

Content

| | |
|---|----|
| 1 CANopen | 2 |
| 1.1 EDS Files | 2 |
| 1.2 Features | 2 |
| 1.2.1 Basic information | 2 |
| 1.2.2 Basics based on CiA DS-301, V4.2.0 | 2 |
| 1.2.3 Basics based on CiA DSP-406, V3.2 | 3 |
| 1.2.4 Basics SDO communication | 3 |
| 1.2.5 Basics PDO communication based on CiA 301, V4.2.0 | 3 |
| 1.3 Object Library | 4 |
| 1.3.1 Communication Profile Area based on DS 301 V4.2.0 | 4 |
| 1.3.2 Device Profile Area | 6 |
| 1.3.3 Manufacturer specific Area | 8 |
| 1.4 Explanations to Object Library | 9 |
| 1.4.1 Object 0x6300 Encoder Cams | 9 |
| 1.4.2 Cam state registers | 9 |
| 1.4.3 Object 0x6400 Work Area | 9 |
| 1.4.3.1 Work Area Supervision | 9 |
| 1.4.3.2 Work Area State | 9 |
| 1.5 LSS / Layer Setting Service | 9 |
| 1.5.1 Configuration of Node-ID | 10 |
| 1.5.2 Configuration of Bit Rate | 10 |
| 1.5.3 Store Configuration Data | 11 |
| 1.6 SDO Services | 11 |
| 1.6.1 SDO Download | 11 |
| 1.6.2 SDO Upload | 12 |
| 1.6.3 SDO Abort | 12 |
| 1.7 Process Data PDO | 12 |
| 1.7.1 PDO Default Setting | 12 |
| 1.7.2 PDO Parameter Setting | 12 |
| 1.8 Error Handling | 14 |
| 1.8.1 Emergency Messages | 14 |
| 1.9 Error Objects | 15 |
| 1.9.1 Manufacturer-specific Status | 15 |
| 1.10 Non-Volatile Storage and Data Restoration | 16 |
| 1.11 Abbreviations | 17 |
| 1.12 Document Changes | 17 |

1 CANopen

This document reflects the Novotechnik sensor protocol implementation of the standard CANopen protocol. A basic knowledge of the CAN Bus is required for a proper understanding of this document. Most of the definitions made are according to the following CiA Standard specifications. For making use of all the features that these specifications offer, a knowledge about them is absolutely necessary. The sensor supports the CANopen Communication profile DS-301, V4.2.0, Encoder profile DSP-406, V3.2 and Layer Setting Services (LSS) DSP-305, V1.1.2.

1.1 EDS Files

For integration in a common CANopen projecting tool, electronic data sheet (*.eds) files are provided. These files can be downloaded from the Novotechnik Web Site, see Downloads/Operating manuals where also this document can be found.

⇒ Electric data sheet see file *Product series_CANopen*.

1.2 Features

1.2.1 Basic information

Vendor ID: 386 = 0x0182 (Novotechnik)
 Product code: TP1: 04035 = 0x0FC3, TH1: 04042 = 0x0FCA, TM1: 04228 = 0x1084, TF1: 04052 = 0x0FD4
 Rev.-No.: f.e 196613 = 0x30005
 Serial No.: see product label, "YYMMxxxx"

1.2.2 Basics based on CiA DS-301, V4.2.0

| CAN Identifier | Standard 11 bits according to pre-defined connection set: <table border="1"> <thead> <tr> <th>Services</th> <th>COB-ID</th> </tr> </thead> <tbody> <tr> <td>NMT</td> <td>0x00</td> </tr> <tr> <td>SYNC</td> <td>0x080</td> </tr> <tr> <td>EMCY</td> <td>0x080 + Node-ID</td> </tr> <tr> <td>PDO1 (Tx)</td> <td>0x180 + Node-ID</td> </tr> <tr> <td>PDO2 (Tx)</td> <td>0x280 + Node-ID</td> </tr> <tr> <td>SDO (Rx)</td> <td>0x600 + Node-ID</td> </tr> <tr> <td>SDO (Tx)</td> <td>0x580 + Node-ID</td> </tr> </tbody> </table> | Services | COB-ID | NMT | 0x00 | SYNC | 0x080 | EMCY | 0x080 + Node-ID | PDO1 (Tx) | 0x180 + Node-ID | PDO2 (Tx) | 0x280 + Node-ID | SDO (Rx) | 0x600 + Node-ID | SDO (Tx) | 0x580 + Node-ID |
|----------------------|---|----------|--------|-----|------|------|-------|------|-----------------|-----------|-----------------|-----------|-----------------|----------|-----------------|----------|-----------------|
| Services | COB-ID | | | | | | | | | | | | | | | | |
| NMT | 0x00 | | | | | | | | | | | | | | | | |
| SYNC | 0x080 | | | | | | | | | | | | | | | | |
| EMCY | 0x080 + Node-ID | | | | | | | | | | | | | | | | |
| PDO1 (Tx) | 0x180 + Node-ID | | | | | | | | | | | | | | | | |
| PDO2 (Tx) | 0x280 + Node-ID | | | | | | | | | | | | | | | | |
| SDO (Rx) | 0x600 + Node-ID | | | | | | | | | | | | | | | | |
| SDO (Tx) | 0x580 + Node-ID | | | | | | | | | | | | | | | | |
| CAN Bit rates | Bit rate is defined in the ordering code: 6_8: 20 kBaud 6_7: 50 kBaud 6_5: 125 kBaud 6_4: 250 kBaud 6_3: 500 kBaud 6_2: 800 kBaud 6_1: 1000 kBaud setting per LSS (see chapter 1.5 LSS / Layer Setting Service) or object 0x2001 (see chapter 1.6 SDO Services) | | | | | | | | | | | | | | | | |
| Node-ID | 0x7F setting per LSS (see chapter 1.5 LSS / Layer Setting Service) or object 0x2000 (see chapter 1.6 SDO Services) | | | | | | | | | | | | | | | | |
| SYNC | Consumer | | | | | | | | | | | | | | | | |
| Time Stamp | no | | | | | | | | | | | | | | | | |
| Emergency Messages | Producer | | | | | | | | | | | | | | | | |
| Node Guarding | yes | | | | | | | | | | | | | | | | |
| Heartbeat | Producer | | | | | | | | | | | | | | | | |
| Non-volatile storage | yes | | | | | | | | | | | | | | | | |
| Program Download | no | | | | | | | | | | | | | | | | |
| NMT Service | Slave | | | | | | | | | | | | | | | | |

1.2.3 Basics based on CiA DSP-406, V3.2

| | |
|---|---|
| Encoder class | <input type="checkbox"/> C1 <input checked="" type="checkbox"/> C2 |
| Encoder type | Absolute Linear Encoder |
| Max. bit bandwidth of position value | 32 bits |
| Encoder Cams Channels to be detected Cams per channel Polarity Hysteresis | Pos.ch1, Pos.ch2 (ch2: only product series TP1 / TH1) 4 invertable yes |
| Work Area Supervision channels | Pos.ch1, Pos.ch2 (ch2: only product series TP1 / TH1) |

1.2.4 Basics SDO communication

| | |
|--------------------|----------|
| SDO communication | 1 Server |
| expedited transfer | yes |
| segmented transfer | no |
| Block transfer | no |

1.2.5 Basics PDO communication based on CiA 301, V4.2.0

| | |
|-----------------------------|--|
| PDO communication principle | Producer |
| TPDO's | TPDO1: <ul style="list-style-type: none"> • Event-driven transmission <ul style="list-style-type: none"> - only product series TP1 / TH1 / TM1: Synchronous to measurement cycle (measuring length dependent 0.5/1/2/4 ms with delay 100 µs) - Event Timer • Synchronous (SYNC Object) TPDO2: <ul style="list-style-type: none"> • Synchronous (SYNC Object) |
| PDO Mapping | dynamic |
| max. PDO Mappings per PDO | 5 |

1.3 Object Library

1.3.1 Communication Profile Area based on DS 301 V4.2.0

| Object description | | | Entry description | | | |
|--------------------|------------------------------|----------------|-------------------|----------------|----------------------------------|---|
| Index/ subindex | Name | Data Type | Access | PDO Mapping | Default value | Comment |
| 1000 | device type | unsigned32 | const | no | 0x000A0196 | Device profile 406 multi-sensor encoder interface |
| 1001 | error register | unsigned8 | ro | no | 0x00 | See chapter <i>1.8 Error Handling</i> |
| 1002 | manufacturer status register | unsigned32 | ro | no | 0x00000000 | Additional manufacturer spec. status register |
| 1005 | COB-ID SYNC | unsigned32 | rw | no | 0x00000080 | COB-ID SYNC message (CAN-identifier) |
| 1008 | manufacturer device name | visible_string | const | no | e.g. TP1-0100-101-614-105 | Device name, see datasheet/ordering code |
| 1009 | hardware version | visible_string | const | no | - | Only product series TM1 / TF1: hardware version |
| 100A | software version | visible_string | const | no | - | Only product series TM1 / TF1: software version |
| 100C | guard time | unsigned16 | rw | no | 0x00000000 disabled | Time base (in ms), which gives combined with 100D the time in which the response of the node guard is expected |
| 100D | life time factor | unsigned8 | rw | no | 0x00000000 disabled | The life time factor multiplied with the guard time gives the life time for the device. |
| 1010 | store parameter field | unsigned32 | | | | This entry supports saving of parameters in non volatile memory. With a read access the device provides information about its saving capabilities. For saving the signature "save" (0x65766173) must be written. |
| 1010/1 | store parameter field | unsigned32 | rw | no | 0x00000000 | Save all parameters |
| 1010/2 | | unsigned32 | rw | no | 0x00000000 | Save communication parameters |
| 1010/3 | | unsigned32 | rw | no | 0x00000000 | Save application parameters |
| 1010/4 | | unsigned32 | rw | no | 0x00000000 | Save user data parameters |
| 1010/5 | | Unsigned32 | rw | no | 0x00000000 | Save manufacturer defined parameters |
| 1011 | restore default parameters | unsigned32 | | | | This entry supports restoring of default parameters. With a read access the device provides information about its capabilities to restore these values. For restoring the signature "load" (0x64616f6c) must be written. |
| 1011/1 | | unsigned32 | rw | no | 0x00000000 | Restore all default parameters |
| 1011/2 | | unsigned32 | rw | no | 0x00000000 | Restore communication default parameters |
| 1011/3 | | unsigned32 | rw | no | 0x00000000 | Restore application default parameters |
| 1011/4 | | unsigned32 | rw | no | 0x00000000 | Restore user data parameters |
| 1014 | COB-ID EMCY | unsigned32 | ro | no | 0x00000080 + Node-ID | COB-ID used for emergency message (Emergency Server). |
| 1017 | producer heart-beat time | unsigned16 | rw | no | 0x0000 disabled | Heartbeat time periode in ms. Range 0 ... 0xFFFF |
| 1018 | identify object | identity | | | | General information about the device. This information is also used as the LSS address when using "switch mode selective" command |
| 1018/1 | | unsigned32 | ro | no | 0x00000182 | Vendor ID |
| 1018/2 | | unsigned32 | ro | no | 0x0 (see 1.2.1Basic information) | Product code |
| 1018/3 | | unsigned32 | ro | no | (see 1.2.1Basic information) | Revision number |
| 1018/4 | | unsigned32 | ro | | (see 1.2.1Basic information) | Serial number |

| Object description | | | Entry description | | | |
|--------------------|-------------------------------|--------------|-------------------|----------------|--|--|
| Index/ subindex | Name | Data Type | Access | PDO Mapping | Default value | Comment |
| 1800 | TPDO1 communication parameter | PDO_COMM_PAR | | | | It contains the communication parameters of the current PDO the device is able to transmit. |
| 1800/1 | | unsigned32 | rw | no | 0x40000180 + Node ID | COB-ID of the PDO |
| 1800/2 | | unsigned8 | rw | no | 0xFE =254 | Transmission modes: TPDO off: 0 Event-driven transmission - Only product series TP1 / TH1 / TM1: Synchronous to meas. cycle: 255 - Event Timer: 254 Synchronous: 1 ... 240 |
| 1800/3 | | unsigned16 | rw | no | 0x0000 | Inhibit Time in (multiples of) 100 µs |
| 1800/4 | | unsigned8 | rw | no | 0x00 | Compatibility entry |
| 1800/5 | | unsigned16 | rw | no | 0x0000 disabled | Event timer in ms Disabled: 0 Enabled: Range 1 ... 65535 |
| 1801 | TPDO2 communication parameter | PDO_COMM_PAR | | | | It contains the communication parameters of the current PDO the device is able to transmit. |
| 1801/1 | | unsigned32 | rw | no | 0x40000280 + Node ID | COB-ID of the PDO |
| 1801/2 | | unsigned8 | rw | no | 0x01 | Transmission mode TPDO off: 0 Synchronous: 1 ... 240 |
| 1801/3 | | unsigned16 | rw | no | 0x0000 | Inhibit Time in (multiples of) 100 µs |
| 1A00 | TPDO1 mapping parameter | PDO_MAPPING | | | | Contains the mapping for the PDOs the device is able to transmit |
| 1A00/0 | | unsigned8 | rw | no | TM1 / TF1: 0x03 TP1 / TH1: 0x03 ¹⁾ 0x02 ²⁾ | Number of entries |
| 1A00/1 | | unsigned32 | rw | no | 0x60200120 | Mapping entry 1, default: Position value channel 1 |
| 1A00/2 | | unsigned32 | rw | no | TM1 / TF1: 0x60300110 TP1 / TH1: 0x60300110 ¹⁾ 0x60200220 ²⁾ | Mapping entry 2, default: Speed value default: Speed value channel 1 ¹⁾ default: Position value channel 2 ²⁾ |
| 1A00/3 | | unsigned32 | rw | no | 0x63000108 ¹⁾ | Mapping entry 3 default: CAM Status channel 1 ¹⁾ |
| 1A00/4 | | unsigned32 | rw | no | 0x00000000 | Mapping entry 4 |
| 1A00/5 | | unsigned32 | rw | no | 0x00000000 | Mapping entry 5 |
| 1A01/0 | TPDO2 mapping parameters | unsigned8 | rw | no | TM1 / TF1: 0x03 TP1 / TH1: 0x03 ¹⁾ 0x02 ²⁾ | Number of entries |
| 1A01/1 | | unsigned32 | rw | no | 0x60200120 | Mapping entry 1, default: Position value channel 1 |
| 1A01/2 | | unsigned32 | rw | no | TM1 / TF1: 0x60300110 TP1 / TH1: 0x60300110 ¹⁾ 0x60200220 ²⁾ | Mapping entry 2, default: Speed value default: Speed value channel 1 ¹⁾ default: Position value channel 2 ²⁾ |
| 1A01/3 | | unsigned32 | rw | no | 0x63000108 ²⁾ | Mapping entry 3 default: CAM Status channel 1 ¹⁾ |
| 1A01/4 | | unsigned32 | rw | no | 0x00000000 | Mapping entry 4 |
| 1A01/5 | | unsigned32 | rw | no | 0x00000000 | Mapping entry 5 |
| 1F80 | | NMT startup | unsigned32 | rw | no | 0x00000000 |

¹⁾ for 1 position marker
²⁾ for 2 position markers

1.3.2 Device Profile Area

* for 1 position marker: default value 0x01

** for 1 position marker and product series TM1 / TF1: not available

| Object description | | | Entry description | | | |
|--------------------|--|------------|-------------------|----------------|---|---|
| Index/ subindex | Name | Data Type | Access | PDO Mapping | Default value | Comment |
| 6000 | operating parameter | unsigned16 | rw | no | 0x0000 | This object contains the functions for code sequence and commissioning diagnostic control |
| 6005 | Linear encoder measuring step settings | | | | | Object sets the number of distinguishable steps over the total measuring range (total resolution) |
| 6005/0 | | unsigned8 | ro | no | 0x02 | Number of objects |
| 6005/1 | | unsigned32 | rw | no | TP1 / TH1 / TF1: 0x00003E8 TM1: 0x000186A0 | Position measuring steps in 0.001 µm |
| 6005/2 | | unsigned32 | rw | no | 0x0000064 | Speed measuring step in 0.01 mm/s |
| 6010 | preset value | | | | | This object supports adaption of the encoder zero point to the mechanical zero point of the system |
| 6010/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels 0x01: 1 position marker, 1 channel 0x02: 2 position markers, 2 channels |
| 6010/1 | | integer32 | rw | no | 0x00000000 | Preset value channel 1 |
| 6010/2 | | integer32 | rw | no | 0x00000000** | Preset value channel 2 |
| 6020 | position value | | | | | This object defines the output position value |
| 6020/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6020/1 | | integer32 | ro | yes | 0x00000000 | Position value channel 1 |
| 6020/2 | | integer32 | ro | yes | 0x00000000** | Position value channel 2 |
| 6030 | speed value | | | | | This object defines the output speed value |
| 6030/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6030/1 | | integer16 | ro | yes | 0x0000 | Speed value channel 1 |
| 6030/2 | | integer16 | ro | yes | 0x0000** | Speed value channel 2 |
| 6300 | CAM state register | | | | | defines the status bit of the cam in a cam channel. The bit value 0 means "cam inactive". The bit value 1 means "cam active". If the polarity bit of a cam is set the actual cam state will be inverted. |
| 6300/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6300/1 | | unsigned8 | ro | yes | 0x00 | CAM state channel 1 |
| 6300/2 | | unsigned8 | ro | yes | 0x00** | CAM state channel 2 |
| 6301 | CAM enable | | | | | This object contains the calculation state for 4 cams for one position channel. If the enable bit is set to 1, the cam state will be calculated by the device. In the other case the cam state of the related cam will be set permanently to 0. |
| 6301/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6301/1 | | unsigned8 | rw | no | 0x00 | CAM enable channel 1 |
| 6301/2 | | unsigned8 | rw | no | 0x00** | CAM enable channel 2 |
| 6302 | CAM polarity register | | | | | This object contains the actual polarity settings for 4 cams for one position channel. If the polarity bit is set to 1, the cam state of an active cam will signal by setting the related cam state bit to zero. In the other case the cam state of the related cam will not be inverted. |
| 6302/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6302/1 | | unsigned8 | rw | no | 0x00 | CAM polarity channel 1 |
| 6302/2 | | unsigned8 | rw | no | 0x00** | CAM polarity channel 2 |

| Object description | | | Entry description | | | |
|--------------------|------------------|------------|-------------------|----------------|-------------------------------------|---|
| Index/ subindex | Name | Data Type | Access | PDO Mapping | Default value | Comment |
| 6310 | CAM 1 low limit | | | | | determines the lower limit of position for cam 1 |
| 6310/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6310/1 | | integer32 | rw | no | 0x00000000 | CAM 1 low limit channel 1 |
| 6310/2 | | Integer32 | rw | no | 0x00000000** | CAM 1 low limit channel 2 |
| 6311 | CAM 2 low limit | | | | | determines the lower limit of position for cam 2 |
| 6311/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6311/1 | | integer32 | rw | no | 0x00000000 | CAM 2 low limit channel 1 |
| 6311/2 | | Integer32 | rw | no | 0x00000000** | CAM 2 low limit channel 2 |
| 6312 | CAM 3 low limit | | | | | determines the lower limit of position for cam 3 |
| 6312/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6312/1 | | integer32 | rw | no | 0x00000000 | CAM 3 low limit channel 1 |
| 6312/2 | | Integer32 | rw | no | 0x00000000** | CAM 3 low limit channel 2 |
| 6313 | CAM 4 low limit | | | | | determines the lower limit of position for cam 4 |
| 6313/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6313/1 | | integer32 | rw | no | 0x00000000 | CAM 4 low limit channel 1 |
| 6313/2 | | Integer32 | rw | no | 0x00000000** | CAM 4 low limit channel 2 |
| 6320 | CAM 1 high limit | | | | | determines the higher limit of position for cam 1 |
| 6320/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6320/1 | | integer32 | rw | no | max. pos. value | CAM 1 high limit channel 1 |
| 6320/2 | | Integer32 | rw | no | max. pos. value** | CAM 1 high limit channel 2 |
| 6321 | CAM 2 high limit | integer32 | | | | determines the higher limit of position for cam 2 |
| 6321/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6321/1 | | integer32 | rw | no | max. pos. value | CAM 2 high limit channel 1 |
| 6321/2 | | Integer32 | rw | no | max. pos. value** | CAM 2 high limit channel 2 |
| 6322 | CAM 3 high limit | integer32 | | | | determines the higher limit of position for cam 3 |
| 6322/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6322/1 | | integer32 | rw | no | max. pos. value | CAM 3 high limit channel 1 |
| 6322/2 | | Integer32 | rw | no | max. pos. value** | CAM 3 high limit channel 2 |
| 6323 | CAM 4 high limit | integer32 | | | | determines the higher limit of position for cam 4 |
| 6323/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6323/1 | | integer32 | rw | no | max. pos. value | CAM 4 high limit channel 1 |
| 6323/2 | | Integer32 | rw | no | max. pos. value** | CAM 4 high limit channel 2 |
| 6330 | CAM 1 hysteresis | unsigned16 | | | | This object contains the delay setting of switch points for cam 1 |
| 6330/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6330/1 | | unsigned16 | rw | no | 0x0000 | CAM 1 hysteresis channel 1 |
| 6330/2 | | unsigned16 | rw | no | 0x0000** | CAM 1 hysteresis channel 2 |
| 6331 | CAM 2 hysteresis | unsigned16 | | | | This object contains the delay setting of switch points for cam 2 |
| 6331/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6331/1 | | unsigned16 | rw | no | 0x0000 | CAM 2 hysteresis channel 1 |
| 6331/2 | | unsigned16 | rw | no | 0x0000** | CAM 2 hysteresis channel 2 |
| 6332 | CAM 3 hysteresis | unsigned16 | | | | This object contains the delay setting of switch points for cam 3 |
| 6332/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6332/1 | | unsigned16 | rw | no | 0x0 | CAM 3 hysteresis channel 1 |
| 6332/2 | | unsigned16 | rw | no | 0x0** | CAM 3 hysteresis channel 2 |
| 6333 | CAM 4 hysteresis | unsigned16 | | | | This object contains the delay setting of switch points for cam 4 |
| 6333/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6333/1 | | unsigned16 | rw | no | 0x0000 | CAM 4 hysteresis channel 1 |
| 6333/2 | | unsigned16 | rw | no | 0x0000** | CAM 4 hysteresis channel 2 |

| Object description | | | Entry description | | | |
|--------------------|--------------------------|------------|-------------------|----------------|-------------------------------------|---|
| Index/ subindex | Name | Data Type | Access | PDO Mapping | Default value | Comment |
| 6400 | Work area state register | | | | | This object contains the actual area status of the encoder position. <u>Bit meaning</u> 0 out of range 1 range overflow 2 range underflow |
| 6400/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available work areas |
| 6400/1 | | unsigned8 | ro | yes | 0x00000000 | Work area state channel 1 |
| 6400/2 | | unsigned8 | ro | yes | 0x00000000** | Work area state channel 2 |
| 6401 | work area low limit | | | | | This object contains the lower limit of the work area |
| 6401/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available work areas |
| 6401/1 | | integer32 | rw | no | 0x00000000 | Low limit work area 1 |
| 6401/2 | | integer32 | rw | no | 0x00000000** | Low limit work area 2 |
| 6402 | work area high limit | | | | | This object contains the higher limit of the work area |
| 6402/0 | | unsigned8 | ro | no | TM1 / TF1: 0x01 TP1 / TH1: 0x02* | Number of available channels |
| 6402/1 | | integer32 | rw | no | max. pos. value | High limit work area 1 |
| 6402/2 | | integer32 | rw | no | max. pos. value** | High limit work area 2 |
| 6500 | operating status | unsigned16 | ro | no | 0x0000 | This gives information on encoder internal programmed parameters. |
| 6501 | measuring steps | unsigned32 | ro | no | 0x000003E8 | Position measuring step in 0.001 µm |

* for 1 position marker: default value 0x01

** for 1 position marker and product series TM1 / TF1: not available

1.3.3 Manufacturer specific Area

| Object description | | | Entry description | | | |
|--------------------|----------------------------|------------|-------------------|----------------|---------------------|--|
| Index/ subindex | Name | Data Type | Access | PDO Mapping | Default value | Comment |
| 2000 | node-ID | unsigned8 | rw | no | 0x7F | Node-ID of the sensor Range 1...127 |
| 2001 | CAN bit rate | unsigned16 | rw | no | 250kBaud =0x00FA | CAN bit rate of the sensor in kbit/s CAUTION: Once the "reset all" operation is performed (see chapter 1.10 <i>Non-Volatile Storage and Data Restoration</i> , even if the delivery bit rate is a different one, the bit rate will be reset to default = 250kBaud |
| 2002 | chip temperature | integer16 | ro | yes | - | Temperature from inside the µC in Celsius e.g. at ambient temperature |
| 2003 | number of position markers | unsigned8 | rw | no | 0x02 | Only product series TP1 / TH1: Number of position markers 0x01: 1 position marker, 1 channel 0x02: 2 position markers, 2 channels |
| 2004 | custom | | | | | Manufacturer defined array, not writable for the user |
| 2005 | average | unsigned8 | rw | no | 0x00 | Only product series TF1 / TM1: Average function for position and speed calculation. n = 0 ... 6: moving average across 2 ⁿ values Note: the average function influences the signal propagation delay |

1.4 Explanations to Object Library

1.4.1 Object 0x6300 Encoder Cams

Encoder cams are used to indicate if a position falls below or exceeds a defined value.

1.4.2 Cam state registers

Cam active: the current position value is between the higher and lower cam-limit
 Cam inactive: the current position value is not between the higher and lower cam-limit.

The values for low limit (0x631x) and high limit (0x632x) regard the values for preset (0x6010) and measuring steps (0x6005). The value of hysteresis (0x633x) is added in direction of motion.

Note: the cam high limit value can have a lower value than the cam low limit

A change in cam state causes an EMCY message.
 The cam state objects (0x6300) are able to be mapped to the TPDOs.

1.4.3 Object 0x6400 Work Area

It is possible for encoders to define a so-called user defined working area.
 The main purpose for a work area is to get a high-priority information (via EMCY message) when the transducer's position leaves its predefined working area.
 The actual work area information with work area low limit and work area high limit may be stored in object 0x6401 and 0x6402. This way, the area state object (0x6400) may also be used as software limit switches.

1.4.3.1 Work Area Supervision

Each work area channel is fixedly linked to a position channel:

| Position | | Work Area |
|----------|----------|-----------|
| Index | Subindex | Channel |
| 0x6020 | 1 | 1 |
| 0x6020 | 2 | 2 |

1.4.3.2 Work Area State

| Condition | State register 0x6400 |
|---|------------------------|
| Position < Work Area Low Limit Position >= Work Area Low Limit | Bit 2 = 1 Bit 2 = 0 |
| Position > Work Area High Limit Position <= Work Area High Limit | Bit 1 = 1 Bit 1 = 0 |
| Position <= Preset Value or Position >= Sensor length otherwise | Bit 0 = 1 Bit 0 = 0 |

The values for low limit (0x6401) and high limit (0x6402) regard the values for preset (0x6010) and scaling (0x6501, 0x6502).

A change in work area state causes an EMCY message.
 The work area state objects (0x6400) are able to be mapped to the TPDOs.

1.5 LSS / Layer Setting Service

To configure the encoder via the LSS (according CiA DS 305) the encoder is handled as a slave, the PLC must have a LSS master functionality.

A LSS-message is composed as follows:

| | | | | | | | | | |
|--------|-----|---------|-------|-------|-------|-------|-------|-------|-------|
| COB-ID | DLC | Command | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 |
|--------|-----|---------|-------|-------|-------|-------|-------|-------|-------|

This applies to the COB-ID:

- LSS-Master ⇒ LSS-Slave: 2021 (0x7E5)
- LSS-Slave ⇒ LSS-Master: 2020 (0x7E4)

LSS can only be used when the encoder is in the stopped status or pre-operational status.
 The NMT command for setting the encoder in stopped status is:

| COB-ID | DLC | Command | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 |
|--------|-----|---------|-------|-------|-------|-------|-------|-------|-------|
| 0x7E5 | 8 | 0x04 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |

To program via LSS the sensor has to be switched to LSS configuration state.
 There are two possible ways to do so:

• **Switch Mode Selective:**

only the addressed CANopen device is switched to the LSS configuration state

LSS requires data content in the following objects:

Example:

| | | | |
|--------------|--------------------|------------|------------------|
| Vendor-ID | (see index 1018/1) | 0x0182 | LSS-Command 0x40 |
| Product code | (see index 1018/2) | 0x0BE0 | LSS-Command 0x41 |
| Rev.No. | (see index 1018/3) | 0x10003 | LSS-Command 0x42 |
| Serial-No. | (see index 1018/4) | 0x12345678 | LSS-Command 0x43 |

After receiving the identification objects, the encoder answers with LSS-Command **0x44**.

| COB-ID | DLC | Rx/Tx | Command | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 |
|--------|-----|-------|---------|-------|-------|-------|-------|-------|-------|-------|
| 0x7E5 | 8 | Rx | 0x40 | 0x82 | 0x01 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 0x7E5 | 8 | Rx | 0x41 | 0xE0 | 0x0B | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 0x7E5 | 8 | Rx | 0x42 | 0x03 | 0x00 | 0x01 | 0x00 | 0x00 | 0x00 | 0x00 |
| 0x7E5 | 8 | Rx | 0x43 | 0x78 | 0x56 | 0x34 | 0x12 | 0x00 | 0x00 | 0x00 |
| 0x7E4 | 8 | Tx | 0x44 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |

- **Switch Mode Global:** all CANopen devices supporting LSS are switched to the LSS configuration state

| COB-ID | DLC | Rx/Tx | Command | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 |
|--------|-----|-------|---------|-------|-------|-------|-------|-------|-------|-------|
| 0x7E5 | 8 | Rx | 0x04 | 0x01 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |

When the CAN devices are in configuration state the Node-ID and/or the bit rate can be changed.

1.5.1 Configuration of Node-ID

The Node-ID can be programmed with the LSS-Command **0x11**

N ID: new Node-ID in the range of 1...127

Err Code: 0: protocol successfully completed / 1: Node-ID out of range

| COB-ID | DLC | Rx/Tx | Command | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 |
|--------|-----|-------|---------|----------|-------|-------|-------|-------|-------|-------|
| 0x7E5 | 8 | Rx | 0x11 | N ID | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 0x7E4 | 8 | Tx | 0x11 | Err Code | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |

Change of Node-ID will cause:

- Automatic alteration of COB-ID's for SDO1, EMCY and Heartbeat and TPDOs.
- Non-volatile Node-ID storage through „Store Configuration“ in the LSS mode configuration.

1.5.2 Configuration of Bit Rate

The Bit Rate can be programmed with LSS-Command **0x13**

| | |
|--------------|------------------|
| Table Index: | 0x07: 20 kBaud |
| | 0x06: 50 kBaud |
| | 0x04: 125 kBaud |
| | 0x03: 250 kBaud |
| | 0x02: 500 kBaud |
| | 0x01: 800 kBaud |
| | 0x00: 1000 kBaud |

Err Code: 0: protocol successfully completed 1: Bit timing not supported

| COB-ID | DLC | Rx/ Tx | Command | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 |
|--------|-----|-----------|---------|----------------|-------|-------|-------|-------|-------|-------|
| 0x7E5 | 8 | Rx | 0x13 | Table Index | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 0x7E4 | 8 | Tx | 0x13 | Err Code | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |

Change of Bit rate will cause:

- The bit rate gets active
- Non-volatile CAN bit rate storage through „Store Configuration“ in the LSS mode configuration

1.5.3 Store Configuration Data

The LSS configuration data (Node-ID and Bit Rate) are stored to the non-volatile memory of the sensor using LSS-Command **0x17**

Err Code: 0: protocol successfully completed 2: storage media access error

| COB-ID | DLC | Rx/ Tx | Command | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 |
|--------|-----|-----------|---------|----------|-------|-------|-------|-------|-------|-------|
| 0x7E5 | 8 | Rx | 0x17 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 0x7E4 | 8 | Tx | 0x17 | Err Code | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |

1.6 SDO Services

Service Data Objects SDO (according to CiA DS 301) manage the parameter data exchange, e.g. the non-cyclical execution of the preset function.

Parameters of device object library (object index/subindex see chapter 1.3 *Object Library*) can be read, written or stored by means of SDO.

1.6.1 SDO Download

The SDO download service is used to configure the parameters.

Command 0x2_: 0x22 write command, parameter to encoder
 0x23 write command, 4 Byte parameter to encoder
 0x27 write command, 3 Byte parameter to encoder
 0x2B write command, 2 Byte parameter to encoder
 0x2F write command, 1 Byte parameter to encoder

Command 0x60: confirmation: parameter received

| COB-ID | DLC | Rx/ Tx | Command | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 |
|---------------|-----|-----------|---------|-------|-------|---------------|-------------|-------|-------|-------------|
| 0x600+Node-ID | 8 | Rx | 0x2_ | Index | | Sub- index | Data LSB | Data | Data | Data MSB |
| 0x580+Node-ID | 8 | Tx | 0x60 | Index | | Sub- index | 0x00 | 0x00 | 0x00 | 0x00 |

Example: object index 0x1010 subindex 01 "store all parameters"

| | | | | | | | | | | |
|---------------|---|----|------|------|------|------|------|------|------|------|
| 0x600+Node-ID | 8 | Rx | 0x23 | 0x10 | 0x10 | 0x01 | 0x73 | 0x61 | 0x76 | 0x65 |
| 0x580+Node-ID | 8 | Tx | 0x60 | 0x10 | 0x10 | 0x01 | 0x00 | 0x00 | 0x00 | 0x00 |

Example: object index 0x1011 subindex 01 "restore all parameters"

| | | | | | | | | | | |
|---------------|---|----|------|------|------|------|------|------|------|------|
| 0x600+Node-ID | 8 | Rx | 0x23 | 0x11 | 0x10 | 0x01 | 0x6C | 0x6F | 0x61 | 0x64 |
| 0x580+Node-ID | 8 | Tx | 0x60 | 0x11 | 0x10 | 0x01 | 0x00 | 0x00 | 0x00 | 0x00 |

Example: object index 0x2000 subindex 00 "set new node-ID" with 64

| | | | | | | | | | | |
|---------------|---|----|------|------|------|------|------|------|------|------|
| 0x600+Node-ID | 8 | Rx | 0x2F | 0x00 | 0x20 | 0x00 | 0x40 | 0x00 | 0x00 | 0x00 |
| 0x580+Node-ID | 8 | Tx | 0x60 | 0x00 | 0x20 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |

NODE-ID

Using writing to object 0x2000, non-volatile storage has to be done by writing the "save"- signature (0x65766173) on object 0x1010/1 (TP1/TH1) or 0x1010/4 (TF1). These changes will become effective after a communication restart or a power up.

Changing the Node-ID will affect all COB-IDs according to the "predefined connection set".

Example: COB-ID TPDO1 = 0x180 + (Node-ID)

BIT-RATE

Using writing to object 0x2001; non-volatile storage has to be done by writing the "save"- signature (0x65766173) on object 0x1010/1 (TP1/TH1) or 0x1010/4 (TF1). These changes will become effective after a communication restart or a power up.

1.6.2 SDO Upload

The SDO upload service is used to read the parameters.

Command 0x40: read command, parameters from encoder
 Command 0x4_: 0x42 read command, parameter to encoder
 0x43 read command, 4 Byte parameter to encoder
 0x47 read command, 3 Byte parameter to encoder
 0x4B read command, 2 Byte parameter to encoder
 0x4F read command, 1 Byte parameter to encoder

| COB-ID | DLC | Rx/ Tx | Command | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 |
|---------------|-----|-----------|---------|-------|-------|-----------|----------|-------|-------|----------|
| 0x600+Node-ID | 8 | Rx | 0x40 | Index | | Sub-index | 0x00 | 0x00 | 0x00 | 0x00 |
| 0x580+Node-ID | 8 | Tx | 0x4_ | Index | | Sub-index | Data LSB | Data | Data | Data MSB |

1.6.3 SDO Abort

If the SDO download or SDO upload service fails for any reason, the sensor responds with a SDO abort protocol.

Abort Code: 0x06090011 subindex does not exist
 0x06090030 value exceeded
 0x06020000 object does not exist
 0x06010001 object is write only
 0x06010002 object is read only
 0x06060000 access error
 0x08000020 data transport error
 0x08000000 general error
 0x08000022 wrong state

| COB-ID | DLC | Rx/ Tx | Command | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 |
|---------------|-----|-----------|---------|-------|-------|-----------|------------|-------|-------|-------|
| 0x580+Node-ID | 8 | Tx | 0x80 | Index | | Sub-index | Abort code | | | |

1.7 Process Data PDO

Process Data Objects (according CiA DS 301) manage the process data exchange, f.e the cyclical transmission of the position value. The process data exchange with the CANopen PDOs is a very slim process without protocol overhead.

1.7.1 PDO Default Setting

2 Transmit PDOs (TPDO) with each max. 8 bytes are provided:

0x1800 TPDO1: default: Event-driven with event timer switched off (changeable to synchronous)
 0x1801 TPDO2: default: synchronous

1.7.2 PDO Parameter Setting

The contents of the encoder-specific TPDOs can be configured by variable mapping according to customer's requirements. This mapping has to be performed for the encoder as well as for the receiver. The PDO is limited to a maximum size of 8 bytes and 5 mappings per each PDO.

| Mappable objects | | |
|------------------|----------------------|------|
| Index/Subindex | Entry | Byte |
| 0x6020/1 | Position value ch. 1 | 4 |
| 0x6020/2 | Position value ch. 2 | 4 |
| 0x6030/1 | Speed value ch. 1 | 2 |
| 0x6030/2 | Speed value ch. 2 | 2 |
| 0x6300/1 | Cam state ch. 1 | 1 |
| 0x6300/2 | Cam state ch. 2 | 1 |
| 0x6400/1 | Work area ch. 1 | 1 |
| 0x6400/2 | Work area ch. 2 | 1 |
| 0x2002 | Chip temperature | 1 |

Step 1: For mapping of further objects, the PDO must be completely disabled and the MSB of PDO COB-ID have to be set to 1.

| PDO | Object | COB-ID for active PDO | COB-ID for disabled PDO (MSB set to 1) |
|-----|--------|-----------------------|--|
| 1 | 0x1800 | 0x40000xxx | 0xC0000xxx |
| 2 | 0x1801 | 0x40000xxx | 0xC0000xxx |

Step 2: Clearing entries in mapping table of PDO1 => subindex 0x0 of object 1A00 has to be set to 0x00.

Step 3: Parameter setting of selected mappings

Example:

A PDO can be mapped in the way that the "current position", the "current speed" and the "current chip temperature" are transmitted in one information without producing more than necessary bus load.

Mapping #1 "current position":

object 0x1A00/1 size: 32 bits = 4 byte => 0x20 position value = object 0x6020/1

| COB-ID | DLC | Rx/ Tx | Command | destination object | | | size | source object | | | |
|---------------|-----|-----------|---------|--------------------|-------------------|---------------------|-------|---------------------|-------------------|-------------------|--|
| | | | | Byte0 (object) | Byte1 (object) | Byte2 (subindex) | Byte3 | Byte4 (subindex) | Byte5 (object) | Byte6 (object) | |
| 0x600+Node-ID | 8 | Rx | 0x23 | 0x00 | 0x1A | 0x01 | 0x20 | 0x01 | 0x20 | 0x60 | |

Mapping #2 "current speed":

object 0x1A00/2 size: 16 bits = 2 byte => 0x10 speed value = object 0x6030/1

| COB-ID | DLC | Rx/ Tx | Command | destination object | | | size | source object | | | |
|---------------|-----|-----------|---------|--------------------|-------------------|---------------------|-------|---------------------|-------------------|-------------------|--|
| | | | | Byte0 (object) | Byte1 (object) | Byte2 (subindex) | Byte3 | Byte4 (subindex) | Byte5 (object) | Byte6 (object) | |
| 0x600+Node-ID | 8 | Rx | 0x23 | 0x00 | 0x1A | 0x02 | 0x10 | 0x01 | 0x30 | 0x60 | |

Mapping #3: "current chip temperature".

object 0x1A00/3 size: 8 bits = 1 byte => 0x08 temperature value = object 0x2002

| COB-ID | DLC | Rx/ Tx | Command | destination object | | | size | source object | | | |
|---------------|-----|-----------|---------|--------------------|-------------------|---------------------|-------|---------------------|-------------------|-------------------|--|
| | | | | Byte0 (object) | Byte1 (object) | Byte2 (subindex) | Byte3 | Byte4 (subindex) | Byte5 (object) | Byte6 (object) | |
| 0x600+Node-ID | 8 | Rx | 0x23 | 0x00 | 0x1A | 0x03 | 0x08 | 0x00 | 0x02 | 0x20 | |

Step 4: Setting entries in mapping table => subindex 0x0 of object 1A00 has to be set to the numbers of mapping entries (e.g. 0x03)

Step 5: For activating the PDO the MSB of PDO COB-ID have to be set to 0.

| PDO | Object | COB-ID for disabled PDO | COB-ID for enabled PDO (MSB cleared) |
|-----|--------|-------------------------|--------------------------------------|
| 1 | 0x1800 | 0xC0000xxx | 0x40000xxx |
| 2 | 0x1801 | 0xC0000xxx | 0x40000xxx |

Note:

TPDO1 value for Event Timer must always be higher than the value for Inhibit Time (except for value 0).

Failed sending of TPDOs can occur if:

- more TPDOs shall be sent than the CANbus may accept due to insufficient CAN bit rate compared to TPDO/Event Timer
- excessive bus load or unfavourable setting of COB-ID in the CANopen network prevents TPDO sending
- Object 0x1800/5- event timer- is set to 0.

1.8 Error Handling

Depending on the type of error occurred, the sensor will react accordingly:

| Error Class | Error | Error Message from Sensor |
|---------------------|--|--|
| Protocol error | SDO protocol error, corrupted data received via SDO | Abort SDO Transfer* |
| | PDO protocol error, corrupted data received via PDO | Not relevant, sensor does support TPDOs only |
| Communication error | CAN bus off CAN error passive CAN overrun CAN buffer overflow | EMCY message** |
| Process error | Position or sensor error | EMCY message** Position value is set to 0x7FFFFFFF (beyond measuring range), speed value is set to 0x0000 |
| | Data error | Abort SDO Transfer* or EMCY message** |
| Change of state | According to Cams and Work Areas | EMCY Message** |

* according to DS-301, see chapter 1.6 SDO Services

** details see chapter 1.8.1 Emergency Messages

1.8.1 Emergency Messages

| COB-ID EMCY | DLC | Rx/ Tx | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 |
|--------------|-----|-----------|----------------|-------|-------|-------|-------|-------|-------|-------|
| 0x80+Node-ID | 8 | Tx | See next table | | | | | | | |

COB-ID EMCY in object 0x1014.
 Error-Register in object 0x1001.

0x50xx Device Hardware
 0x80xx Monitoring

0x60xx Device Software
 0x90xx External Error

| Error-Code | Error-Register | Additional Byte | | | Description |
|------------|----------------|-----------------|--------|----------|---|
| | | Byte 0,1 | Byte 2 | Byte 3,4 | |
| 0x0000 | 0 | 0 | 0 | 0 | Sensor Error resetted, no Error |
| 0x5000 | 1 | 1 | 1 | 0 | Sensor Error <u>Cause:</u> An internal error bit is set. The current process values stay in the object directory; the cam and work area states stay unchanged. If the internal error bit is reset to 0, an EMCY with 0x0000 is sent. |
| 0x5000 | 1 | 1 | 5 | 0 | Unexpected System Data <u>Cause:</u> application received sensor-internal system data instead of process data during normal operation mode. <u>Reaction:</u> sensor changes into PRE-OPERATIONAL mode. System data is being computed as if power-up had occurred (system data are being checked and serial number and product code, if correct, is being stored in EEPROM). If system data are correct, normal operation can continue afterwards. |
| 0x5000 | 1 | 1 | 6 | 0 | No System Data <u>Cause:</u> Sensor configuration is not detected internally. <u>Reaction:</u> sensor changes into PRE-OPERATIONAL mode. NMT Master cannot set the sensor into OPERATIONAL mode, until error has been eliminated and sensor has been reset through power off/on. Attempted setting into OPERATIONAL will result in repeated EMCY message sending. |

| Error-Code Byte 0,1 | Error-Register Byte 2 | Additional Byte | | | Description |
|------------------------|--------------------------|-----------------|----------|--------|---|
| | | Byte 3,4 | Byte 5,6 | Byte 7 | |
| 0x5000 | 1 | 2 | 1 | 0 | <i>Invalid Serial Number in EEPROM</i> <u>Cause:</u> the serial number and product code stored in the FLASH is invalid. <u>Reaction:</u> sensor changes into PRE-OPERATIONAL mode. NMT Master cannot set the sensor into OPERATIONAL mode, until error has been eliminated and sensor has been reset through power off/on. Attempted setting into OPERATIONAL will result in repeated EMCY message sending. |
| 0x6000 | 1 | 1 | 0 | 0 | <i>Insufficient Event Timer Value</i> <u>Cause:</u> CAN bit rate has been set to a low value and an insufficient Event Timer value has been set for TPDO1 accordingly. <u>Reaction:</u> TPDO1 sending will occur partially. |
| 0x8110 | 1 | 1 | 0 | 0 | <i>CAN Controller Overflow , CAN Overrun</i> <u>Cause:</u> data buffers of CAN controller are still holding data and cannot accept new entries. Data is being lost. <u>Reaction:</u> none |
| 0x8110 | 1 | 2 | 0 | 0 | <i>CAN Buffer Overflow</i> <u>Cause:</u> data buffers of CANopen library are still holding data and cannot accept new entries. Data is being lost. <u>Reaction:</u> none |
| 0x8120 | 1 | 0 | 0 | 0 | <i>CAN Error Passive</i> <u>Cause:</u> CAN controller has detected communication errors and is reporting error passive. <u>Reaction:</u> none |
| 0x8140 | 1 | 0 | 0 | 0 | <i>CAN Recovered From Bus-Off</i> <u>Cause:</u> CAN controller registered too many sending errors. CAN communication could be restored afterwards. <u>Reaction:</u> none |
| 0x9080 | 1 | ch. | cam | state | <i>Encoder CAM</i> <u>Cause:</u> the state of cam / channel has changed. State is coded according to 0x6300. <u>Reaction:</u> none |
| 0x9090 | 1 | ch. | 0 | state | <i>Work Areas</i> <u>Cause:</u> the state of Work Area / channel has changed. State is coded according to 0x6400. <u>Reaction:</u> none |

1.9 Error Objects

1.9.1 Manufacturer-specific Status

The object 0x1002 shows the sensor status bit code and is used for internal process control purposes. For servicing this information can be requested via SDO (see chapter 1.6 SDO Services).

| Bit | Definition (if bit value = 1) |
|-----|---|
| 17 | Lifeguarding Error |
| 16 | sensor receiving process data |
| 15 | TPDO1 Event Timer Value insufficient for set CAN bit rate |
| 6 | CAN Controller Overflow |
| 5 | CAN Buffer Overflow |
| 4 | CAN Error Passive |
| 3 | CAN Bus-Off |

| Bit | Definition (if bit value = 1) |
|-----|--|
| 2 | CAN Bus-Off Timer started |
| 0-1 | NMT Condition of Sensor %11 stopped %10 operational %01 pro-operational %00 initialisation |

1.10 Non-Volatile Storage and Data Restoration

Default values for all data objects are stored in the non-volatile program memory.
 Data encryption to the non-volatile memory is only admitted in the pre-operational status.

• **Storage via LSS:**

Data must be stored through the LSS Service Configuration/Store while in LSS Configuration Mode (see chapter 1.5 *LSS / Layer Setting Service*)

• **Storage via SDO:**

Object 0x1010:

Data is stored in the non-volatile memory during encryption of object 0x1010 with „save“ signature (0x65766173).

Object 0x1011:

Encryption of object 0x1011 with the signature „load“ (0x64616663) will upload data from the non-volatile memory.
 Default settings are being restored (see chapter 1.6 *SDO Services*).



CAUTION: In case of custom programmed parameters like node-ID, averaging, bit rate etc. these will be reset to default in case of the corresponding reset command below (default values see chapter 1.3 *Object Library*).

| Object 0x1010 Object 0x1011 | Subindex /1 All | Subindex /2 Communication | Subindex /3 Application | Subindex /4 Manufacturer |
|---|--------------------|------------------------------|----------------------------|-----------------------------|
| COB-ID Sync | | | | |
| Guard Time | X | X | | |
| Life Time Factor | X | X | | |
| Heartbeat Timer | X | X | | |
| TPDO COB-ID | D | X | | |
| TPDO Trans Typ | X | X | | |
| TPDA Inhibit Time | X | X | | |
| TPDO Event Timer | X | X | | |
| TPDO Mapping | X | X | | |
| NMT Startup | X | X | | |
| Node-ID | X (TP1/TH1) | | | X (TF1) |
| BitRate | X (TP1/TH1) | | | X (TF1) |
| Number of position markers (only TP1 / TH1) | X | | | |
| Custom (only TP1 / TH1) | | | | X |
| Operating Parameters | X | | X | |
| Linear Encoder Measuring Step Settings | X | | X | |
| Preset Value | X | | X | |
| CAM Enable | X | | X | |
| CAM Polarity | X | | X | |
| CAM Low Limit | X | | X | |
| CAM High Limit | X | | X | |
| CAM Hysterese | | | X | |
| Work Area Low Limit | | | X | |
| Work Area High Limit | | | X | |

X: data saved or restored
 D: data set to default value

• **Delete via SDO Object 0x1010:**

Additionally to the functionality defined in CiA standard DS-301, CANopen library offers the possibility to delete data in the non-volatile memory. Delete process is initiated by sending the signature "kill" (0x6C6C696B) to object 0x1010.

• **Manufacturing Mode Object 0x1010**

- Only TM1 series:



If the sensor is out of function and the signature "boot" 0x746F6F62 in object 0x1000 (device type) is active, the sensor is in manufacturing mode. This mode can be left by power off-on or via the operational command.

1.11 Abbreviations

| | |
|--------|--|
| CAN | Controller Area Network |
| ch | channel |
| COB-ID | Communication Object Identifier |
| const | constant parameter, only readable |
| DLC | Data Length Code |
| DS | Draft Standard |
| EMCY | Emergency Service |
| NMT | Network-Management |
| PDO | Process Data Object |
| Pos | Position (value) |
| ro | read only, parameter can change |
| rw | read/write |
| Rx | Novotechnik sensor is consumer of the CAN data frame |
| RTR | Remote Transmission Request |
| SDO | Service Data Object |
| SYNC | Synchronisation message |
| TPDO | Transmit Process Data Object |
| Tx | Novotechnik sensor is producer of the CAN data frame |

1.12 Document Changes

| Revision | Changes | Date | Who |
|----------|---|----------|-------|
| V00 | First edition | 16.07.14 | VM/mm |
| V07 | 1.2.5 / 1.3.1 object 1801x2: event driven transmission deleted for TPDO2. 1.3.1 objects: TPDO1 and TPDO2: name modified. 1.3. object 1800/2 and 1801/2 comment: synchronous 1...240 instead 1...239, TPDO off: 0 added | 01.04.20 | VM/mm |
| V08 | 1.3.1 object 1010/4 user parameter data instead of manufacturer defined parameters. 1.3.1 1010/5 added (Manufacturer data parameter). 1.10. signature kill 6C6C696B instead of 6B696C6C, comment regarding manufacturing mode added | 21.01.21 | VM/mm |