8	4B	04	21	00	F4	01	00	00

10 Alarms, warnings, errors, encoder diagnostic

Figure 6 shows the surveillance mechanisms. If one of them fails, an alarm or warning will be indicated.

10.1 Surveillance mechanisms

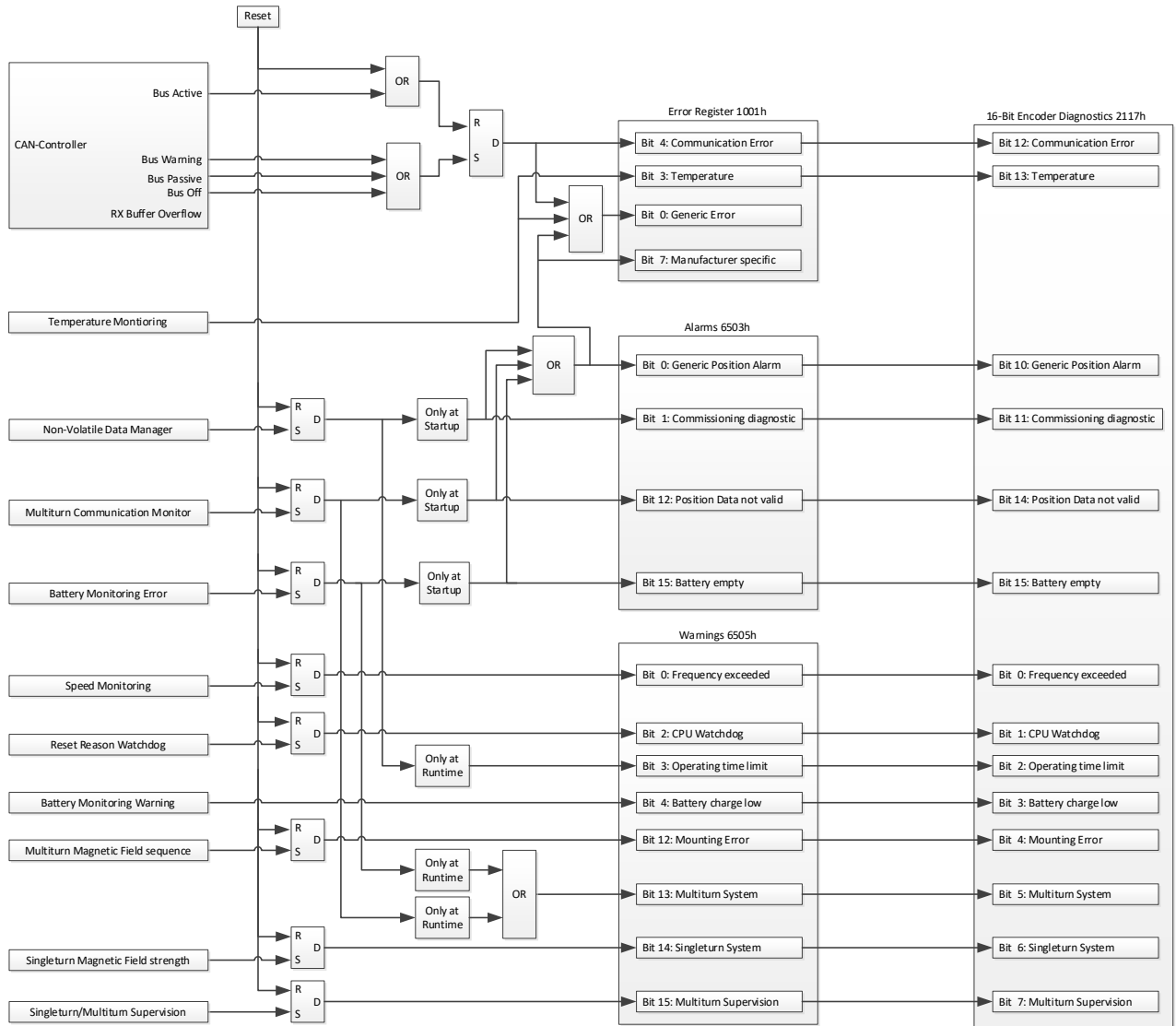


Figure 6: Data flow of error, alarm, warning and diagnostic

10.2 Error injection

The error injection allows testing a system behavior in case of an encoder malfunction. There are two objects that can be used for test purposes to simulate different encoder behaviors.

2116h-01h Diagnostic injection

An error injection code, written on this object simulates a diagnostic source according to chapter 10.3 and activates the corresponding signaling. For error injection codes please refer to chapter 10.3.

2116h-02h Position offset injection

The value written to this object is added to the encoder position value, for test purposes.

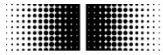
10.3 Encoder diagnostic sources

The following tables provide a summary of all diagnostic sources supported by the encoder.

Source	Diagnostic source
Description	Additional information
Signaling	Object and bit number where the diagnostic source gets visible
Appearance	Describes when the signaling may occur
Reset behavior	Describes when the signaling of a diagnostic source gets reset
Error injection code	Error injection code to simulate a diagnostic source for object 2116h-02h

Source	Description	Appearance	Signaling	Reset behavior	Error injection code
Temperature monitoring	Temperature out of range	at runtime	1001h error, bit 0,3	at runtime	0008'0020h
Non-volatile memory data manager	Problem with non-volatile memory	at startup	1001h error, bit 0 6503h alarm, bit 0,1	at reboot	0001'0024h
		at runtime	6505h warning, bit 3	at reboot	0008'0024h
Reset reason watchdog	SW-reset due to a watchdog timer issue	at runtime	6505h warning, bit 2	at reboot	0008'0025h
CAN controller bus warning	Communication error (bus warning)	at runtime	1001h error, bit 4	at runtime	-
CAN controller RX buffer overflow	CAN receive buffer overflow	at runtime	-	at runtime	-
CAN controller bus passive	CAN in error passive mode	at runtime	1001h error, bit 4	at runtime	-
CAN controller bus-off	CAN in bus-off mode	at runtime	1001h error, bit 4	at runtime	-
Battery monitoring warning	Battery low of multiturn system. Encoder should be replaced	at runtime	6505h warning, bit 4	at runtime	0008'0005h
Battery monitoring error (battery empty)	Battery of the Multiturn system is empty. Absolute position can be incorrect after restart, encoder should to be replaced	at startup	1001h error, bit 0 6503h alarm, bit 0,15	at reboot	0001'0004h
		at runtime	6505h warning, bit 13	at reboot	0008'0004h
Multiturn magnetic field sequence	Incorrect sequence of the magnetic field (i.e. due to weak or disturbed magnetic field)	at startup	6505h warning, bit 12	at reboot	0004'0003h
		at runtime	6505h warning, bit 12	at reboot	0008'0003h
Multiturn communication monitor	Internal communication error	at startup	1001h error, bit 0 6503h alarm, bit 0,12	at reboot	0001'0009h
		at runtime	6505h warning, bit 13	at reboot	0008'0009h
Speed monitor	Maximum rotation speed exceeded	at runtime	6505h warning, bit 0	at reboot	0008'0021h
Singleturn magnetic field strength	Singleturn sensor amplitude out-of-range (i.e. due to weak or disturbed magnetic field)	at runtime	6505h warning, bit 14	at reboot	0008'0022h
Singleturn/multiturn supervision	Multiturn and singleturn sensor out of sync	at runtime	6505h warning, bit 15	at reboot	0008'0023h
-	Manufacturer reserved error codes	-	-	-	-

Remark: In case of any diagnostic source appeared, absolute position may be incorrect.



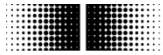
11 Object directory

The following tables provide a summary of all objects supported by the encoder.

Object	Object number in Hex
Name	Object name
Format	U/I = Unsigned/Integer, No. = no of bits, ARR = Array, REC = Record, STR = String
Access	ro = read only, wo = write only, rw = read write, m = mappable
Default	Default value on first init
Save	X = can be stored in the EEPROM
Description	Additional information

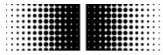
11.1 Communication Profile Area

Object	Sub-Index	Name	Format	Access	Default	Save	Description
1000h		Device type	U32	ro			Device type and functionality 0001'0196h = singleturn encoder 0002'0196h = multiturn encoder
1001h		Error register	U8	ro	0h		Bit0 = Generic error Bit3 = Temperature error Bit4 = Communication error Bit7 = Manufacturer specific error
1008h		Manufacturer device name	STR	ro			Device name as given by manufacturer in ASCII "EBxxx MT" = bearingless absolute multiturn encoder "EBxxx ST" = bearingless absolute singleturn encoder Note: Via PGN61184 only the first four bytes are transmitted
1009h		Manufacturer hardware version	STR	ro			Hardware version in ASCII Note: Via PGN61184 only the first four bytes are transmitted
100Ah		Manufacturer software version	STR	ro			Software version in ASCII Note: Via PGN61184 only the first four bytes are transmitted
1010h		Store parameters	ARR				Control for saving parameters to non-volatile memory
	00h	Highest sub-index supported	U8	ro	4h		
	01h	Store all parameters	U32	rw	1h		="evas" (6576'6173h) to store
	02h	Store communication parameters	U32	rw	1h		="evas" (6576'6173h) to store
	03h	Store application parameters	U32	rw	1h		="evas" (6576'6173h) to store
	04h	Store manufacturer defined parameters	U32	rw	1h		="evas" (6576'6173h) to store
1011h		Restore default parameters	ARR				Control for restoring parameters from non-volatile memory
	00h	Highest sub-index supported	U8	ro	4h		
	01h	Restore all default parameters	U32	rw	1h		="daol" (6461'6F6Ch) to restore
	02h	Restore communication default parameters	U32	rw	1h		="daol" (6461'6F6Ch) to restore
	03h	Restore application default parameters	U32	rw	1h		="daol" (6461'6F6Ch) to restore
	04h	Restore manufacturer defined default parameters	U32	rw	1h		="daol" (6461'6F6Ch) to restore
1018h		Identity object	REC	ro			General identification information of device
	00h	Highest sub-index supported	U8	ro	4h		
	01h	Vendor-ID	U32	ro	5Fh		Vendor-ID
	02h	Product code	U32	ro			Product code: 90h = EBxxx MT bearingless multiturn encoder 91h = EBxxx ST bearingless singleturn encoder
	03h	Revision number	U32	ro			Product revision number
	04h	Serial number	U32	ro			Product serial number

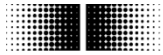


11.2 Manufacturer Specific Profile Area

Object	Sub-Index	Name	Format	Access	Default	Save	Description
2001h		Electronic gear	ARR				Configuration of electronic gear function
	00h	Highest sub-index supported	U8	ro	3h		
	01h	Enable / disable	U8	rw	0h	X	<p>Enable / disable electronic gear 0 = electronic gear disabled 1 = electronic gear enabled</p> <p>NOTE: Object functionality depends on feature control (2110h) settings, for gear legacy mode "on": 1 = electronic gear disabled (default) 2 = electronic gear enabled</p>
	02h	Gear value1	U32	rw	1h	X	<p>Numerator of gear factor, range 1...32767</p> <p>NOTE: Data type depend on feature control (2110h) settings, for gear legacy mode "on": Data type U16</p>
	03h	Gear value2	U32	rw	1h	X	<p>Denominator of gear factor, Range 1...32767</p> <p>NOTE: Data type depend on feature control (2110h) settings, for gear legacy mode "on": Data type U16</p>
2100h		Baud rate	U8	rw	5h	X	<p>Device baud rate 0=10 kbits/s 1=20 kbits/s 2=50 kbits/s 3=100 kbits/s 4=125 kbits/s 5=250 kbits/s 6=500 kbits/s 7=800 kbits/s 8=1000 kbits/s</p> <p>The baud rate is activated after a reset or power-on (if parameter is stored to non-volatile memory)</p>
2102h		ECU default address	U8	rw	ACh	X	ECU Address. Address-Range: 0 ... 253
2103h		Group Extension Offset	U8	rw	0h	X	Group Extension offset, used for PDU2 (proprietary B) Offset range: 0...85
2104h		PGN65450 Transmission Rate	U16	rw	50dec	X	Transmission Rate for PGN65450 [ms]
2110h		Feature control	U16	rw	68h	X	<p>Bit 3: CAN bus-off behavior 0 = Encoder behaves according to object 1029h 1 = Automatic CANopen restart</p> <p>Bit 4: Auto save behavior 0 = Auto save disabled 1 = Auto save enabled, auto save is triggered after preset</p> <p>Bit 5: Speed and acceleration legacy mode 0 = Speed and acceleration legacy mode "on" (data type of object 6030h-01h: I32, behavior like EAM580/360/300) 1 = Speed and acceleration legacy mode "off" (data type of object 6030h-01h: I16), behavior like EAM580S</p> <p>Bit 6: Electronic gear legacy mode 0 = Electronic gear legacy mode "on" (2001h-01h Enable / disable: 2/1, gear values 2001h-02h/03h data type U16, behavior like EAM V01.07.02 and earlier) 1 = Electronic gear legacy mode "off" (2001h-01h Enable / disable: 1/0, gear values 2001h-02h/03h data type U32)</p> <p>NOTE: For the most similar behavior as EAM580/360/300 (not EAM580S) use the value 8h Bit 6 and bit 5 have influence on object data types.</p>
2116h		Error injection	ARR				Simulation of encoder errors for test purpose
	00h	Highest sub-index supported	U8	ro	2h		
	01h	Trigger diagnostic error injection	U32	rw	0h		An error injection code, written on this object simulates a diagnostic source and activates the corresponding signaling according to chapter 10.3.
	02h	Position offset error injection	U32	rw	0h		<p>A temporary position offset could be injected to test a position cross-check in a PLC for example.</p> <p>Allows a position offset error injection. The entered position offset value will be added to the actual position value (6004h).</p>

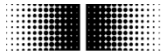


Object	Sub-Index	Name	Format	Access	Default	Save	Description
2117h		Encoder diagnostic	U16	ro, m			Encoder diagnostic bits Bit0 = Frequency exceeded Bit1 = CPU watchdog Bit2 = Operating time limit Bit3 = Battery charge low Bit4 = Mounting error Bit5 = Multiturn system Bit6 = Singleturn system Bit7 = Multiturn supervision Bit10 = Generic position alarm Bit11 = Commissioning diagnostic Bit12 = Communication error Bit13 = Temperature Bit14 = Position data not valid Bit15 = Battery empty
2118h		Speed value	I16	ro, m			Speed value of encoder in [rpm]
2120h		Time stamp	U16	ro, m			Time stamp of output position / value in [us]
2300h		Customer EEPROM	ARR				Customer EEPROM to save any data
	00h	Highest sub-index supported	U8	ro	4h		
	01h	Customer EEPROM [0]	U32	rw	0h	X	
	02h	Customer EEPROM [1]	U32	rw	0h	X	
	03h	Customer EEPROM [2]	U32	rw	0h	X	
	04h	Customer EEPROM [3]	U32	rw	0h	X	
4001h		Speed sampling interval	U16	rw	32h	X	Speed sampling interval for calculation of output speed value in [ms]. Range 1...500 ms. Internally linked to 6031h-02h



11.3 Standardized Device Profile Area

Object	Sub-Index	Name	Format	Access	Default	Save	Description
6000h		Operating parameters	U16	rw	4h	X	Configuration of encoder operating parameters Bit0: Code sequence 0 = Rising values on CW Rotation 1 = Rising values on CCW Rotation Bit2: Scaling function control 0 = Scaling disabled 1 = Scaling enabled If scaling is disabled the position value (6004h) will be calculated with the default values of the measuring units per revolution (6001h) and total measuring range (6002h) instead of the user settings. NOTE: When changing operating parameters, the output position will change and the encoder has to be referenced (see preset value, 6003h)
6001h		Measuring units per revolution	U32	rw	4000h	X	Number of steps per revolution in [steps/rev]. Allowed range is 0 or 2...65'536 steps/rev NOTE: Writing this object will adjust object 6502h. 6502h = 6002h / 6001h (rounded down to integer)
6002h		Total measuring range in measuring units	U32	rw	0h	X	Number of distinguishable steps over total measuring range in [steps]. Allowed range 0 or 1...2 ³² -1 steps (value 0h means 2 ³² steps, i.e. 4'294'967'296dec or 1'0000'0000h steps) For singleturn encoder, the range is 2...value of 6001h Depending on values of 6001h and 6002h, encoder automatically operates in endless mode (please refer to chapter 4.3.2). NOTE: Writing to this object will adjust object 6502h. 6502h = 6002h / 6001h (rounded down to integer)
6003h		Preset value	U32	rw	0h	X	Preset value for position value (object 6004h) in [steps] Internally linked to object 6010h-01h
6004h		Position value	U32	ro, m			Position value in [steps] Internally linked to object 6020h-01h
600Ch		Position raw value	U32	ro	-		Output position value (raw value without scaling or inversion) Multiturn encoders: 16-bit singleturn + 16-bit multiturn Singleturn encoders: 16-bit singleturn
6010h		Preset values for multi-sensor devices	Array			-	Preset value for output position value of multi-sensor device. NOTE: Only channel 1 supported.
	00h	Highest sub-index supported	U08	ro	1h		
	01h	Preset value channel 1	U32	rw	0h		Preset value for channel 1 in steps Internally linked to object 6003h
6020h		Position values for multi-sensor devices	Array			-	Output position value for multi-sensor device NOTE: Only channel 1 is supported.
	00h	Highest sub-index supported	U8	ro	1h		
	01h	Position value channel 1	U32	ro, m	-		Output position value Internally linked to object 6004h
6030h		Speed value	Array				Output speed value
	00h	Highest sub-index supported	U08	ro	1h		
	01h	Speed value for channel 1	I16	ro, m			Output speed value for channel 1 in [steps/s] NOTE: Object functionality and data type depend on feature control (2110h) settings, for speed and acceleration legacy mode "on": Data type I32
6031h		Speed parameters	Array				
	00h	Highest sub-index supported	U8	ro	4h		
	01h	Speed source selector	U8	rw	1h		Selects position object from where speed value is derived. 1 = Object 6004h (position value) is used 2 = Object 600Ch (position raw value) is used
	02h	Speed integration time	U16	rw	50dec		Integration time of speed filter [ms] Range 1...500 ms Internally linked to object 4001h.
	03h	Multiplier value	U16	rw	1h		Output value multiplier
	04h	Divider value	U16	rw	1h		Output value divider



Object	Sub-index	Name	Format	Access	Default	Save	Description
6040h		Acceleration value					Output acceleration value
	00h	Highest sub-index supported	U08	ro	1h		
	01h	Acceleration value channel 1	I16	ro, m			Acceleration value in [steps/s ²] scaled with settings of object 6041h
6041h		Acceleration parameters					
	00h	Highest sub-index supported	U08	ro	4h		
	01h	Acceleration source selector	U08	rw	2h	X	Selects position object from where acceleration value is derived. 1 = Object 6004h (position value) is used 2 = Object 600Ch (position raw value) is used
	02h	Acceleration integration time	U16	rw	200dec	X	Integration time of acceleration filter in [ms] Range 1...500 ms
	03h	Multiplier value	U16	rw	1h	X	Output value multiplier
	04h	Divider value	U16	rw	1092dec	X	Output value divider
6500h		Operating status	U16	ro	4h Bit 2 = 1		Configuration of encoder operating parameters Internally linked to 6000h read-only Bit0: Code sequence 0 = Rising values on CW Rotation 1 = Rising values on CCW Rotation Bit2: Scaling function 0 = Disable scaling 1 = Enable scaling
6501h		Used single turn resolution [steps/rev]	U32	ro	4000h		Internally linked to 6001h read-only
6502h		Number of distinguishable revolutions	U32	ro	40000h		Multiturn: 6002h / 6001h (rounded down to integer) Singleturn: 1
6503h		Alarms	U16	ro, m	0h		The following alarms are evaluated: Bit0 = Generic position alarm Bit1 = Commissioning Diagnostic Bit12 = Position data not valid Bit15 = Battery empty
6504h		Supported alarms	U16	ro	9003h		The following alarms are supported: Bit0 = Generic position alarm Bit1 = Commissioning diagnostics Bit12 = Position data not valid Bit15 = Battery empty
6505h		Warnings	U16	ro, m	0h		The following warnings are evaluated: Bit0 = Frequency exceeded Bit2 = CPU watchdog Bit3 = Operating time limit Bit4 = Battery charge low Bit12 = Mounting error Bit13 = Multiturn system Bit14 = Singleturn system Bit15 = Multiturn supervision
6506h		Supported warnings	U16	ro	F01Dh		The following warnings are supported: Bit0 = Frequency exceeded Bit2 = CPU watchdog Bit3 = Operating time limit Bit4 = Battery charge low Bit12 = Mounting error Bit13 = Multiturn system Bit14 = Singleturn system Bit15 = Multiturn supervision
6507h		Profile & software version	U32	ro			Byte 0...1: Profile version, e. g. 4.1 = 0401h Byte 2: Software minor version Byte 3: Software major version
6508h		Operating time	U32	ro	0h		Total operating time in steps of 6 minutes
6509h		Offset value	I32	ro	0h		Offset value in [steps] (calculated by preset function, shifts position value with calculated value, internally linked to object 650Ch-01h)
650Ah		Module identification	Array				
	00h	Highest sub-index supported	U08	ro	1h		
	01h	Manufacturer offset	I32	ro	0h		
650Bh		Serial number	U32	ro			Internally linked to object 1018h-04h
650Ch		Offset values for multi-sensor devices	Array				
	00h	Highest sub-index supported	U08	ro	1h		
	01h	Offset encoder	I32	ro	0h	X	Offset value channel 1 in [steps] Internally linked to object 6509h

12 Appendix

12.1 Electronic gear function (gear legacy mode activated)

The electronic gear function (2001h) divides the encoder position value by the gear factor. Therefore it transforms the encoder position value into the view of the application:

$$\text{Application position} = \frac{\text{Encoder position}}{i}$$

The electronic gear function is also called numerator/denominator scaling or round axis function.

With electronic gear function enabled, the encoder can be mechanically mounted on the primary / input side (drive side) of a gearbox, while it outputs position data as if it was mounted on the secondary / output side (driven side) of the gearbox.

The parameter "measuring units per revolution" (6001h) defines the number of requested steps for one revolution, the parameter „total measuring range“ (6002h) defines the total measuring range in measuring units on the secondary side of the gear box.

To get a position value 6004h that describes the position within only one revolution of the secondary side of the gear box, set both parameters (6001h & 6002h) on the same value.

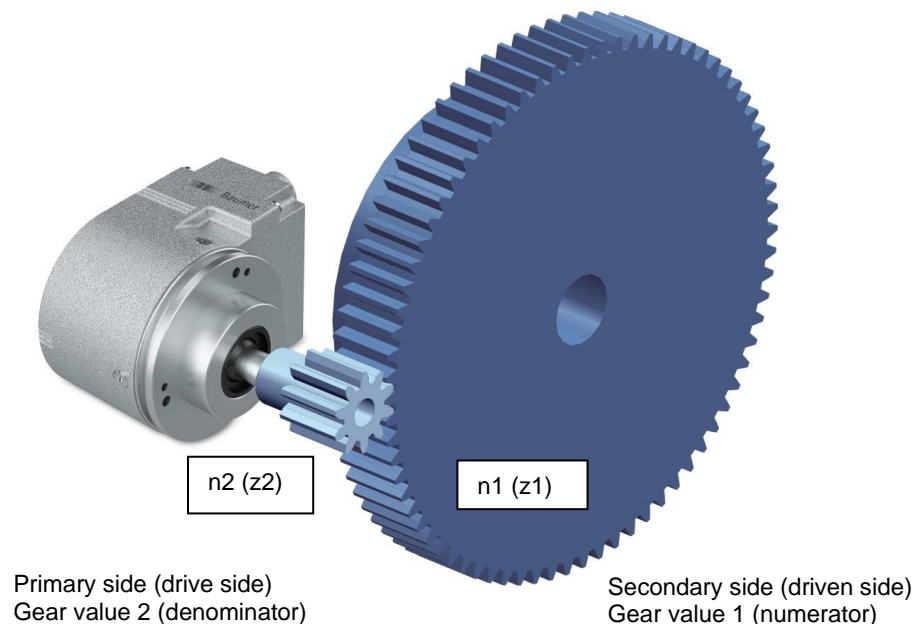


Figure 7: Electronic gear function (2001h)

The gear ratio or gear factor i is defined as:

$$\text{Gear factor } i = \frac{\text{Gear value 1}}{\text{Gear value 2}} = \frac{\text{Numerator}}{\text{Denominator}} = \frac{\text{Speed at drive side (n2)}}{\text{Speed at driven side (n1)}} = \frac{\text{Number of teeth at driven side (z1)}}{\text{Number of teeth at drive side (z2)}}$$

Gear value 1 (numerator, 2001h-02h) and gear value 2 (denominator, 2001h-03h) of the gear factor i directly result from the number of teeth / gear ratio in the application. In the above example, the number of teeth at the driven side is 75. At the drive side, the number of teeth is 10.

Example

The gear factor shall be 75:10 (means $i = 7,5$).

The resolution on the secondary side of the gearbox shall be "1 revolution = 10000 steps".

The numerator is 75 and the denominator is 10.

Only integer values are admissible for numerator and denominator. The total measuring range is 10000.

Electronic gear enable / disable (2001h-01h)

Set object 2001h-01h to value 2h to enable the electronic gear function, while value 1h disables it (further details see table 11.2).

NOTE: A change of object 2001h-01h Electronic gear enable / disable clears internal position offsets (if any). In this case, the current absolute multiturn position reference is lost and the encoder has to be referenced (see preset value, 6003h).

Gear value 1 (2001h-02h)

This object defines the numerator of the gear factor.
The range of this integer value is 1...32'767.

Gear value 2 (2001h-03h)

This object defines the denominator of the gear factor.
The range of this integer value is 1...32'767.

NOTE: After setting and storing the electronic gear function setting, the encoder has to be restarted to activate the electronic gear function.

Limitations

A useful gear factor is greater than 0.125, while a gear ratio smaller than 1 may result in higher signal noise.
Valid combinations of numerator, denominator and total measuring range fulfill following condition:

$$\text{total measuring range (6002h)} * \frac{\text{denominator (2001h - 03h)}}{\text{numerator (2001h - 02h)}} \leq 65536$$

The maximum encoder revolutions when the encoder is unpowered or power on time is shorter than 5 minutes must be smaller than 2^{29} (536'870'912) turns. If this number of revolutions is exceeded, the encoder has to be referenced (see preset value, 6003h) after every power-up.
The electronic gear function is useful for multiturn encoders. In case of singleturn encoders, the position value gets lost after a power cycle.