Baumer

Passion for Sensors

Manual

Cable transducers GCAxx with SAE J1939 interface

Inclination sensor integrated (option)

Baumer Sensors Srl

Via Caduti del lavoro 9
25032 Chiari (Bs)
Phone +39 030 7000916
info.bsit@baumer.com
www.baumer.com

12.23
Subject to modification in technic and design
Errors and omissions excepted

Contents

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	CANopen Revision Number Obj. 1018	Author	Changes
0001	27.04.2023	N.A.	0x060C.0100	gia	Draft version
0002	20.12.2023	From V2.6	0x060C.0100	bson	Updated company name. General revision.
0003	19.04.2024	From V2.8	0x060C.0100	bson	Changed object 0x6503 description and added object 0x2117. Removed object 0x2100 and replaced with object 0x3000. Added sub-indexes 03 and 04 for object 0x2101 (custom presets). Added object 0x2111 related to 16-bit length values. Added objects 0x6E11 and 0x7511 related to device temperatures. Added chapter 4.4 Redundant or full redundant design.

Disclaimer of liability
The present manual was compiled with utmost care, errors and omissions reserved. For this reason Baumer rejects any liability for the information compiled in the present manual. Baumer nor the author will accept any liability for direct or indirect damages resulting from the use of the present information.

At any time we should be pleased receiving your comments and proposals for further improvement of the present manual.

Created by:
Baumer Sensors Srl
Chiari, Italy

2 Safety and operating instructions

Intended purpose of the equipment

- The cable transducer is a precision measuring device that is used to determine linear and angular positions It provides measuring values as electronic output signals for the subsequently connected device. The sensor 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 sensor, 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.
- Consider also the operation manual of the machine manufacturer.

Maintenance

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

Safety remarks

- Prior to commissioning the equipment, check all electrical connections.
- If installation, electrical connection or any other work performed at the sensor or at the equipment is not correctly executed, this can result in a malfunction or failure of the sensor.
- Steps must be taken to exclude any risk of personal injury, damage to the plant or to the operating equipment as a result of sensor failure or malfunction by providing suitable safety precautions.
- Sensor must not be operated outside the specified limited values (see detailed product documentation).

Failure to comply with the safety remarks can result in malfunctions, personal injury or damage to property.

Transport, storage and disposal

- Only ever transport or store sensors in their original packaging.
- Never drop sensors or expose them to major vibrations.
- The sensor contains electronic components. At its disposal, local environmental guidelines must be followed.

Assembly

- Avoid impacts or shocks on the housing.
- Avoid any twist or torsion on the housing.
- Do not open the sensor or make any mechanical changes to it.

The sensor housing or electronic components can be damaged. In this case, safe and reliable operation cannot be guaranteed.

Electrical commissioning

- Do not modify the sensor electrically and remove power supply while connecting it electrically.
- The electrical connection must not be attached or removed under power supply.
- Ensure that the entire plant is installed in line with EMC requirements. The installation environment and wiring affect the electromagnetic compatibility of the sensor.
- Install the sensor and supply cables separately or at a long distance from cables with high interference emissions (frequency converters, contactors etc.)
- Where working with consumers which have high interference emissions, make available a separate power supply for the sensor.
- Unused outputs must not be connected.

Failure to observe these instructions can result in malfunctions, material damage or personal injury.

Supplementary information

- This manual is intended as a supplement to already existing documentation (catalogues, data sheets and assembly instructions).
- The manual must be read without fail before initial commissioning of the equipment.

3 Product Assignment

3.1 Cable transducer

Product
Cable transducer GCAxx with J1939 interface

4 System Overview

4.1 General

The sensor is a linear measuring system with a J1939 interface and supports scaling and presetting.

4.2 Supported J1939 Services

Following J1939 services are supported:

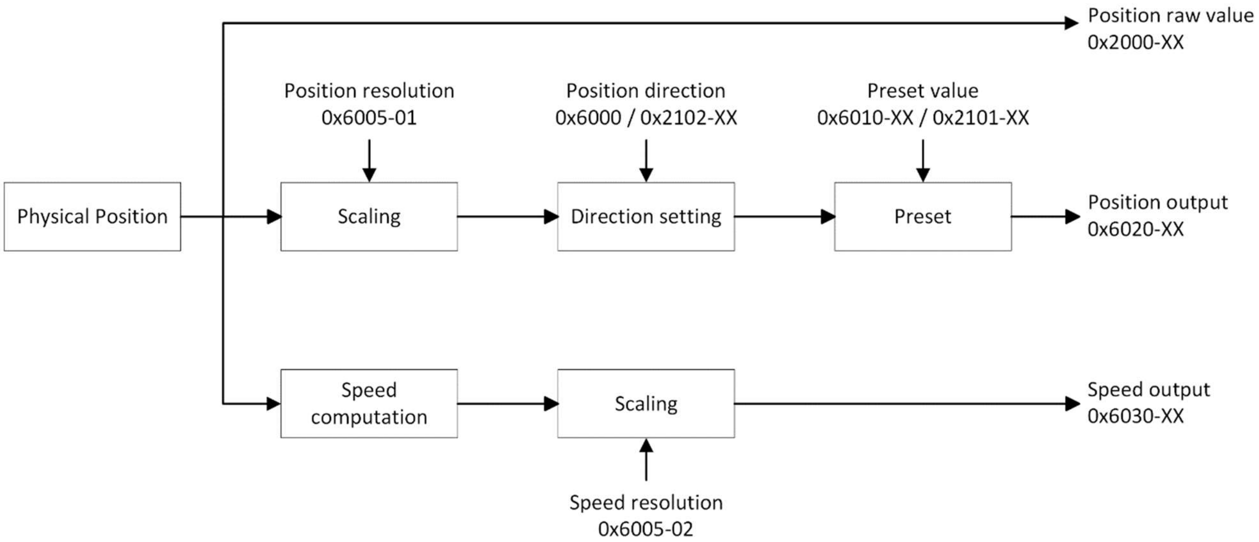
- Network Management (PGN 60928)
  - Address claiming with Name-Field
- PGN 65363 (PGN 65280 + Group Extension)
  - Cyclic Process data. The following objects are mapped into this PGN
    - Position
    - Speed
    - Error
- PGN 65364 (PGN 65280 + Group Extension)
  - Cyclic Process data. The following objects are mapped into this PGN
    - Inclination
- PGN 61184 (Device Configuration)
  - Sensor parameters - Length
    - Direction
    - Resolution
    - Preset
    - Digital length low pass filter
  - Sensor parameters - Angle
    - Resolution
    - Operating parameters
    - Preset
    - Digital angle low pass filter
  - Transmission Time
  - Baudrate
  - Group Extension
  - ECU address
  - Save and Restore commands
- BAM messages
- Transport protocol

4.3 Function Principle

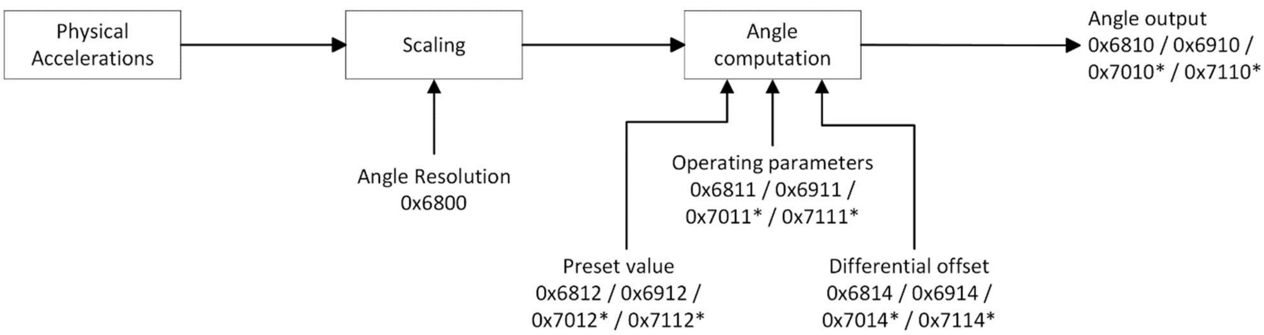
4.3.1 Overview

Figure 1: Function principle overview

Length measurement:



Angle measurement:
(Only for sensors that integrate the inclination sensor)



* For redundant version only (2-sensing)

4.4 Redundant or full redundant design

In a sensor with full redundant design (2-channel), there are two separate nodes with two different ECU addresses connected with the same connector to the network. This means that each node has its own objects that can be read by addressing the correct ECU.

In a sensor with redundant design (2-sensing), there are two primary sensors that are read by a single node with a single ECU address connected to the network. The values of each primary sensor read by the node are written to separate objects.

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 0x3102)
- Configuring of Group Extension (Object 0x3103)

		Parameter Group Number (PGN)				
Description	Priority	Extended Data page	Data Page	PDU Format	PDU Specific	Source address
Bit	28 … 26	25	24	23 … 16	15 … 8	7 … 0

5.1.1 Overview

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

PDU format values 0 … 239 (PDU1 Format)

- Peer-to-Peer and broadcast communication
- Field PDU Specific always contains the receivers address (Destination Address), or broadcast (0xFF)
- Used for parametrization

PDU format values 240 … 255 (PDU2 Format)

- Only broadcast communication
- Field PDU Specific is used as Group Extension value
- Used for cyclic data process

5.1.2 Detailed description

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 0x00 < Values <= 0xEF -> PDU1 format is used If 0xEF < Values <= 0xFF -> PDU2 format is used
PDU Specific	• PDU1: it is the Destination Address (i.e. when requestion data, it is the address of the sensor, when answering a data request, it is the ECU address) • PDU2: it is the Group Extension. Default group extension is 0x53 in case of linear position data or 0x54 in case of inclination data (only for sensors that integrate inclination sensor). Group Extension, can be used to create an offset to the cyclic message: ○ PGN 65280 (0xFF00) + Group Extension (0x53) = PGN 65363 (0xFF53) ○ PGN 65280 (0xFF00) + Group Extension (0x54) = PGN 65364 (0xFF54)

Source address	It is the source address. In case of ECU address, the default value is 0x04*. This address is claimed with the NM Service (see address claiming)

* In case of full redundant product (two separate ECU addresses), the default value of the second ECU address is 0x05.

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 (Sensor Parameters)
- PGN 65363 (Cyclic process data)
- PGN 65364 (Cyclic process data)

Calculation of Parameter Group Number (PGN):

PDU1 Format:
PGN = "PDU Format Field" * 256 + 0

PDU2 Format:
PGN = "PDU Format Field" * 256 + "Group Extension value"

5.2.1 Identifier Examples

Identifier for Cyclic PGN 65363 message:

PGN 65363 => in Hex => 0xFF53

➔ PDU Format = 0xFF (Value ≥ 240 -> PDU2 format with Group Extension is used)

➔ PDU Specific = 0x53 (PDU2 -> Group Extension)

Source address = 0x04

		Parameter Group Number				
Description	Priority	Extended Data page	Data Page	PDU Format	PDU Specific	Source address
Value	6	0	0	0xFF	0x53	0x04
Bit	28 … 26	25	24	23 … 16	15 … 8	7 … 0
Cob-ID	0x18FF5304					

Identifier for Cyclic PGN 65364 message:

PGN 65363 => in Hex => 0xFF54

➔ PDU Format = 0xFF (Value ≥ 240 -> PDU2 format with Group Extension is used)

➔ PDU Specific = 0x54 (PDU2 -> Group Extension)

Source address = 0x04

		Parameter Group Number				
Description	Priority	Extended Data page	Data Page	PDU Format	PDU Specific	Source address
Value	6	0	0	0xFF	0x54	0x04
Bit	28 … 26	25	24	23 … 16	15 … 8	7 … 0
Cob-ID	0x18FF5404					

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

PGN 61184 => in Hex => 0xEF00

➔ PDU Format = 0xEF (Value < 240 -> PDU1 format with Destination Address is used)

➔ PDU Specific = 0x04 (default address of ECU, could be higher, according to address claim)

Source address = 0x00

		Parameter Group Number				
Description	Priority	Extended Data page	Data Page	PDU Format	PDU Specific	Source address
Value	6	0	0	0xEF	0x04	0x00
Bit	28 … 26	25	24	23 … 16	15 … 8	7 … 0
Cob-ID	0x18EF0400					

6 Network Management Service

6.1 NAME Field (PGN 60928)

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.

			Bits of Protocol Data Unit (PDU)									
Description	Identifier	DLC	21 Bit	11 Bit	3 Bit	5 Bit	8 Bit	1 Bit	7 Bit	4 Bit	3 Bit	1 Bit
	PGN 60928	8	Serial Nr.	Manufact. code	ECU Instance	Function Instance	Function	Reserved	Vehicle System	Vehicle Instance	Industry Group	Arbitrary Address capable
GCAxx	0x18EEFF04	8	yyyyy	0x157	0x0	0x0	0xFF	0x0	0x0	0x0	0x3	0x1

Where:
yyyyy: depending by the SN of the product

6.2 Address Claiming

At start up, every device on the J1939 network needs to observe can messages identifiers (CAN-ID) on the bus in order to detect, if another device is trying to use the *same source address*.
When it is detected, that another device is using the same source address, the device with the lower priority (higher device name) increments its own source address by one and broadcast the name-field on the bus again.

7 PGN 65363 - cyclic message (PDU2 Format)

7.1 General

The device supports a cyclic PDU2 message with PGN 65363 (PGN 65280 + Group Extension), which can be used for cyclic transmitting of the sensor values.
The data/objects which are sent by means of the PDU2 can be ordered customer specific. However, the cycle time of the message 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)	0x18FF5304	8	0x10	0x0E	0x00	0x00	0x1E	0x00	0x00	0x00

Interpretation of the values according the example values:

Identifier:
Priority: 6
PGN: 0xFF53 = 65363
Source Address: 0x04

Data-Bytes are according to the defined mapping.
Default-mapping is:

PDU:

Bytes	**Description**	**Example**	**Interpretation**
0…3	Position Value	0xE10	Position Value = 3600
4…5	Speed Value	0x1E	Speed Value = 30
6…7	Error flags	0x00	No error

In case of redundant sensor (2-sensing) the frame format is:

			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)	0x18FF5304	8	0x10	0x0E	0x10	0x0E	0x1E	0x00	0x40	0x00

Interpretation of the values according the example values:

Identifier:
Priority: 6
PGN: 0xFF53 = 65363
Source Address: 0x04

Data-Bytes are according to the defined mapping.
Default-mapping is:

PDU:

Bytes	**Description**	**Example**	**Interpretation**
0…1	Position Value CH1	0xE10	Position Value = 3600

2…3	Position Value CH2	0xE10	Position Value = 3600
4…5	Speed Value CH1	0x1E	Speed Value = 30
6…7	Error flags CH1	0x40	Position Error

7.3 Mapping

The sensor supports a customer specific mapping for PGN 65363, which can be specified, when ordering a sensor.

7.3.1 Default mapping

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]	Error Flag[0]	Error Flag[1]

Byte 0…3 Absolute position

Datatype: 4 Byte, S32

Resolution: According to scaling factors (see parametrization)

Byte 4…5 Speed value

Datatype: 2 Byte, S16

Resolution: According to scaling factors (see parametrization)

Byte 6…7 Error flags

Datatype: 2 Byte, U16

See Object description in chapter 11.16 0x6503 Sensor Diagnostics

In case of redundant sensor (2-sensing) the default mapping is:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Position[0] CH1	Position[1] CH1	Position[0] CH2	Position[1] CH2	Speed[0] CH1	Speed[1] CH1	Error Flag[0]	Error Flag[1]

Byte 0…1 Absolute position CH1

Datatype: 2 Byte, S16

Resolution: According to scaling factors (see parametrization)

Byte 2…3 Absolute position CH2

Datatype: 2 Byte, S16

Resolution: According to scaling factors (see parametrization)

Byte 4…5 Speed value CH1

Datatype: 2 Byte, S16

Resolution: According to scaling factors (see parametrization)

Byte 6…7 Error flags CH1

Datatype: 2 Byte, U16

See Object description in chapter 11.16 0x6503 Sensor Diagnostics

7.3.2 Mappable objects

The mappable objects are listed in chapter 12 Object directory and identified with the “m” (mappable) symbol in the Access column.
Follow the procedure defined in chapter 9 Mapping procedure to change the mapping entries.

7.4 Timing

The minimal cycle time for PGN 65363 is 10ms. The default value is 100ms.

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:

- Non-volatile memory operations (save and load parameters)
- Changing the scaling parameters

8 PGN 65364 - cyclic message (PDU2 Format)

8.1 General

The device supports a cyclic PDU2 message with PGN 65364 (PGN 65280 + Group Extension), which can be used for cyclic transmitting of the sensor values.
The data/objects which are sent by means of the PDU2 can be ordered customer specific. However, the cycle time of the message can be configured.

8.2 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)	0x18FF5404	8	0x84	0x03	0x00	0x00	0x00	0x00	0x00	0x00

Interpretation of the values according the example values:

Identifier:
Priority: 6
PGN: 0xFF54 = 65364
Source Address: 0x04

Data-Bytes are according to the defined mapping.
Default-mapping is:

PDU:

Bytes	Description	Example	Interpretation
0…3	Inclination Value	0x384	Inclination Value = 900
4…7	Reserved	-	-

In case of redundant sensor (2-sensing) the frame format is:

			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)	0x18FF5404	8	0x84	0x03	0x84	0x03	0x00	0x00	0x00	0x00

Interpretation of the values according the example values:

Identifier:
Priority: 6
PGN: 0xFF54 = 65364
Source Address: 0x04

Data-Bytes are according to the defined mapping.
Default-mapping is:

PDU:

Bytes	Description	Example	Interpretation
0…1	Inclination Value CH1	0x384	Inclination Value = 900
2…3	Inclination Value CH2	0x384	Inclination Value = 900

4…7	Reserved	-	-

8.3 Mapping

The sensor supports a customer specific mapping for PGN 65364, which can be specified, when ordering a sensor.

8.3.1 Default mapping

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Inclination[0]	Inclination[1]	Inclination[2]	Inclination[3]	Reserved	Reserved	Reserved	Reserved

Byte 0…3 Inclination angle

Datatype: 4 Byte, S32

Resolution: According to scaling factors (see parametrization)

Byte 4…7 Reserved

In case of redundant sensor (2-sensing) the default mapping is:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Inclination[0] CH1	Inclination[1] CH1	Inclination[0] CH2	Inclination[1] CH2	Reserved	Reserved	Reserved	Reserved

Byte 0…1 Inclination angle CH1

Datatype: 2 Byte, S16

Resolution: According to scaling factors (see parametrization)

Byte 2…3 Inclination angle CH2

Datatype: 2 Byte, S16

Resolution: According to scaling factors (see parametrization)

Byte 4…7 Reserved

8.3.2 Mappable objects

The mappable objects are listed in chapter 12 Object directory and identified with the “m” (mappable) symbol in the Access column.
Follow the procedure defined in chapter 9 Mapping procedure to change the mapping entries.

8.4 Timing

The minimal cycle time for PGN 65364 is 10ms. The default value is 100ms.

8.5 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:

- Non-volatile memory operations (save and load parameters)
- Changing the scaling parameters

9 Mapping procedure

Mapping entries can only be changed using the defined mapping procedure:

1. Disable mapping by switching Bit 31 in the object 0x180X-01

Example of disabling mapping for the ECU address 0x04:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Disable mapping (Object 0x180X-01 bit 31 =1)	0x18EF**04**00	8	23	0X	18	01	84*	n*	00	80
Successful response (0x60)	0x18EF00**04**	8	60	0X	18	01	00	00	00	00

* 0xn84: COB-ID for PDOX

Where:
X is: 0 for PGN 65363, 1 for PGN 65364
n is: 1 for PGN 65363, 2 for PGN 65364

2. Set mapping invalid by writing 0x00 to sub-index 0x00 of the related mapping entries.

Example of setting mapping invalid for the ECU address 0x04:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Set mapping invalid (Object 0x1A0X-00)	0x18EF**04**00	8	2F	0X	1A	00	00	00	00	00
Successful response (0x60)	0x18EF00**04**	8	60	0X	1A	00	00	00	00	00

Where:
X is: 0 for PGN 65363, 1 for PGN 65364

3. Adjust the desired mapping.

Example of adjusting mapping for the ECU address 0x04:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Adjust the desired mapping (Object 0x1A0X-00)	0x18EF**04**00	8	23	0X	1A	OBJ_NBR	SIZE	SUB	OBJ_{LSB}	OBJ_{MSB}
Successful response (0x60)	0x18EF00**04**	8	60	0X	1A	OBJ_NBR	00	00	00	00

Where:
X is: 0 for PGN 65363, 1 for PGN 65364
OBJ_NBR increase starting from 1 to maximum 8 every object associated mapped.
SIZE is:

- 0x8 for 1 byte object
- 0x10 for 2 byte object

- 0x20 for 4 byte object

4. Set sub-index 0x00 of the related mapping index to number of mapped objects.

Example of setting the number of mapped object for the ECU address 0x04:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Set number of mapped objects (Object 0x1A0X-00)	0x18EF**04**00	8	2F	0X	1A	00	TOT_ NBR	00	00	00
Successful response (0x60)	0x18EF00**04**	8	60	0X	1A	00	00	00	00	00

Where:
X is: 0 for PGN 65363, 1 for PGN 65364
TOT_NBR is the Total number of mapped object.

5. Enable mapping by means of Bit 31 in the object 0x180X-01

Example of enabling mapping for the ECU address 0x04:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Enable mapping (Object 0x180X-01 bit 31 =0)	0x18EF**04**00	8	23	0X	18	01	84*	n*	00	40
Successful response (0x60)	0x18EF00**04**	8	60	0X	18	01	00	00	00	00

* 0xn84: COB-ID for PDOX

Where:
X is: 0 for PGN 65363, 1 for PGN 65364
n is: 1 for PGN 65363, 2 for PGN 65364

10 PGN 61184 - Sensor Parameters and values (PDU1 Format)

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

The following parameters can be accessed

- Sensor Values
  - Position
  - Speed
  - Inclination
  - Error
- Sensor parameters - Length
  - Direction
  - Resolution
  - Preset
  - Digital low pass filter configuration
- Sensor parameters - Angle
  - Resolution
  - Operating parameters
  - Preset
  - Digital low pass filter configuration
- Transmission Time
- Baudrate
- Group Extension
- ECU address
- Save and Restore commands

10.1 Request frame format for PDU1

		Protocol Data Unit (PDU)							
Description	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Bit	8	CMD	Index LSB	Index MSB	Sub-Index	Data bytes			

Be aware that the LSB of Index and Data bytes are always transferred first. For example Index = 0x2104, Byte 1 = 0x04, Byte 2 = 0x21.
The PDU Format value is shall always be 0xEF.

10.2 CMD Codes

CMD-Codes	Description
0x22	Writing request, unspecified length
0x23	Writing request, 4 bytes
0x27	Writing request, 3 bytes
0x2B	Writing request, 2 bytes
0x2F	Writing request, 1 bytes
0x40	Read command

10.3 Response Frame format for PDU1

		Protocol Data Unit (PDU)							
Description	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Bit	8	Resp. Code	Index LSB	Index MSB	Sub-Index	Data of Object / Error-Code			

Be aware that the LSB of Index and Data bytes are always transferred first. For example Index = 0x2104, Byte 1 = 0x04, Byte 2 = 0x21.

10.4 Response Codes

Response-Code	Description
0x80	Error occurred -> Check Error-Code
0x60	Write command successfully
0x4F	Reading of 1 byte of data successfully
0x4B	Reading of 2 byte of data successfully
0x47	Reading of 3 byte of data successfully
0x43	Reading of 4 byte of data successfully

10.5 Error-Codes

Error-Code	Description
0x0504 0001	Sent CMD is unknown
0x0601 0000	Access to object unsupported
0x0601 0001	Read-Access not supported to object
0x0601 0002	Write-Access not supported to object
0x0602 0000	Object doesn‘t exist
0x0609 0011	Sub-Index doesn’t exist
0x0609 0030	Value out of Range
0x0609 0031	Value too high
0x0609 0032	Value too low

11 Object descriptions and examples

The frequently used objects are described in this chapter. More objects can be found in the chapter 12 Object directory. After setting the new entries a SAVE command (see paragraph 11.1.1 0x1010 Save parameters) followed by a reset or power-on is necessary to adopt the new value.

11.1 Save and restore parameters

The sensor loads the configuration parameters at the power on, so to make effective any configuration parameters change, the device *MUST* be power cycled.

11.1.1 0x1010 Save parameters

With the object 0x1010, the parameters can be saved to the non-volatile memory and applied at startup. The response is sent by the sensor immediately after receiving the "Save" command, before actually saving the parameters to the non-volatile memory. The device must not reset or turned off before the object 0x1010 has responded (see paragraph 11.1.3 Safe non-volatile operation).

Object 0x1010-01
Writing "save" (ASCII) to 0x1010-01 saves **all** parameters to the non-volatile memory.

Object 0x1010-02
Not used for J1939

Object 0x1010-03
Writing "save" (ASCII) to 0x1010-03 saves the **application** parameters (12.3 Standardized device profile area) to the non-volatile memory.

Object 0x1010-04
Writing "save" (ASCII) to 0x1010-04 saves the **manufacturer specific** parameters (12.2 Manufacturer specific profile area) to the non-volatile memory.

Example of saving all parameters:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
SAVE **all** parameters	0x18EF0400	8	23	10	10	**01**	73 *s*	61 *a*	76 *v*	65 *e*
Successful response (0x60)	0x18EF0004	8	60	10	10	01	00	00	00	00

11.1.2 0x1011 Restoring of default values

With the object 0x1011, the parameters can be restored to the factory preset values. The response is sent by the sensor immediately after receiving the "Restore" command, before actually restoring the parameters from the non-volatile memory. The device must not reset or turned off before the object 0x1011 has responded (see paragraph 11.1.3 Safe non-volatile operation).

Object 0x1011-01
Writing "load" (ASCII) to 0x1011-01 restores **all** parameters from the non-volatile memory.

Object 0x1011-02
Not used for J1939

Object 0x1011-03
Writing "load" (ASCII) to 0x1011-03 restores the **application** parameters (12.3 Standardized device profile area) from the non-volatile memory.

Object 0x1011-04

Writing “load” (ASCII) to 0x1011-04 restores the **manufacturer specific** parameters (12.2 Manufacturer specific profile area) from the non-volatile memory.

Example of restoring the application parameters:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
RESTORE **all** parameters	0x18EF0400	8	23	11	10	**01**	6C *l*	6F *o*	61 *a*	64 *d*
Successful response (0x60)	0x18EF0004	8	60	11	10	01	00	00	00	00

11.1.3 Safe non-volatile operation

To ensure safe non-volatile operation, the user must ensure no power interruption immediately after sending of the save command to object 0x1010-xx (otherwise, the factory values are restored at the next power up).
The time between access object 0x1010-01 or 0x1011-01 and a reset or power-off has to be at least 600ms.

11.1.4 Side effect

Save and Restore operations interrupt the updating of measurement.

11.2 0x1018-2 Read the product code

Product	Product-Code (Byte 4 and Byte 5)	Device Name
Cable transducer	0x0521	“GCA3-5” for hall effect based sensors “GCA8-12-20” for potentiometer based sensors
Cable transducer redundant (2-sensing)	0x0520	“GCA3-5” for hall effect based sensors “GCA8-12-20” for potentiometer based sensors
Cable transducer with inclination sensor integrated	0x0523	“GCA3-5” for hall effect based sensors “GCA8-12-20” for potentiometer based sensors
Cable transducer redundant (2-sensing) with inclination sensor integrated	0x0522	“GCA3-5” for hall effect based sensors “GCA8-12-20” for potentiometer based sensors

Byte 5: Major Product Code
Byte 4: Minor Product Code

Example of reading the product code:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Reading of product code	0x18EF0400	8	40	18	10	02	00	00	00	00
Successful response (0x60)	0x18EF0004	8	60	02	21	00	21	05	00	00

11.3 0x1009 Reading of the hardware version

The hardware version in ASCII encoded and null terminated. For example: “1.0” -> 31 2E 30 00

Example of reading the hardware version:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Reading of hardware version	0x18EF0400	8	40	09	10	00	00	00	00	00
Successful response (0x60)	0x18EF0004	8	43	09	10	00	00	30	2E	31

11.4 0x100A Reading of the software version

The software version in ASCII encoded and null terminated. For example: “1.2” -> 31 2E 32 00

Example of reading the software version:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Reading of software version	0x18EF0400	8	40	0A	10	00	00	00	00	00
Successful response (0x60)	0x18EF0004	8	43	0A	10	00	00	32	2E	31

11.5 0x3000 Baudrate

The following table describes all possible baud rate settings:

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

Example of changing baudrate to 50 kBit/s (value 6):

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Changing baudrate to	0x18EF0400	8	2F	00	30	00	06	00	00	00
Successful response (0x60)	0x18EF0004	8	60	00	30	00	00	00	00	00

Remark: the new parameter value will be effective at the next power cycle.

11.6 Length direction

There are 2 ways for changing direction.

11.6.1 First method

Send the following message.

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Writing length direction	0x18EF0400	8	2B	00	60	00	**PAR**	00	00	00
Successful response (0x60)	0x18EF0004	8	60	00	60	00	00	00	00	00

Where **PAR** is the parameter setting as follows:

PAR	MEANING
0x00	Positive direction (length increase pulling the cable), length expressed in unit of 0.1mm
0x01	Negative direction (length decrease pulling the cable), length expressed in unit of 0.1mm
0x04	Positive direction (length increase pulling the cable), resolution depends on object 0x6005
0x05	Negative direction (length decrease pulling the cable), resolution depends on object 0x6005

Remark: the new parameter value will be effective at the next power cycle.

11.6.2 Second method

Send the following message.

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Writing length direction	0x18EF0400	8	2F	02	21	01	**DIR**	00	00	00
Successful response (0x60)	0x18EF0004	8	60	02	21	01	00	00	00	00

Where **DIR** is the direction: 0 = positive (0 to Full Scale output), 1 = negative (0 to minus Full Scale)
If it is necessary to set negative direction (Full Scale output to 0), the object 0x2119-00 must be set to 0.

Remark: the new parameter value will be effective at the next power cycle.

11.7 0x6005-01 Length resolution

Set the parameters as explained in paragraph 11.6, then send the following message.

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Writing length resolution	0x18EF0400	8	23	05	60	01	R0	R1	R2	R3
Successful response (0x60)	0x18EF0004	8	60	05	60	01	00	00	00	00

Where:
R0...R3 is the measurement unit of the output length, expressed in nm (Nanometer).

For example, to obtain a resolution of 1mm:
1mm = 1’000’000nm = 0x000F4240
Thus: R0 = 0x40, R1 = 0x42, R2 = 0x0F, R3 = 0x00

I.e.: if it's necessary to have the measure expressed in 10mm, the value to write must be 10'000'000nm, to obtain instead a resolution of 0.1mm, the value to write will be 100'000nm.

Note: it's necessary to be sure that in the object 0x6000 the value present is 4 or 5.

Remark: the new parameter value will be effective at the next power cycle.

11.8 0x6010-XX Length preset

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Writing preset value on channel **CH**	0x18EF0400	8	23	10	60	**CH**	R0	R1	R2	R3
Successful response (0x60)	0x18EF0004	8	60	10	60	**CH**	00	00	00	00

Where:
CH is the channel index (1 for first channel, 2 for second channel)
R0...R3 represent the desired output length, expressed in according with the resolution.

For example, if the resolution is 1mm and the desired value in that position is 300mm:
300 = 0x0000012C
Thus: R0 = 0x2C, R1 = 0x01, R2 = 0x00, R3 = 0x00

If the resolution is 0.1mm and the desired value in that position is 300mm:
3000 = 0x00000BB8
Thus: R0 = 0xB8, R1 = 0x0B, R2 = 0x00, R3 = 0x00

To set a preset value to 0 (zero) it's only necessary to set R = 0x00000000 (0 decimal).

Remark: the new parameter value will be effective at the next power cycle.

11.9 0x2103 Digital low pass filter configuration for length

The digital Length low pass filter's cut-off frequency can be adjusted through the object 0x2103. The IIR coefficient is expressed in %. The value can be selected in a range from 0 to 100.

The default coefficient is 20.

Example of changing the cut-off filter frequency:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Writing cut-off frequency	0x18EF0400	8	2B	03	21	00	R0	R1	00	00
Successful response (0x60)	0x18EF0004	8	60	03	21	00	00	00	00	00

Where:
R0...R1 represent the cut-off filter frequency expressed in %.

For example, to write a cut-off frequency of 10%:
10% = 0xA
Thus: R0 = 0x0A, R1 = 0x00

Remark: the new parameter value will be effective at the next power cycle.

11.10 0x6800 Angle resolution

This object indicates the resolution of the inclinometer of the Slope long16 (object 0x6810) based on 0,001°. This resolution is also valid for the 32-bit value objects (0x6910). In case of low resolution, the value is 10. In case of high resolution the value is 1.

The following table describes all possible resolutions:

Resolution (6000h)	
Value	**Description**
0x01 (1)	0.001°
0x0A (10)	0.01°
0x64 (100)	0.1°
0x3E8 (1000)	1°

Example of changing the angle resolution of the Slope long16:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Writing angle resolution	0x18EF0400	8	2B	00	68	00	R0	R1	00	00
Successful response (0x60)	0x18EF0004	8	60	00	68	00	00	00	00	00

Where:
R0...R1 is the resolution of the slope based on 0.001°.

For example, to obtain a resolution of 0.1°:
0.1° -> 100= 0x0064
Thus: R0 = 0x64, R1 = 0x00

Remark: the new parameter value will be effective at the next power cycle.

11.11 Angle offset parameters

11.11.1 Operating parameters

The operating parameters influence the output inclination value as follows.

Bit Mask:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Reserved						s	i
Default	-						1	0

i = Inversion (0 = Do not enable inversion; 1 = Enable inversion)
s = Scaling (0 = Do not enable scaling; 1 = Enable scaling)

Scaling means that the following equation is applied:

Inclination = A + B + C

where

A is a physically measured angle;
B is a differential slope offset;
C is a slope offset.

If scaling is disabled, the Slope value is equal to the physically measured angle.

The operating parameters are applied for the according slope. The table below shows the correspondences between the object and the influenced slope.

Operating parameters	
Object	**Slope**
0x6811	Longitudinal, 16-bit (0x6810)
0x6911	Longitudinal, 32-bit (0x6910)

The 16bit and 32bit values are hardwired internally (i.e. changing the operating parameter at 0x6811 changes the operating parameter at 0x6911).

Example of changing the operating parameters of the Slope long16:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Writing operating parameters	0x18EF0400	8	2F	11	68	00	R0	00	00	00
Successful response (0x60)	0x18EF0004	8	60	11	68	00	00	00	00	00

Where:
R0 represent the desired value, expressed in according with the bit mask.

For example, to enable inversion and scaling:
Bit 0: 1, Bit 1: 1
Thus: R0 = 0x03

Remark: the new parameter value will be effective at the next power cycle.

11.11.2 Angle preset

Accessing this object sets directly the actual slope value to a desired slope value. The value shall be given in angular degrees with the resolution given in object 0x6800.

The preset values are applied for the according slope. The table below shows the correspondences between the object and the influenced slope.

Preset	
Object	**Slope**
0x6812	Longitudinal, 16-bit (0x6810)
0x6912	Longitudinal, 32-bit (0x6910)

The 16bit and 32bit values are hardwired internally (i.e. changing the preset value at 0x6812 changes the preset value at 0x6912).

Example of setting the zero of the Slope long16:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Writing offset parameter	0x18EF0400	8	2B	12	68	00	00	00	00	00
Successful response (0x60)	0x18EF0004	8	60	12	68	00	00	00	00	00

Remark: the new parameter value will be effective at the next power cycle.

11.11.3 Differential offset

This object indicates the shifting of the Slope value independent of the Angle preset value and the Offset value. The value shall be given in angular degrees with the resolution given in object 0x6800.

The differential offset values are applied for the according slope. The table below shows the correspondences between the object and the influenced slope.

Differential offset	
Object	**Slope**
0x6814	Longitudinal, 16-bit (0x6810)
0x6914	Longitudinal, 32-bit (0x6910)

The 16bit and 32bit values are hardwired internally (i.e. changing the differential offset at 0x6814 changes the differential offset at 0x6914).

Example of setting the differential offset of the Slope long16:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Writing differential offset	0x18EF0400	8	2B	14	68	00	R0	R1	00	00
Successful response (0x60)	0x18EF0004	8	60	14	68	00	00	00	00	00

Where:
R0...R1 is the desired differential offset, expressed in according with the resolution (object 0x6800).

For example, to write a differential offset of 10° and assuming a resolution of 0.1°:
10° -> 100= 0x0064
Thus: R0 = 0x64, R1 = 0x00

Remark: the new parameter value will be effective at the next power cycle.

11.12 0x2603 Digital low pass filter configuration for angle

The digital angle low pass filter cut-off frequency can be adjusted by object 0x2603. The IIR coefficient is expressed in tenths of hertz (e.g.: Fc = 3 Hz -> 30 (0x1E)). The value can be selected in a range from 0 to 300 (30Hz), where the value 0 disables the filter.

The default cut-off filter frequency is 3.0Hz.

Example of changing the cut-off filter frequency of the Slope long16:

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Writing cut-off frequency	0x18EF0400	8	2B	03	26	00	R0	R1	00	00
Successful response (0x60)	0x18EF0004	8	60	03	26	00	00	00	00	00

Where:
R0...R1 represent the cut-off filter frequency based on 0.1Hz.

For example, to write a cut-off frequency of 0.4Hz:
0.4Hz -> 4 = 0x4
Thus: R0 = 0x04, R1 = 0x00

Remark: the new parameter value will be effective at the next power cycle.

11.13 0x3102 ECU address

The ECU address is a 8-Bit value, which is transmitted in Byte 4.

Example of changing the ECU address to 0x40:

Let's suppose we have a Cyclic PGN 65363 with ECU address 0x04. The identifier is 0x18FF00**04.**

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Changing ECU address	0x18EF0400	8	2F	02	31	00	**40**	00	00	00
Successful response (0x60)	0x18EF0004	8	60	02	31	00	00	00	00	00

The new identifier of the Cyclic PGN 65363 with ECU address 0x40 is now 0x18FF00**40**

Remark: the new parameter value will be effective at the next power cycle.

Example of reading the ECU address (0x04):

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Reading ECU address	0x18EF0400	8	40	02	31	00	00	00	00	00
Successful response (0x60)	0x18EF0004	8	4F	02	31	00	**04**	00	00	00

11.14 0x3103/0x3106 Length/Angle Group Extension

The group extension is a 8-Bit value, which is transmitted in Byte 4.

Example of changing the Length Group Extension (0x3103) to 0x33:

Let's suppose we have a Cyclic PGN with Group Ext. 0x53 (PGN 65363). The identifier is 0x18FF**53**0A.

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Changing Group Extension	0x18EF0400	8	2F	03	31	00	**33**	00	00	00
Successful response (0x60)	0x18EF0004	8	60	03	31	00	00	00	00	00

The new identifier of the Cyclic PGN with Group Ext. 0x33 (PGN 65331) is now: 0x18FF**33**0A.

Remark: the new parameter value will be effective at the next power cycle.

Example of reading Length Group Extension (0x53):

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Reading Group Extension	0x18EF0400	8	40	03	31	00	00	00	00	00
Successful response (0x60)	0x18EF0004	8	4F	03	31	00	**53**	00	00	00

11.15 0x3104/0x3107 Transmission rate for PGN 65363/PGN 65364

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 Length Transmission Rate (0x3104) to 500ms (0x1F4):

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Changing transmission time	0x18EF0400	8	2B	04	31	00	**F4**	**01**	00	00
Successful response (0x60)	0x18EF0004	8	60	04	31	00	00	00	00	00

Remark: the new parameter value will be effective at the next power cycle.

Example of reading Length Transmission time for PGN 65363 (0x1F4):

			Protocol Data Unit (PDU) Byte 0 … 7 [hex]							
Description	**Identifier**	**DLC**	**Cmd**	**Obj. LSB**	**Obj. MSB**	**Sub-index**	**Data 0**	**Data 1**	**Data 2**	**Data 3**
Reading transmission time	0x18EF0400	8	40	04	31	00	00	00	00	00
Successful response (0x60)	0x18EF0004	8	4B	04	31	00	**F4**	**01**	00	00

11.16 0x6503 Sensor Diagnostics

In case of sensor failure, this object provides information about the failure. It is a bit field of 16 bits where each bit indicates the fault detected. If an fault occurs, the value provided by the sensor is no longer valid (the output value remains fixed at the last valid value) and the according bit indicates the error. When the fault ceases and the sensor is able to provide an accurate position value, the error is cleared.

Bit	Description	Meaning
0	Position error	Position error occurred (wire break or over stroke or communication error)
1	-	Not used
2	-	Reserved
3	-	Reserved
4	-	Reserved
5	-	Reserved
6	-	Reserved
7	-	Reserved
8	-	Reserved
9	-	Reserved
10	-	Reserved
11	-	Reserved
12	Wire break error	Set when wire break or over stroke error occurred
13	Length sensor error	Set when communication with the length sensor fails
14	Accelerometer sensor error	Set when communication with the accelerometer sensor fails
15	-	Not used

12 Object directory

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

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	Yes = can be stored in the EEPROM, No = cannot be stored in the EEPROM
Description	Additional information

12.1 Communication profile area

Object	Sub-index	Name	Format	Access	Default value	Saved	Description
0x1000	0	Device type	U32	ro	0x07080196	No	As per CANopen CiA406
0x1008	0	Device name	STR	ro	"GCA3-5" for hall effect based sensors "GCA8-12-20" for potentiometer based sensors	No	Device designation
0x1009	0	HW version	STR	ro	1.0	No	Hardware version
0x100A	0	FW version	STR	ro	-	No	Software version (ASCII characters, i.e. version 1.28 = 31 56 32 38)
0x1010	0	Numbers of save-options	U8	ro	0x04	No	See paragraph 11.1.1 for details
	1	Save all parameters	U32	rw	0x00000001	Yes	The parameters are saved only writing the key string "save" (0x73-0x61-0x76-0x65)
	3	Save application parameters	U32	rw	0x00000001	Yes	The parameters are saved only writing the key string "save" (0x73-0x61-0x76-0x65)
	4	Save manufacturer specific parameters	U32	rw	0x00000001	Yes	The parameters are saved only writing the key string "save" (0x73-0x61-0x76-0x65)
0x1011	0	Numbers of restore-options	U8	ro	0x04	No	See paragraph 11.1.2 for details
	1	Reset for all parameters	U32	rw	0x00000001	Yes	If the key string "load" (0x6C-0x6F-0x61-0x64) is entered here, the parameters are assigned to the factory default values and are valid after the next reset.
	3	Reset application parameters	U32	rw	0x00000001	Yes	If the key string "load" (0x6C-0x6F-0x61-0x64) is entered here, the parameters are assigned to the factory default values and are valid after the next reset.
	4	Reset manufacturer specific parameters	U32	rw	0x00000001	Yes	If the key string "load" (0x6C-0x6F-0x61-0x64) is entered here, the parameters are assigned to the factory default values and are valid after the next reset.
0x1018	0	Numbers of identity-options	U8	ro	0x04	No	
	1	Vendor ID	U32	ro	0x0000005F	No	Baumer
	2	Product code	U32	ro	0x520 * 0x521 0x522 * 0x523	No	GCAxx series products Redundant without inclination sensor: 0x520 Single without inclination sensor: 0x521 Redundant with inclination sensor: 0x522 Single with inclination sensor: 0x523
	3	Revision number	U32	ro	0x00000000	No	
	4	Serial number	U32	ro	-	No	Depending by the SN of the product

12.2 Manufacturer specific profile area

Object	Sub-index	Name	Format	Access	Default value	Save	Description
0x2000	0	Length raw channel value	U8	ro	2	No	Number of the entries
	1	Length raw value channel 1	U32	ro, m	-	No	length in 0.1mm (internal raw value)

Object	Sub-index	Name	Format	Access	Default value	Save	Description
	2	Length raw value channel 2*	U32	ro, m	-	No	length in 0.1mm (internal raw value)
0x2101	0	Length preset values	U8	ro	0x04	No	Number of the entries
	1	Length preset value Channel 1	I32	rw	0x00000000	Yes	This object is the same as the 0x6010 object
	2	Length preset value Channel 2*	I32	rw	0x00000000	Yes	This object is the same as the 0x6010 object
	3	Custom preset value Channel 1	I32	rw	0x00000000	Yes	-
	4	Custom preset value Channel 2*	I32	rw	0x00000000	Yes	-
0x2102	0	Length direction	U8	ro	2	No	Number of the entries
	1	Length direction channel 1	U8	rw		Yes	direction (0=increasing when pull; 1= decreasing when pull). See paragraph 11.6 for details
	2	Length direction channel 2*	U8	rw		Yes	direction (0=increasing when pull; 1= decreasing when pull). See paragraph 11.6 for details
0x2103	0	Length filter strength	U8	rw	0x14 for hall effect based sensors 0x50 for potentiometer based sensors	Yes	IIR coefficient expressed in % (lower values-> lower cut off frequency)
0x2105	0	Speed filter strength	U8	rw	0x8 for hall effect based sensors 0x50 for potentiometer based sensors	Yes	IIR coefficient expressed in % (lower values-> lower cut off frequency)
0x2110	0	Length value	U8	ro	4	No	Number of the entries
	1	Length value channel 1	I32	ro, m	-	No	This object is the same as the 0x6020 object
	2	Length value channel 2*	I32	ro, m	-	No	This object is the same as the 0x6020 object
	3	Inverted length value channel 1	I32	ro, m	-	No	opposite value (FS-value)
	4	Inverted length value channel 2*	I32	ro, m	-	No	opposite value (FS-value)
0x2111	0	16-bit length value	U8	ro	4	No	Number of the entries
	1	16-bit length value channel 1	U16	ro, m	-	No	16-bit version of object 0x2110
	2	16-bit length value channel 2*	U16	ro, m	-	No	16-bit version of object 0x2110
0x2116	0	Biggest subindex	U8	ro	0x02	No	Number of entries
	1	Speed value 16-bit channel 1	U16	ro, m		No	-
	2	Speed value 16-bit channel 2 *	U16	ro, m		No	-
0x2117	0	Reserved	U16	ro, m		No	For internal use (sensor diagnostic flags)
0x2119	0	Inversion Length Behavior	U8	rw	1 for hall effect based sensors 0 for potentiometer based sensors	Yes	See paragraph 11.6 for details
0x2195	0	FW version	U16	ro, m	-	No	This object is the value of object 0x100A in 16 bit. E.g.: If object 0x100A = “1.28”, then 0x2195 = 0x80 = 128.
0x2196	0	Customer specific area	U32	rw, m	0x00000000	Yes	4 byte of empty space for customer specific content (e.g. customer name)
0x2603	0	Angle filter strength	U16	rw	0x1E		IIR coefficient expressed in Hz (lower values->more filtering)
0x3000	0	Baud rate	U8	rw	0x03	Yes	0=1000 kbits/s 1=800 kbits/s 2=500 kbits/s

Object	Sub-index	Name	Format	Access	Default value	Save	Description
							3=250 kbits/s 4=125 kbits/s 5=100 kbits/s 6=50 kbits/s 7=20 kbits/s 8=10 kbits/s The baud rate is activated after a reset or power-on (if parameter is saved to non-volatile memory)
0x3102	0	ECU default address	U8	rw	0x04 (0x05)	Yes	ECU Address. Address-Range: 0 … 253 Single products are equipped with one ECU address: Channel A: ECU address 4 (default factory setting 0x04) Full redundant products (2-channel) are equipped with two ECU addresses (each channel has its own ECU address): Channel A: ECU address 4 (default factory setting 0x04) Channel B: ECU address 5 (default factory setting 0x05)
0x3103	0	Length Group extension	U8	rw	0x53	Yes	Group Extension, used for length PDU (proprietary B)
0x3104	0	Length Transmission rate	U16	rw	0x64	Yes	Transmission Rate for PGN 65363 [ms]
0x3106	0	Angle Group extension	U8	rw	0x54	Yes	Group Extension, used for angle PDU (proprietary B)
0x3107	0	Angle Transmission rate	U16	rw	0x64	Yes	Transmission Rate for PGN 65364 [ms]

* This value is available only for redundant version (2-sensing). For the full redundant version (2-channel, two separate ECU addresses), this value is available on the respective ECU address.

12.3 Standardized device profile area

Object	Sub-index	Name	Format	Access	Default value	Save	Description
0x6000	0	Length parameters	U16	rw	0x0000	Yes	See paragraph 11.6 for details on changing length direction
0x6003	0	Length preset value Channel 1	I32	rw	0x00000000	Yes	Set the zero length values
0x6004	0	Length value Channel 1	I32	ro, m	-	No	Length value
0x6005	0	Length position setting	U8	ro	0x02	No	Number of the entries
	1	Length position setting	U32	rw	0x0000F424 0	Yes	The measurement unit of resolution is nanometers (see paragraph 11.7 for details)
	2	Speed setting	U32	rw	0x0000F424 0	Yes	The measurement unit of resolution is nanometers (see paragraph 11.7 for details)
0x6010	0	Length preset values	U8	ro	0x02	No	Number of the entries
	1	Length preset value Channel 1	I32	rw	0x00000000	Yes	length in 0.1mm (or other scale depending on resolution)
	2	Length preset value Channel 2*	I32	rw	0x00000000	Yes	length in 0.1mm (or other scale depending on resolution)
0x6020	0	Length values	U8	ro	0x02	No	Number of the entries
	1	Length value Channel 1	I32	ro, m	-	No	length in 0.1mm (or other scale depending on resolution)
	2	Length value Channel 2*	I32	ro, m	-	No	length in 0.1mm (or other scale depending on resolution)
0x6030	0	Speed values	U8	ro	0x02	No	Number of the entries
	1	Speed value Channel 1	I32	ro, m	-	No	-
	2	Speed value Channel 2	I32	ro, m	-	No	-
0x6503	0	Alarm	U16	ro, m	0x00	No	See paragraph 11.16 for details
0x6800	0	Resolution Angle**	U16	rw	0x0064	Yes	This object shall indicate the resolution of Slope long16-bit (object 0x6810) based on 0,001°.

Object	Sub-index	Name	Format	Access	Default value	Save	Description
							This resolution is also valid for the 32-bit value objects (0x6910).
0x6810	0	Slope Long 16-bit**	I16	ro, m	-		This object shall provide the 16-bit slope value of the longitudinal axis. The value shall be given in degree (angle) with the resolution given in object 0x6800.
0x6811	0	Slope long 16-bit operating parameter**	U8	rw	0x02	Yes	If scaling is enabled, the Slope long16-bit value shall be calculated accordingly to the following equation: Slope long16-bit = physically measured angle + Differential slope long16-bit offset + Slope long16-bit offset If scaling is disabled, the Slope long16-bit value shall be equal to the physical measured angle.
0x6812	0	Slope long 16-bit preset value**	I16	rw	0x0000	Yes	Accessing this object shall set directly the actual longitudinal slope value to a desired longitudinal slope value. The calculated application-offset of the longitudinal slope value is given in Slope long16-bit offset (object 0x6813). The Slope long16-bit offset is calculated with respect to object 0x6814. The value shall be given in degree (angle) with the resolution given in object 0x6800.
0x6813	0	Slope long 16-bit offset**	I16	ro	0x0000		This object shall indicate the application-offset of the longitudinal axis. The value shall be given in degree (angle) with the resolution given in object 0x6800. The following equation shall be applied: Slope long16-bit offset = Slope long16-bit preset value at tacc – slope physical measured at tacc – Differential slope long16-bit offset (tacc = time when accessing object 0x6812)
0x6814	0	Differential slope long 16-bit offset**	I16	rw	0x0000	Yes	This object shall shift the Slope long16-bit value (object 0x6810) independent of Slope long16-bit preset value (object 0x6812) and Slope long16-bit offset (object 0x6813). The value shall be given in degree (angle) with the resolution given in object 0x6800.
0x6910	0	Slope Long 32-bit**	I32	ro, m			See description of object 0x6810
0x6911	0	Slope long 32-bit operating parameter**	U8	rw	0x02	Yes	See description of object 0x6811
0x6912	0	Slope long 32-bit preset value**	I32	rw	0x00000000	Yes	See description of object 0x6812
0x6913	0	Slope long 32-bit offset**	I32	ro	0x00000000		See description of object 0x6813
0x6914	0	Differential slope long 32-bit offset**	I32	rw	0x00000000	Yes	See description of object 0x6814
0x6E11	0	Device temperature**	I16	ro, m	0x0000	No	Internal device temperature of inclination sensor
0x7010	0	Slope Long 16-bit Channel 2***	I16	ro, m	-		This object shall provide the 16-bit slope value of the longitudinal axis. The value shall be given in degree (angle) with the resolution given in object 0x6800.
0x7011	0	Slope long 16-bit operating parameter Channel 2***	U8	rw	0x02	Yes	If scaling is enabled, the Slope long16-bit value shall be calculated accordingly to the following equation: Slope long16-bit = physically measured angle + Differential slope long16-bit offset + Slope long16-bit offset If scaling is disabled, the Slope long16-bit value shall be equal to the physical measured angle.
0x7012	0	Slope long 16-bit preset value Channel 2***	I16	rw	0x0000	Yes	Accessing this object shall set directly the actual longitudinal slope value to a desired longitudinal slope value. The calculated application-offset of the longitudinal slope value is given in Slope long16-bit offset (object 0x7013). The Slope long16-bit offset is calculated with respect to object 0x7014. The value shall be given in degree (angle) with the resolution given in object 0x6800.

Object	Sub-index	Name	Format	Access	Default value	Save	Description
0x7013	0	Slope long 16-bit offset Channel 2***	I16	ro	0x0000		This object shall indicate the application-offset of the longitudinal axis. The value shall be given in degree (angle) with the resolution given in object 0x6800. The following equation shall be applied: Slope long16-bit offset = Slope long16-bit preset value at tacc – slope physical measured at tacc – Differential slope long16-bit offset (tacc = time when accessing object 0x7012)
0x7014	0	Differential slope long 16-bit offset Channel 2***	I16	rw	0x0000	Yes	This object shall shift the Slope long16-bit value (object 0x7010) independent of Slope long16-bit preset value (object 0x7012) and Slope long16-bit offset (object 0x7013). The value shall be given in degree (angle) with the resolution given in object 0x6800.
0x7110	0	Slope Long 32-bit Channel 2***	I32	ro, m			See description of object 0x7010
0x7111	0	Slope long 32-bit operating parameter Channel 2***	U8	rw	0x02	Yes	See description of object 0x7011
0x7112	0	Slope long 32-bit preset value Channel 2***	I32	rw	0x00000000	Yes	See description of object 0x7012
0x7113	0	Slope long 32-bit offset Channel 2***	I32	ro	0x00000000		See description of object 0x7013
0x7114	0	Differential slope long 32-bit offset Channel 2***	I32	rw	0x00000000	Yes	See description of object 0x7014
0x7511	0	Device temperature Channel 2***	I16	ro, m	0x0000	No	Internal device temperature of inclination sensor

* This value is available only for redundant version (2-sensing). For the full redundant version (2-channel, two separate ECU addresses), this value is available on the respective ECU address.

** The inclinometer objects 0x6800...0x6914 and 0x6E11 exist only for products with integrated inclinometer.

*** The inclinometer objects 0x7010...0x7114 and 0x7511 exist only for products with integrated inclinometer and redundant version (2-sensing).