

Handbuch

Absolute Drehgeber mit **CANopen** (Bushaube und integrierte Schnittstelle)

Ab Revisionsnummer 1.04

Inhalt

| | Seite |
|------------------------------------------------------------------------------|-----------|
| 1. Einleitung | 3 |
| 1.1. Produktzuordnung | 3 |
| 2. Sicherheits- und Betriebshinweise | 4 |
| 3. Produktfamilien | 5 |
| 4. CAN-Bus und CANopen-Kommunikation | 6 |
| 4.1. CAN-Bus | 6 |
| 4.1.1. CAN-Bus-Eigenschaften | 6 |
| 4.2. CANopen | 7 |
| 4.3. CANopen-Kommunikation | 8 |
| 4.3.1. Kommunikationsprofil | 8 |
| 4.3.2. CANopen Meldungsaufbau | 8 |
| 4.3.3. Servicedaten-Kommunikation | 9 |
| 4.3.4. Prozessdaten-Kommunikation | 10 |
| 4.3.5. Emergency-Dienst | 12 |
| 4.3.6. Netzwerkmanagement-Dienste | 13 |
| 4.3.7. Layer Setting Services | 17 |
| 4.4. Drehgeber Profil | 20 |
| 4.4.1. Drehgeber-Objekte Übersicht | 20 |
| 5. Diagnose und Wissenswertes | 24 |
| 5.1. Fehlerdiagnose Feldbus-Kommunikation | 24 |
| 5.2. Fehlerdiagnose über Feldbus | 24 |
| 5.3. Wissenswertes zum Sensor | 25 |
| 6. Applikationen | 26 |
| 6.1. Objekte setzen und lesen | 26 |
| 6.2. Konfiguration | 27 |
| 6.3. Betrieb | 28 |
| 7. Anschlussbelegung und Inbetriebnahme der Bushaube | 30 |
| 7.1. Mechanischer Anbau | 30 |
| 7.2. Elektrischer Anschluss | 30 |
| 7.2.1. Teilnehmeradresse (Node ID) einstellen | 30 |
| 7.2.2. Baudrate einstellen | 31 |
| 7.2.3. Abschlusswiderstand | 31 |
| 7.2.4. Anschluss Bushaube | 31 |
| 7.2.5. Anschlussbelegung der Klemmen | 34 |
| 7.3. Anzeigeelemente | 34 |
| 8. Anschluss und Inbetriebnahme bei integrierter Feldbusschnittstelle | 35 |
| 8.1.1. Teilnehmeradresse und Baudrate einstellen | 35 |
| 8.1.2. Abschlusswiderstand | 36 |
| 8.1.3. Anschlussbelegung D-SUB-Stecker | 36 |
| 8.1.4. Anschlussbelegung Stecker M12 | 36 |

1. Einleitung

1.1. Produktzuordnung Wellen-Drehgeber

| Produkt | Produkt-Code | Device Name | EDS-Datei | Produktfamilie | Variante |
|---------|--------------|-------------|--------------|--------------------------|------------|
| BPSV 58 | 0x0B | BPSx | BPSxBusC.eds | ProCoder – Singleturn | Bushaube |
| BPMV 58 | 0x0A | BPMx | BPMxBusC.eds | ProCoder – Multiturn | Bushaube |
| BEMV 58 | 0x0A | BPMx | BPMxBusC.eds | ProCoder – Multiturn | Bushaube |
| BMSV 42 | 0x17 | BMSx | BMSxInt.eds | MAGRES – Singleturn | integriert |
| BMMV 42 | 0x16 | BMMx | BMMxInt.eds | MAGRES – Multiturn | integriert |
| BOSV 58 | 0x0F | BOSx | BOSxBusC.eds | Digitalizer – Singleturn | Bushaube |
| BOMV 58 | 0x0E | BOMx | BOMxBusC.eds | Digitalizer – Multiturn | Bushaube |

Endwellen-Drehgeber

| Produkt | Produkt-Code | Device Name | EDS-Datei | Produktfamilie | Variante |
|---------|--------------|-------------|--------------|--------------------------|------------|
| BPSH 58 | 0x0B | BPSx | BPSxBusC.eds | ProCoder – Singleturn | Bushaube |
| BPMH 58 | 0x0A | BPMx | BPMxBusC.eds | ProCoder – Multiturn | Bushaube |
| BMSH 58 | 0x0D | BMSx | BMSxBusC.eds | MAGRES – Singleturn | Bushaube |
| BMMH 58 | 0x0C | BMMx | BMMxBusC.eds | MAGRES – Multiturn | Bushaube |
| BMSH xx | 0x17 | BMSx | BMSxInt.eds | MAGRES – Singleturn | Integriert |
| BMMH xx | 0x16 | BMMx | BMMxInt.eds | MAGRES – Multiturn | Integriert |
| BOSH 58 | 0x0F | BOSx | BOSxBusC.eds | Digitalizer – Singleturn | Bushaube |
| BOMH 58 | 0x0E | BOMx | BOMxBusC.eds | Digitalizer – Multiturn | Bushaube |
| BOSH 58 | 0x19 | BOSx | BOSxInt.eds | Digitalizer - Singleturn | Integriert |
| BOMH 58 | 0x18 | BOMx | BOMxInt.eds | Digitalizer - Multiturn | Integriert |
| BFF | 0x02 | BFx | BFx13Bit.eds | BFF | Integriert |
| BFG | 0x02 | BFx | BFx13Bit.eds | BFG | Integriert |

Hohlwellen-Drehgeber

| Produkt | Produkt-Code | Device Name | EDS-Datei | Produktfamilie | Variante |
|---------|--------------|-------------|--------------|-----------------------|----------|
| BISD 58 | 0x0B | BPSx | BPSxBusC.eds | ProCoder – Singleturn | Bushaube |
| BIMD 58 | 0x0A | BPMx | BPMxBusC.eds | ProCoder – Multiturn | Bushaube |
| BPSD 58 | 0x0B | BPSx | BPSxBusC.eds | ProCoder – Singleturn | Bushaube |
| BPMD 58 | 0x0A | BPMx | BPMxBusC.eds | ProCoder – Multiturn | Bushaube |
| BPSD 14 | 0x0B | BPSx | BPSxBusC.eds | ProCoder – Singleturn | Bushaube |
| BPMD 14 | 0x0A | BPMx | BPMxBusC.eds | ProCoder – Multiturn | Bushaube |
| BPSD 25 | 0x0B | BPSx | BPSxBusC.eds | ProCoder – Singleturn | Bushaube |
| BPMD 25 | 0x0A | BPMx | BPMxBusC.eds | ProCoder – Multiturn | Bushaube |
| BPSD 50 | 0x0B | BPSx | BPSxBusC.eds | ProCoder – Singleturn | Bushaube |
| BPMD 50 | 0x0A | BPMx | BPMxBusC.eds | ProCoder – Multiturn | Bushaube |

2. Sicherheits- und Betriebshinweise

Zusätzliche Informationen

- Das Handbuch ist eine Ergänzung zu bereits vorhandenen Dokumentationen (Kataloge, Datenblätter und Montageanleitungen).
- Die Anleitung muss unbedingt vor Inbetriebnahme gelesen werden.

Bestimmungsgemässer Gebrauch

- Der Drehgeber ist ein Präzisionsmessgerät. Er dient ausschliesslich zur Erfassung von Winkelpositionen und Umdrehungen, der Aufbereitung und Bereitstellung der Messwerte als elektrische Ausgangssignale für das Folgegerät. Der Drehgeber darf ausschliesslich zu diesem Zweck verwendet werden.

Inbetriebnahme

- Einbau und Montage des Drehgebers darf ausschliesslich durch eine Fachkraft erfolgen.
- Betriebsanleitung des Maschinenherstellers beachten.

Sicherheitshinweise

- Vor Inbetriebnahme der Anlage alle elektrischen Verbindungen überprüfen.
- Wenn Montage, elektrischer Anschluss oder sonstige Arbeiten am Drehgeber und an der Anlage nicht fachgerecht ausgeführt werden, kann es zu Fehlfunktion oder Ausfall des Drehgebers führen.
- Eine Gefährdung von Personen, eine Beschädigung der Anlage und eine Beschädigung von Betriebseinrichtungen durch den Ausfall oder Fehlfunktion des Drehgebers muss durch geeignete Sicherheitsmassnahmen ausgeschlossen werden.
- Drehgeber darf nicht ausserhalb der Grenzwerte betrieben werden (siehe weitere Dokumentationen).

Bei Nichtbeachtung der Sicherheitshinweise kann es zu Fehlfunktionen, Sach- und Personenschäden kommen!

Transport und Lagerung

- Transport und Lagerung ausschliesslich in Originalverpackung.
- Drehgeber nicht fallen lassen oder grösseren Erschütterungen aussetzen.

Montage

- Schläge oder Schocks auf Gehäuse und Welle / Hohlwelle vermeiden.
- Gehäuse nicht verspannen.
- Keine starre Verbindung von Drehgeberwelle und Antriebswelle vornehmen.
- Drehgeber nicht öffnen oder mechanisch verändern.

Welle, Kugellager, Glasscheibe oder elektronische Teile können beschädigt werden. Die sichere Funktion ist dann nicht mehr gewährleistet.

Elektrische Inbetriebnahme

- Drehgeber elektrisch nicht verändern.
- Keine Verdrahtungsarbeiten unter Spannung vornehmen.
- Der elektrische Anschluss darf unter Spannung nicht aufgesteckt oder abgenommen werden.
- Die gesamte Anlage EMV gerecht installieren. Einbaumgebung und Verkabelung beeinflussen die EMV des Drehgebers. Drehgeber und Zuleitungen räumlich getrennt oder in grossem Abstand zu Leitungen mit hohem Störpegel (Frequenzumrichter, Schütze usw.) verlegen.
- Bei Verbrauchern mit hohen Störpegeln separate Spannungsversorgung für den Drehgeber bereitstellen.
- Drehgebergehäuse und die Anschlusskabel vollständig schirmen.
- Drehgeber an Schutzerde (PE) anschliessen. Geschirmte Kabel verwenden. Schirmgeflecht muss mit der Kabelverschraubung oder Stecker verbunden sein. Anzustreben ist ein beidseitiger Anschluss an Schutzerde (PE), Gehäuse über den mechanischen Anbau, Kabelschirm über die nachfolgenden angeschlossenen Geräte. Bei Problemen mit Erdschleifen mindestens eine einseitige Erdung.

Bei Nichtbeachtung kann es zu Fehlfunktionen, Sach- und Personenschäden kommen!

3. Produktfamilien

Die Produktfamilie ist modular aufgebaut. Basis-Drehgeber und Bushauben können je nach Anforderungen an den Drehgeber und dem gewählten Bussystem beliebig kombiniert werden. Die Basis-Drehgeber unterscheiden sich in der Genauigkeit, den Umgebungsbedingungen und dem Abtastsystem. Bei der MAGRES Produktfamilie gibt es zusätzlich noch die sogenannte integrierte Version mit Steckeranschluss ohne Bushaube.

Bushaube

In der Bushaube ist die gesamte Elektronik der Messwertaufbereitung und des Feldbusses integriert. Die Kommunikation mit dem CAN Bus erfolgt über den im Mikrocontroller integrierten CAN Controller. Der verwendete CAN Controller ist Full CAN tauglich und unterstützt die CAN Spezifikation 2.0B. Die Busan Kopplung ist genormt nach ISO/DIS 11898. Die maximale Datenrate ist 1 MBit/s.

MAGRES

Er hat mit 13 Bit eine Auflösung von 8192 Schritten/Umdrehung, enthält ein magnetisches Abtastsystem und ist für extreme Umwelteinflüsse geeignet. Die Produktfamilie MAGRES besteht aus zwei Varianten, zum einen die sogenannte integrierte Version mit Stecker und Buchsenanschluss ohne Bushaube und zum anderen das modular aufgebaute Bushauben-System.

Procoder

Er hat mit 13 Bit eine Auflösung von 8192 Schritten/Umdrehung, enthält ein optisch/magnetisches Abtastsystem und ist für Standardanwendungen geeignet.

Dignalizer

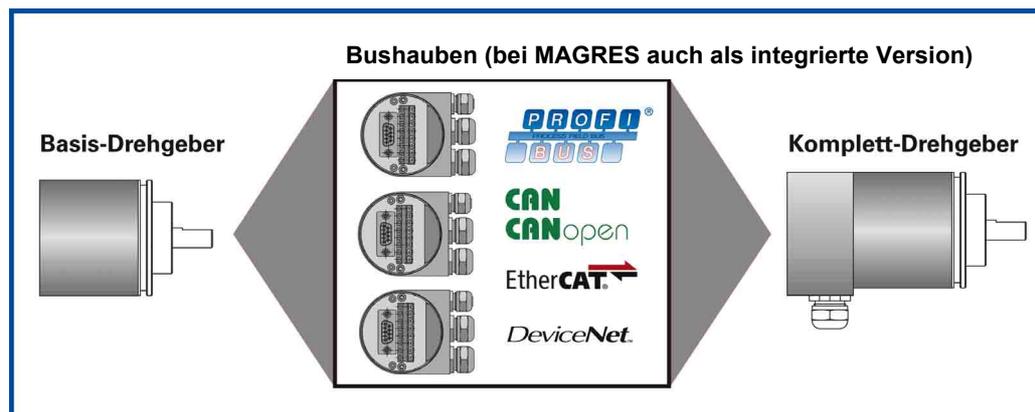
Er hat mit 18 Bit eine Auflösung von 262144 Schritten/Umdrehung, enthält ein optisch/magnetisches Abtastsystem mit integrierter Analog/Digital Umwandlung und ist für hochgenaue Messungen geeignet.

Die Basis-Drehgeber sind nochmals unterteilt in einen Singleturn- und Multiturn-Drehgeber. Der Multiturnteil kann bis 16 Bit oder 65536 Umdrehungen oder 18 Bit entsprechend 226144 Umdrehungen (Dignalizer) auflösen.

Die Bushauben unterscheiden sich durch die jeweiligen Busschnittstellen. Schnittstellen sind: CANopen, EtherCAT, DeviceNet und Profibus-DP.

Alle Drehgeber sind über die Bus-Schnittstelle parametrierbar.

Funktionsprinzip: MAGRES / Procoder / Dignalizer entsprechend für Endwelle, Hohlwelle oder Welle



BFF

Er hat mit 13 Bit eine Auflösung von 8192 Schritten/Umdrehung, enthält ein optisch/magnetisches Abtastsystem und ist für Standardanwendungen geeignet.

4. CAN-Bus und CANopen-Kommunikation

4.1. CAN-Bus

Der CAN-Bus (CAN: Controller Area Network) wurde ursprünglich von Bosch und Intel für die schnelle, kostengünstige Datenübertragung in der Kraftfahrzeug-Technik entwickelt. Der CAN-Bus wird heute auch in der industriellen Automatisierung verwendet.

Der CAN-Bus ist ein Feldbus (die Normen werden durch die Vereinigung CAN in Automation (CiA) festgelegt) über den Geräte, Aktoren und Sensoren verschiedener Hersteller miteinander kommunizieren.

4.1.1. CAN-Bus-Eigenschaften

- Datenrate von 1 MBit/s bei einer Netzausdehnung bis zu 20 m
- Beidseitig abgeschlossenes Netzwerk
- Busmedium ist Twisted-Pair-Kabel
- Echtzeitfähigkeit: Definierte max. Wartezeit für Nachrichten hoher Priorität.
- Theoretisch 127 Teilnehmer an einem Bus, physikalisch aber nur 32 (durch den Treiber bedingt).
- Sicherstellung netzweiter Datenkonsistenz. Gestörte Nachrichten werden für alle Netzknoten als fehlerhaft bekannt gemacht.
- Nachrichtenorientierte Kommunikation
Die Nachricht wird mit einer Nachrichtenennung (Identifizier) gekennzeichnet. Alle Netzknoten prüfen anhand des Identifizier, ob die Nachricht für sie relevant ist.
- Broadcasting, Multicasting
Alle Netzknoten erhalten gleichzeitig jede Nachricht. Daher ist eine Synchronisation möglich.
- Multi-Master-Fähigkeit
Jeder Teilnehmer im Feldbus kann selbständig Daten senden und empfangen, ohne dabei auf eine Priorität der Master angewiesen zu sein. Jeder kann seine Nachricht beginnen, wenn der Bus nicht belegt ist. Bei einem gleichzeitigen Senden von Nachrichten setzt sich der Teilnehmer mit der höchsten Priorität durch.
- Priorisierung von Nachrichten
Der Identifizier setzt die Priorität der Nachricht fest. Dadurch können wichtige Nachrichten schnell über den Bus übertragen werden.
- Restfehlerwahrscheinlichkeit
Sicherungsverfahren im Netzwerk reduzieren die Wahrscheinlichkeit einer unentdeckten, fehlerhaften Datenübertragung auf unter 10^{-11} . Praktisch kann von einer 100% sicheren Übertragung ausgegangen werden.
- Funktionsüberwachung
Lokalisation fehlerhafter oder ausgefallener Stationen. Das CAN-Protokoll beinhaltet eine Funktionsüberwachung von Netzknoten. Netzknoten, die fehlerhaft sind, werden in ihrer Funktion eingeschränkt oder ganz vom Netzwerk abgekoppelt.
- Datenübertragung mit kurzer Fehler-Erholzeit
Durch mehrere Fehlererkennungsmechanismen werden verfälschte Nachrichten mit grosser Wahrscheinlichkeit erkannt. Wird ein Fehler erkannt, so wird die Nachrichtensendung automatisch wiederholt.

Im CAN-Bus sind mehrere Netzwerkteilnehmer über ein Buskabel miteinander verbunden. Jeder Netzwerkteilnehmer kann Nachrichten senden und empfangen. Die Daten zwischen den Netzwerkteilnehmern werden seriell übertragen.

Netzwerkteilnehmer Beispiele für CAN-Bus-Geräte sind:

- Automatisierungsgeräte, z. B. SPS
- PCs
- Ein- /Ausgangsmodule
- Antriebssteuerungen
- Analysegeräte, z. B. ein CAN-Monitor
- Bedien- und Eingabegeräte als Mensch-Maschine Schnittstelle HMI (HMI, Human Machine Interface)
- Sensoren und Aktoren

4.2. CANopen

Unter technischer Leitung des Steinbeis Transferzentrums für Automatisierung wurde auf der Basis der Schicht 7 Spezifikation CAL (CAN-Application Layer) das CANopen-Profil entwickelt. Im Vergleich zu CAL sind in CANopen nur die für diesen Einsatz geeigneter Funktionen enthalten. CANopen stellt somit eine für die Anwendung optimierte Teilmenge von CAL dar und ermöglicht dadurch vereinfachten Systemaufbau und den Einsatz vereinfachte Geräte. CANopen ist optimiert für den schnellen Datenaustausch in Echtzeitsystemen.

Die Organisation CAN in Automation (CiA) ist zuständig für die geltenden Normen der entsprechenden Profile.

CANopen ermöglicht:

- Einfachen Zugriff auf alle Geräte- und Kommunikationsparameter
- Synchronisation von mehreren Geräten
- Automatische Konfiguration des Netzwerkes
- zyklischen und ereignisgesteuerten Prozessdatenverkehr

CANopen besteht aus vier Kommunikationsobjekten (COB) mit unterschiedlichen Eigenschaften:

- Prozess-Daten-Objekte für Echtzeitdaten (PDO)
- Service-Daten-Objekte für Parameter- und Programmübertragung (SDO)
- Netzwerk Management (NMT, Heartbeat)
- Vordefinierte Objekte (für Synchronisation, Notfallnachricht)

Alle Geräte- und Kommunikationsparameter sind in einem Objektverzeichnis gegliedert. Ein Objekt umfasst Name des Objekts, Datentyp, Anzahl Subindexe, Struktur der Parameter und die Adresse. Nach CiA ist dieses Objektverzeichnis in drei verschiedene Teile unterteilt: Kommunikationsprofil, Geräteprofil und ein herstellerspezifisches Profil. (siehe Objektverzeichnis)

4.3. CANopen-Kommunikation

4.3.1. Kommunikationsprofil

Die Kommunikation zwischen den Netzwerkteilnehmern und dem Master (PC / Steuerung) erfolgt über Objektverzeichnisse und Objekte. Die Objekte werden über einen 16bit-Index adressiert. Das CANopen-Kommunikationsprofil DS 301 standardisiert die verschiedenen Kommunikationsobjekte. Dementsprechend werden sie in mehrere Gruppen unterteilt:

- Prozessdatenobjekte PDO (process data object) zur Echtzeitübertragung von Prozessdaten
- Servicedatenobjekte SDO (service data object) für den Schreib- und Lesezugriff auf das Objektverzeichnis
- Objekte zur Synchronisation und Fehleranzeige von CAN-Teilnehmern:
 - SYNC-Objekt (synchronisation object) zur Synchronisation von Netzwerkteilnehmern
 - EMCY-Objekt (emergency object) zur Fehleranzeige eines Gerätes oder seiner Peripherie
- Netzwerk-Management NMT (network management) zur Initialisierung und Netzwerksteuerung
- Layer Setting Services LSS zur Konfiguration mittels Seriennummer, Revisionsnummer usw. inmitten eines vorhandenen Netzwerks

4.3.2. CANopen Meldungsaufbau

Der erste Teil einer Meldung ist die COB-ID (Identifier).

Aufbau der 11-Bit COB-ID:

| Funktions Code | | | | Node-ID | | | | | | |
|---------------------|--|--|--|---------------|--|--|--|--|--|--|
| 4 Bit FunktionsCode | | | | 7 Bit Node-ID | | | | | | |
| | | | | | | | | | | |

Der Funktionscode gibt Aufschluss über die Art der Meldung und die Priorität
Je niedriger die COB-ID, desto höher die Priorität der Meldung.

Broadcast Meldungen:

| Funktionscode | COB-ID |
|---------------|--------|
| NMT | 0 |
| SYNC | 80h |

Peer to Peer Meldungen:

| Funktionscode | COB-ID |
|-------------------------|----------------|
| Emergency | 80h + Node-ID |
| PDO1 (tx) ¹⁾ | 180h + Node-ID |
| PDO2 (tx) ¹⁾ | 280h + Node-ID |
| SDO (tx) ¹⁾ | 580h + Node-ID |
| SDO (rx) ¹⁾ | 600h + Node-ID |
| Heartbeat | 700h + Node-ID |
| LSS (tx) ¹⁾ | 7E4h |
| LSS (rx) ¹⁾ | 7E5h |

1): (tx) und (rx) aus der Sicht des Drehgebers

Die Node-ID kann über den CANopen-Bus zwischen 1 und 127 frei gewählt werden (wenn Drehschalter = 0).
Die Drehgeber werden mit Node-ID 1 ausgeliefert.

Eine Änderung erfolgt mit dem Service Daten Objekt 2101h oder über LSS.

Ein CAN-Telegramm besteht aus der COB-ID und bis zu 8 Byte Daten:

| COB-ID | DLC | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte 7 | Byte 8 |
|--------|-----|--------|--------|--------|--------|--------|--------|--------|--------|
| Xxx | x | xx |

Die genauen Telegramme werden später noch ausführlich aufgeführt.

4.3.3. Servicedaten-Kommunikation

Die Servicedatenobjekte entsprechen den Normen von CiA. Über Index und Subindex kann auf ein Objekt zugegriffen werden. Die Daten können angefordert oder gegebenenfalls ins Objekt geschrieben werden.

Allgemeines zu den SDO

Aufbau eines **SDO-Telegramms**:

| | | | | | | | | | |
|--------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------|-----|----------|----------|----------|----------|--------|--------|--------|--------|

Eine SDO-**COB-ID** setzt sich folgendermassen zusammen:

Master -> Drehgeber : 600h + Node-ID

Drehgeber -> Master : 580h + Node-ID

DLC (Data length code) bezeichnet die Länge des Telegramms. Diese setzt sich wie folgt zusammen:

1 Byte Kommando + 2 Byte Objekt + 1 Byte Subindex + Anzahl Datenbyte (0..4).

Das **Kommando-Byte** legt fest, ob Daten gelesen oder gesetzt werden und um wie viele Datenbyte es sich handelt:

| SDO Kommando | Beschreibung | Datenlänge | |
|--------------|-------------------|-------------|---------------------------------------|
| 22h | Download Request | Max. 4 Byte | Parameter an Drehgeber senden |
| 23h | Download Request | 4 Byte | |
| 2Bh | Download Request | 2 Byte | |
| 2Fh | Download Request | 1 Byte | |
| 60h | Download Response | - | Bestätigung der Übernahme an Master |
| 40h | Upload Request | - | Parameter vom Drehgeber anfordern |
| 42h | Upload Response | Max. 4 Byte | Parameter an Master mit max. 4 Byte |
| 43h | Upload Response | 4 Byte | |
| 4Bh | Upload Response | 2 Byte | |
| 4Fh | Upload Response | 1 Byte | |
| 80h | Abort Message | - | Drehgeber meldet Fehlercode an Master |

Eine **Abort Message** zeigt einen Fehler in der CAN-Kommunikation an. Das SDO Kommando-Byte ist 80h. Objekt und Subindex sind die des gewünschten Objektes. In Byte 8...5 steht der Fehler Code.

| ID | DLC | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte 7 | Byte 8 |
|----------------|-----|--------|----------|----------|----------|-----------|-----------|-----------|-----------|
| 580h + Node-ID | 8 | 80h | Objekt L | Objekt H | Subindex | ErrByte 0 | ErrByte 1 | ErrByte 2 | ErrByte 3 |

Byte 8..5 ergibt die SDO Abort Meldung (Byte 8 = MSB).

Folgende Meldungen werden unterstützt:

| | |
|-----------|-----------------------------------------|
| 05040001h | : Command Byte wird nicht unterstützt |
| 06010000h | : Falscher Zugriff auf ein Objekt |
| 06010001h | : Lesezugriff auf Write Only |
| 06010002h | : Schreibzugriff auf Read Only |
| 06020000h | : Objekt wird nicht unterstützt |
| 06090011h | : Subindex wird nicht unterstützt |
| 06090030h | : Wert ausserhalb der Limite |
| 06090031h | : Wert zu gross |
| 08000000h | : Genereller Error |
| 08000020h | : Falsche Speichersignatur ("save") |
| 08000021h | : Daten können nicht gespeichert werden |

Beispiele SDO

Anfrage eines Wertes vom Master beim Slave

Eine häufige Anfrage wird diejenige nach der Position sein. → Objekt 6004h

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 40h | 04h | 60h | 0 | x | x | x | x |

Antwort des Slaves auf die Anfrage eines Wertes

Die Position ist 4 Byte lang, die genauen Werte sind unter Objekt 6004h zu finden.

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 43h | 04h | 60h | 0 | a | b | c | d |

Schreiben eines Wertes vom Master in den Slave

Position setzen kann mit Preset erfolgen. → Objekt 6003h

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 22h | 03h | 60h | 0 | a | b | c | d |

Antwort des Slaves auf das Schreiben eines Wertes

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 60h | 03h | 60h | 0 | 0 | 0 | 0 | 0 |

4.3.4. Prozessdaten-Kommunikation

Prozessdatenobjekte dienen dem Echtzeit-Datenaustausch für Prozessdaten wie zum Beispiel die Position oder den Betriebsstatus. PDO's können synchron oder zyklisch (asynchron) gesendet werden. Der Drehgeber unterstützt das PDO1 und das PDO2. Beide PDO's liefern die aktuelle Position des Drehgebers und sind in den Objekten 1800h, 1801h, 1A00h, 1A01, 2800h, 2801h und 6200h festgelegt.

Synchron

Um die Prozessdaten synchron zu senden, muss im Objekt 1800h bzw. 1801h Subindex 2 ein Wert zwischen 1 und F0h (=240) eingeschrieben werden. Wenn nun der Wert 3 beträgt, wird das PDO auf jedes dritte Sync-Telegramm gesendet (beim Wert 1 wird auf jedes Sync-Telegramm gesendet), solange im Objekt 2800h bzw. 2801h ein 0 eingeschrieben ist. Ist dort zum Beispiel eine 5 eingeschrieben, wird das PDO nach wie vor auf jedes dritte Sync-Telegramm geschrieben, insgesamt aber nur 5 mal. Dem entsprechend folgt auf das 15. Sync-Telegramm das letzte PDO. Der Zähler für die Anzahl der zu übertragenden PDO's wird bei einer Positions-änderung oder das NMT-Reset zurückgesetzt. D.h. die Position wird, falls sie sich nicht ändert, 5mal gesendet. Ändert sich die Position, wird sie wieder 5 mal gesendet.

Im synchronen Betrieb werden die PDO vom Master über das Sync-Telegramm angefordert:

| Byte 0 | Byte 1 |
|-------------|--------|
| COB-ID = 80 | 0 |

Zyklisch (Asynchron)

Sollen die PDO's zyklisch gesendet werden, muss ins Objekt 1800h bzw. 1801h Subindex 2 der Wert FEh geschrieben werden. Zusätzlich muss im gleichen Objekt Subindex 5 die Zykluszeit in Millisekunden eingetragen werden. Die eingeschriebene Zeit wird auf 1ms aufgerundet. Wird der Wert 0ms gespeichert, werden die PDO's nicht gesendet. Die Funktion ist ausgeschaltet.

Eine weitere Möglichkeit bringt das Objekt 2800h bzw. 2801h: Beträgt der Wert 0, läuft das Zyklische Senden wie oben beschrieben. Beträgt der Wert 1, wird zyklisch geprüft ob eine Änderung des Wertes vorliegt. Wenn nicht, wird nicht gesendet. Beträgt der Wert 4, wird bei jedem Zyklus, falls eine Änderung besteht, das PDO viermal gesendet.

Übersicht

In nachfolgender Tabelle werden die verschiedenen Sende-Arten von PDO's zusammengefasst:

| 1800h | | 2800h | Kurzbeschreibung |
|-------|------|-------|-----------------------------------------------------------------------|
| Sub2 | Sub5 | | |
| FEh | 3ms | 0 | Zyklisches Senden alle 3ms |
| FEh | 5ms | 2 | Alle 5ms wird das PDO doppelt gesendet, falls eine Änderung vorliegt. |
| FEh | 0ms | xxx | PDO senden ausgeschaltet |
| 3 | xxx | 0 | Bei jedem dritten Sync-Telegramm senden |
| 3 | xxx | 2Bh | Auf jedes dritte Sync-Telegramm, aber gesamt nur 43 mal (=2Bh). |

PDO (Position)

PDO1 Telegrammaufbau:

| ID | DLC | Byte 1 | Byte 2 | Byte 3 | Byte 4 |
|------|-----|--------|--------|--------|--------|
| 181h | 4 | Xx | Xx | Xx | Xx |

ID : 180h + Node-ID
 Länge : 4 DataByte
 Byte1.. 4 : Aktuelle Position in Inkrementen

PDO2 Telegrammaufbau:

| ID | DLC | Byte 1 | Byte 2 | Byte 3 | Byte 4 |
|------|-----|--------|--------|--------|--------|
| 281h | 4 | Xx | Xx | Xx | Xx |

ID : 280h + Node-ID
 Länge : 4 DataByte
 Byte1.. 4 : Aktuelle Position in Inkrementen

4.3.5. Emergency-Dienst

Interne Gerätefehler oder Busprobleme lösen eine Emergency-Meldung aus:

| COB-ID | DLC | Byte0 | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte 7 |
|-------------|-----|-----------------------|--------|------------------------|--------------|--------|---------------|--------|--------|
| 80h+Node-ID | 8 | Error Code 00h 01h | | Errorregister 1001h | Alarms 6503h | | Warning 6505h | | n.u. |

Byte 0..1: Error Codes

| Error Code (hex) | Meaning |
|------------------|------------------------------------|
| 0000 | Error Reset or No Error |
| 1000 | Generic Error |
| 5530 | EEProm error |
| 6010 | Software reset (Watchdog) |
| 7320 | Position error |
| 7510 | Internal communication error |
| 8130 | Life Guard error or Hearbeat error |
| FF00 | Battery low |

Byte 2: Error-Register

| Bit | Meaning |
|-----|-----------------------|
| 0 | Generic Error |
| 4 | Communication error |
| 7 | manufacturer specific |

Byte 3..4 Alarms

| Bit | Meaning | Wert = 0 | Wert = 1 |
|-----|----------------------|----------|----------|
| 0 | Position error aktiv | Nein | Ja |

Byte 5..6 Warning

| Bit | Meaning | Wert = 0 | Wert = 1 |
|-----|---------------------|----------|------------------|
| 2 | CPU watchdog status | OK | Reset ausgeführt |
| 4 | Battery charge | OK | Ladung zu tief |

Byte 7: Wird nicht gebraucht

4.3.6. Netzwerkmanagement-Dienste

Das Netzwerkmanagement kann in zwei Gruppen unterteilt werden:

Mit den NMT-Diensten für die **Gerätekontrolle** können die Busteilnehmer initialisiert, gestartet und gestoppt werden.

Zusätzlich gibt es die NMT-Dienste zur **Verbindungsüberwachung**.

Beschreibung der NMT-Kommandos

Die Kommandos werden als unbestätigte Objekte übertragen und sind folgendermassen aufgebaut:

| Byte 0 | Byte 1 | Byte 2 |
|------------|---------------|---------------|
| COB-ID = 0 | Kommando Byte | Knoten Nummer |

COB-ID für NMT-Kommandos ist immer Null. Die Node-ID wird in Byte 2 des NMT-Kommandos übertragen.

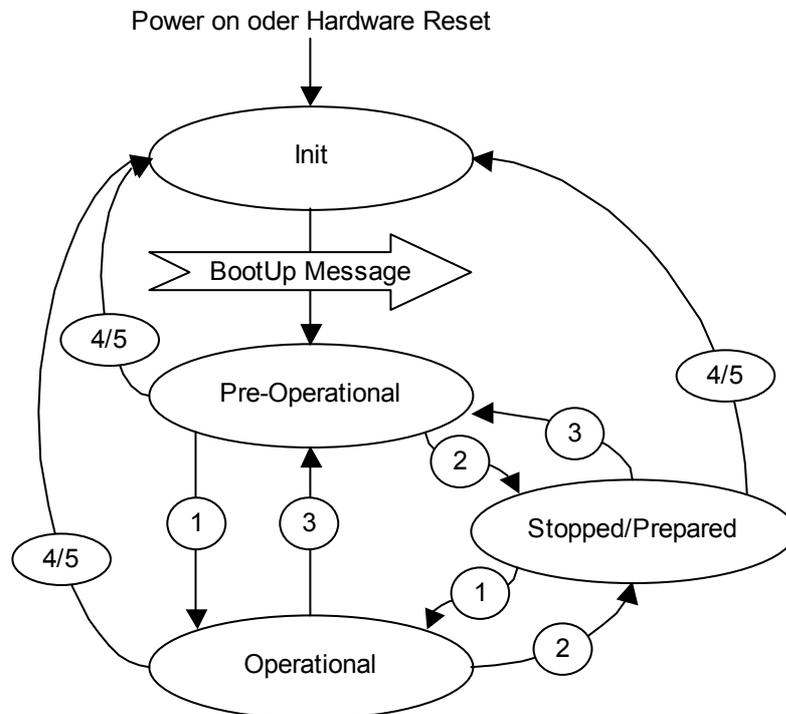
Kommando Byte

| Kommando Byte | Beschreibung | In State Event Zeichnung |
|---------------|----------------------------|--------------------------|
| 01h | Start Remote Node | 1 |
| 02h | Stop Remote Node | 2 |
| 80h | Enter Pre-Operational Mode | 3 |
| 81h, 82h | Reset Remote Node | 4, 5 |

Die **Knotennummer** entspricht der Node-ID des gewünschten Teilnehmers. Mit Knotennummer = 0 werden alle Teilnehmer angesprochen.

NMT State Event

Nach dem Initialisieren ist der Drehgeber im Pre-Operational Mode. In diesem Zustand können SDO Parameter gelesen und geschrieben werden. Um PDO Parameter anzufordern, muss der Drehgeber zuerst in den Zustand Operational Mode gefahren werden.



Die verschiedenen NMT Zustände

Init

Nach dem Initialisieren meldet sich der Drehgeber mit einer BootUp Meldung am CANopen-Bus. Danach geht der Drehgeber automatisch in den Zustand Pre-Operational Mode über.

Die COB-ID der BootUp Meldung setzt sich aus 700h und der Node-ID zusammen.

| COB-ID | Byte 0 |
|----------------|--------|
| 700h + Node-ID | 00 |

Pre-Operational Mode

Im Pre-Operational Mode können SDO gelesen und geschrieben werden.

Operational Mode

Im Zustand Operational Mode sendet der Drehgeber die gewünschten PDO's. Zudem können SDO gelesen und geschrieben werden.

Stopped oder Prepared Mode

Im Stopped Mode ist nur NMT Kommunikation möglich. Es können keine SDO Parameter gelesen oder gesetzt werden. LSS ist nur im Stopped Mode möglich.

Zustandswechsel

Start Remote Node (1)

Mit dem Startbefehl wird der Drehgeber in den Zustand Operational Mode gebracht.

| COB-ID | Kommando Byte | Knoten Nummer |
|--------|---------------|---------------|
| 0 | 1h | 0..127 |

Stop Remote Node (2)

Mit dem Stoppbefehl wird der Drehgeber in den Zustand Stopped oder Prepared Mode gebracht.

| COB-ID | Kommando Byte | Knoten Nummer |
|--------|---------------|---------------|
| 0 | 2h | 0..127 |

Enter Pre-Operational Mode (3)

Wechsle in den Zustand Pre-Operational Mode.

| COB-ID | Kommando Byte | Knoten Nummer |
|--------|---------------|---------------|
| 0 | 80h | 0..127 |

Reset Remote Node (4) oder Reset Kommunikation (5)

Mit dem Resetbefehl wird der Drehgeber neu initialisiert.

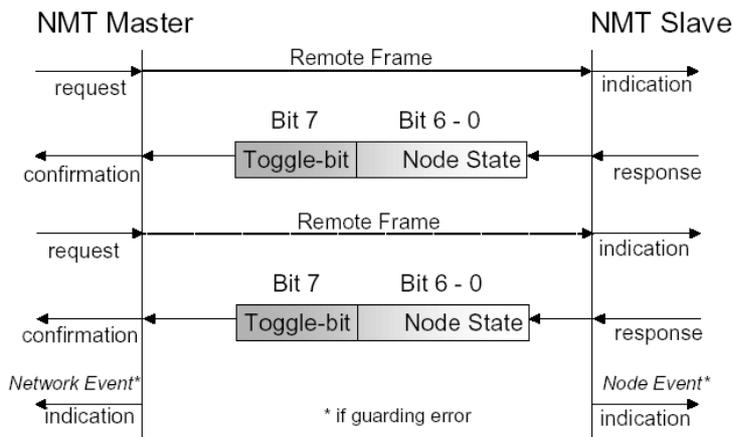
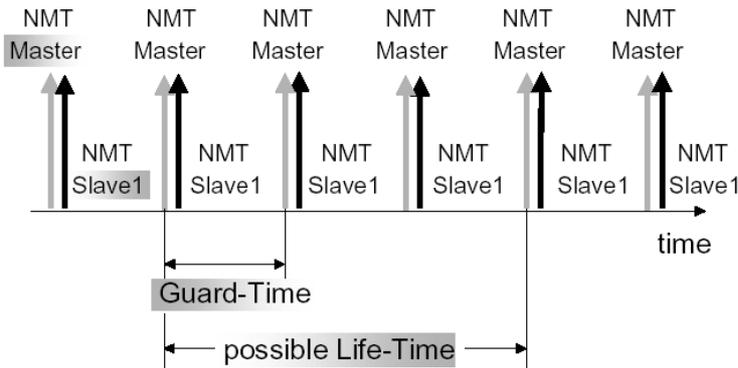
Reset Remote Node (4):

| COB-ID | Kommando Byte | Knoten Nummer |
|--------|---------------|---------------|
| 0 | 81h | 0..127 |

Reset Kommunikation (5):

| COB-ID | Kommando Byte | Knoten Nummer |
|--------|---------------|---------------|
| 0 | 82h | 0..127 |

Node und Life Guarding



Fehlerfall wird im Objekt 1029h-1h "Communication Error" definiert.

Beispiel eines Nodeguarding-Protokolls:

| COB-ID | Data/ Remote | Byte 0 |
|--------|--------------|------------|
| 701h | r | 00h (0d) |
| 701h | d | FFh (255d) |
| 701h | r | 00h (0d) |
| 701h | d | 7Fh (127d) |

Mögliche NMT-Zustände der Teilnehmer:

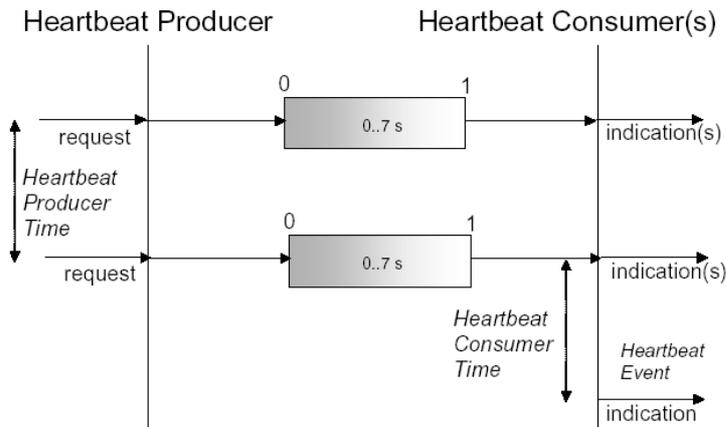
- 0: BootUp-Event
- 4: Stopped
- 5: Operational
- 127: Pre-Operational

Die unteren 7 Bits ergeben in diesem Fall 7Fh. D.h. der Drehgeber befindet sich im Pre-Operational Modus.

Zur Überwachung der Teilnehmer kann das Nodeguarding-Protokoll verwendet werden. Die CANopen-Nutzerorganisation Can in Automation CiA hat die Weisung herausgegeben, dass man wenn möglich das aktuelle Überwachungsprotokoll Heartbeat verwenden sollte. Will man trotzdem das Nodeguarding verwenden, muss im Objekt 2110h das Bit5 gesetzt werden.

Der NMT-Master kann eine Datenbank anlegen mit den jeweiligen NMT-Zuständen jedes einzelnen Teilnehmers. Mit diesem Protokoll kann überprüft werden, ob sich ein Teilnehmer vom Bus zurückgezogen hat. Zusätzlich kann auch jeder Teilnehmer überwachen, ob die Steuerung noch aktiv ist. Der NMT-Master startet den Überwachungsdiens mit einem Remot-Frame an den gewünschten Teilnehmer. Durch jedes Remote-Frame wird beim Teilnehmer die Life-Time zurück gesetzt. Zusätzlich liefert der Teilnehmer seinen NMT-Zustand zurück. Somit kann der NMT-Master überprüfen, ob sich der Teilnehmer im richtigen NMT-Zustand befindet und im Fehlerfall darauf reagieren.

Falls die Life-Time abläuft, wird ein "Node Event" ausgelöst. Das Verhalten im

Heartbeat-Protokoll


Wahlweise kann das neue Heartbeat-Protokoll verwendet werden. Heartbeat ist aktiv, wenn im Objekt 2110h Bit5 auf '0' ist. Für neue Applikationen empfiehlt es sich das moderne Überwachungsprotokoll Heartbeat zu verwenden. Ein "Heartbeat-Producer" produziert zyklisch eine Heartbeat-Meldung. Ein oder mehrere "Heartbeat-Consumer" können diese Heartbeat-Meldung empfangen. Falls das zyklische senden dieser Heartbeat-Meldung ausbleibt, wird ein "Heartbeat Event" ausgelöst. Das Verhalten im Fehlerfall wird im Objekt 1029h-1h "Communication Error" definiert.

Beispiel einer Heartbeat-Protokolls

| COB-ID | Data/Remote | Byte 0 |
|--------|-------------|------------|
| 701h | d | 7Fh (127d) |

Die Heartbeat-Meldungen bestehen aus der COB-ID und einem Byte. In diesem Byte wird der NMT-Zustand überliefert.

- 0: BootUp-Event
- 4: Stopped
- 5: Operational
- 127: Pre-Operational

D.h. der Drehgeber befindet sich im Pre-Operational Modus (7Fh = 127).

4.3.7. Layer Setting Services

Im Frühling 2000 wurde von CiA ein neues Protokoll entworfen, um ein einheitliches Auftreten zu gewährleisten. Beschrieben ist das Vorgehen unter

Layer Setting Services and Protokoll, CiA Draft Standard Proposal 305 (LSS).

Der Drehgeber wird von uns standardmässig mit der Node-ID 1 und der Baudrate 50 kBaud ausgeliefert. Es können mehrere Drehgeber mit der selben Node-ID an das Bussystem angeschlossen werden. Um nun die einzelnen Drehgeber ansprechen zu können, wird LSS verwendet.

Jeder Drehgeber besitzt eine eindeutige Seriennummer und wird über diese Nummer angesprochen. Also können beliebig viele Drehgeber mit gleicher Node-ID an ein Bussystem angeschlossen werden und dann über LSS initialisiert werden. Es können sowohl die Node-ID als auch die Baudrate neu gesetzt werden. LSS kann nur im **Stopped Mode** ausgeführt werden.

Meldungsaufbau

COB-ID:

Master → Slave: 2021 = 7E5h

Master ← Slave: 2020 = 7E4h

Nach der COB-ID wird ein LSS command specifier gesandt.

Danach werden bis zu sieben Datenbyte angehängt.

| | | | | | | | | |
|--------|----|--------|--------|--------|--------|--------|--------|--------|
| COB-ID | cs | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte 7 |
|--------|----|--------|--------|--------|--------|--------|--------|--------|

Switch Mode Global

| | | | |
|--------|-----|------|----------|
| 7E5h → | 04h | Mode | Reserved |
|--------|-----|------|----------|

Mode : 0 → Operationsmode

1 → Konfigurationsmode

Switch Mode Selektiv

Mit folgendem Ablauf kann ein ganz bestimmter Drehgeber im Bussystem angesprochen werden.

| | | | |
|--------|-----|----------|----------|
| 7E5h → | 40h | VendorId | Reserved |
|--------|-----|----------|----------|

| | | | |
|--------|-----|-------------|----------|
| 7E5h → | 41h | ProductCode | reserved |
|--------|-----|-------------|----------|

| | | | |
|--------|-----|----------------|----------|
| 7E5h → | 42h | RevisionNumber | reserved |
|--------|-----|----------------|----------|

| | | | |
|--------|-----|--------------|----------|
| 7E5h → | 43h | SerialNumber | reserved |
|--------|-----|--------------|----------|

| | | | |
|--------|-----|------|----------|
| 7E4h ← | 44h | Mode | reserved |
|--------|-----|------|----------|

VendorId : 5Fh

ProductCode : Interner Produkt-Code für den jeweiligen Drehgeber

RevisionNumber : Aktuelle Revisionsnummer des Drehgebers

SerialNumber : Eindeutige, fortlaufende Seriennummer

Mode : Antwort des Drehgebers ist der neue Mode (0=Operationsmode; 1=Konfigurationsmode)

Node-ID setzen

| | | | |
|--------|----|---------|----------|
| 7E5h → | 17 | Node-ID | reserved |
|--------|----|---------|----------|

| | | | | |
|--------|-----|---------|------------|----------|
| 7E4h ← | 11h | ErrCode | Spec Error | reserved |
|--------|-----|---------|------------|----------|

Node-ID : Die neue Node-ID des Drehgebers

ErrorCode : 0=OK; 1=Node-ID ausserhalb des Bereiches; 2..254=reserved; 255→specificError

SpecificError : Falls ErrorCode=255 → Applikationsspezifischer Errorcode.

Produktcode anfordern

Produktcode eines selektierten Drehgebers anfordern

| | | |
|--------|-----|----------|
| 7E5h → | 5Bh | reserved |
|--------|-----|----------|

| | | | |
|--------|-----|--------------|----------|
| 7E4h ← | 5Bh | ProdukteCode | reserved |
|--------|-----|--------------|----------|

Produktcode : Herstellerabhängiger Produktcode

Revisionsnummer anfordern

Revisionsnummer eines selektierten Drehgebers anfordern

| | | |
|--------|-----|----------|
| 7E5h → | 5Ch | reserved |
|--------|-----|----------|

| | | | |
|--------|-----|------------------------|----------|
| 7E4h ← | 5Ch | 32 Bit Revisionsnummer | reserved |
|--------|-----|------------------------|----------|

Revisionsnummer : aktuelle Revision

Seriennummer anfordern

Seriennummer eines selektierten Drehgebers anfordern

| | | |
|--------|-----|----------|
| 7E5h → | 5Dh | reserved |
|--------|-----|----------|

| | | | |
|--------|-----|---------------------|----------|
| 7E4h ← | 5Dh | 32 Bit Seriennummer | reserved |
|--------|-----|---------------------|----------|

Seriennummer : eindeutige fortlaufende Seriennummer des Drehgebers

Bereichsanfrage

Drehgeber können auch in einem gewissen Bereich gesucht werden. Hierzu werden folgende Objekte nacheinander versandt:

| | | | |
|--------|-----|----------|----------|
| 7E5h → | 46h | VendorId | reserved |
|--------|-----|----------|----------|

| | | | |
|--------|-----|-------------|----------|
| 7E5h → | 47h | ProductCode | reserved |
|--------|-----|-------------|----------|

| | | | |
|--------|-----|--------------------|----------|
| 7E5h → | 48h | RevisionNumber LOW | reserved |
|--------|-----|--------------------|----------|

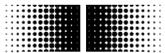
| | | | |
|--------|-----|---------------------|----------|
| 7E5h → | 49h | RevisionNumber HIGH | reserved |
|--------|-----|---------------------|----------|

| | | | |
|--------|-----|------------------|----------|
| 7E5h → | 4Ah | SerialNumber LOW | reserved |
|--------|-----|------------------|----------|

| | | | |
|--------|-----|-------------------|----------|
| 7E5h → | 4Bh | SerialNumber HIGH | reserved |
|--------|-----|-------------------|----------|

Jeder Drehgeber mit den entsprechenden Parametern meldet sich mit folgender Meldung:

| | | |
|--------|-----|----------|
| 7E4h ← | 4Fh | reserved |
|--------|-----|----------|



4.4. Drehgeber Profil

4.4.1. Drehgeber-Objekte Übersicht

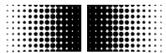
Nach CiA (CAN in Automation) werden die Objekte in drei Gruppen unterteilt:

- **Standard-Objekte:**
1000h, 1001h, 1018h
- **Herstellerspezifische Objekte:**
2000h - 5FFFh
- **Gerätespezifische Objekte:**
Alle anderen Objekte von 1000h - 1FFFh, 6000h - FFFFh

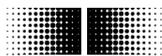
Folgende Tabelle zeigt eine Zusammenfassung aller vom Drehgeber unterstützten SDO Objekte.

| | |
|---------------------|----------------------------------------------------------------------|
| Objekt | Objekt Nummer in Hex |
| Name | --- |
| Format | U/I = Unsigned/Integer, Zahl = Anzahl Bit, ARR = Array, REC = Record |
| Zugriff | ro = ReadOnly, wo = WriteOnly, rw = ReadWrite |
| Default | Default Wert beim ersten Init oder Restore Default |
| Save | ja → Wird im EEPROM gespeichert |
| Beschreibung | zusätzliche Beschreibung |

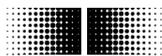
| Objekt Sub-Index | Name | Format | Zugriff | Default | Save | Beschreibung |
|---------------------|--------------------------|--------|---------|-------------------------------------------------------------------|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1000h | Device Type | U32 | ro | 00020196h 00010196h | | Multiturn Drehgeber: Byte 0..1: ProfilNr=196h=406 Byte 2..3: Drehgeber Type =2 (Multiturn, absolut) Singleturn Drehgeber: Byte 0..1: ProfilNr=196h=406 Byte 2..3: Drehgeber Type =1 (Singleturn, absolut) |
| 1001h | Error Register | U8 | ro | 0h | | Bit0 = Generic error Bit4 = Communication error (overrun, ...) Bit7 = Manufacturer specific |
| 1003h | PreDefined ErrorField | ARR | | | | Enthält die letzten 8 Fehler oder Warnungen |
| 00h | Grösster Subindex | U8 | rw | 0h | | Anzahl gespeicherten Meldungen (0..8) |
| 01h | Letzter Eintrag | U32 | ro | | | Letzter Fehler oder Warnung 1000h Generic Error 5530h EEPROM Error 6010h Software Reset (Watchdog) 7320h Positions-Error 7510h Interner Kommunikations-Error 8130h Life Guard Error oder Heartbeat Error FF00h Batterieladung zu tief |
| .. | .. | .. | .. | .. | .. | .. |
| 08h | Ältester Eintrag | U32 | ro | | | Fehler oder Warnung, Siehe Sub-Index 01h |
| 1005h | Sync COB-ID | U32 | rw | 80h | ja | COB-ID des Sync Objektes |
| 1008h | DeviceName | U32 | ro | "BFx" "BPMx" "BPSx" "BMMx" "BMSx" "BOMx" "BOSx" | ja | Devicename = "BFx" 13Bit Singleturn "BPMx" Procoder Multiturn "BPSx" Procoder Singleturn "BMMx" MAGRES Multiturn "BMSx" MAGRES Singleturn "BOMx" Digitalizer Multiturn "BOSx" Digitalizer Singleturn |
| 1009h | Hardware Version | U32 | ro | werkseitig | | Produkt Hardware Version in ASCII |
| 100Ah | Software Version | U32 | ro | werkseitig | | Produkt Software Version in ASCII |
| 100Ch | Guard Time | U16 | rw | 0h | ja | Timer für Nodeguarding |
| 100Dh | Life Time factor | U8 | rw | 0h | ja | Multiplikator der Guard Time |
| 1010h | Store Parameters | ARR | | | | |
| 00h | Grösster Subindex | U8 | ro | 4h | | Anzahl Speichermöglichkeiten = 4 |



| Objekt Sub-Index | Name | Format | Zugriff | Default | Save | Beschreibung |
|---------------------|----------------------------------|--------|---------|---------------------------------------------------------------------------|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 01h | Alle Parameter speichern | U32 | rw | | | =“evas“ (0x65766173) zum speichern |
| 02h | Communication Parameters | U32 | rw | | | =“evas“ (0x65766173) zum speichern |
| 03h | Application Parameters | U32 | rw | | | =“evas“ (0x65766173) zum speichern |
| 04h | Manuf. Specific Parameters | U32 | rw | | | =“evas“ (0x65766173) zum speichern |
| 1011h | Restore Default Parameters | ARR | | | | |
| 00h | Grösster Subindex | U8 | ro | 4h | | Anzahl Zurücksetz Möglichkeiten = 4 |
| 01h | Alle Parameter | U32 | rw | | | =“daol“ (0x64616F6C) zum laden |
| 02h | Communication Parameters | U32 | rw | | | =“daol“ (0x64616F6C) zum laden |
| 03h | Application Parameters | U32 | rw | | | =“daol“ (0x64616F6C) zum laden |
| 04h | Manufacturer Specific Parameters | U32 | rw | | | =“daol“ (0x64616F6C) zum laden |
| 1014h | Emergency COB-ID | U32 | rw | 80h + Node-ID | ja | COB-ID des Emergency Objektes |
| 1016h | Consumer heartbeat time | ARR | | | | |
| 00h | Grösster Subindex | U8 | ro | 1h | | |
| 01h | Consumer heartbeat time | U32 | rw | 10000h | ja | Bit0..15 Consumer Heartbeat time in ms Bit16..23 Node-ID |
| 1017h | Producer Heartbeat Time | U16 | rw | 0h | ja | Producer Heartbeat time in ms |
| 1018h | Identity Object | REC | ro | | | |
| 00h | Grösster Subindex | U8 | ro | 4h | | |
| 01h | VendorID | U32 | ro | 5Fh | ja | Von CiA vergebene Vendornummer |
| 02h | Product Code | U32 | ro | 02h 0Ah 0Bh 0Ch 0Dh 0Eh 0Fh 16h 17h 18h 19h | ja | Product Code: 02h = BFx 0Ah = Procoder Multiturn Bushaube 0Bh = Procoder Singleturn Bushaube 0Ch = MAGRES Multiturn Bushaube 0Dh = MAGRES Singleturn Bushaube 0Eh = Dignalizer Multiturn Bushaube 0Fh = Dignalizer Singleturn Bushaube 16h = MAGRES Multiturn Integriert 17h = MAGRES Singleturn Integriert 18h = Dignalizer Multiturn Integriert 19h = Dignalizer Singleturn Integriert |
| 03h | Revision Number | U32 | ro | werkseitig | ja | Produkt Revisionsnummer |
| 04h | Serial Number | U32 | ro | werkseitig | ja | Eindeutige fortlaufende Seriennummer |
| 1029h | Error behaviour | ARR | | | | |
| 00h | Grösster Subindex | U8 | ro | 1h | | |
| 01h | Communication error | U8 | rw | 1h | ja | 0h = Wechsel in den Pre-Operation Modus 1h = kein Modus-Wechsel 2h = Wechsel in den Stop Modus 3h = Knoten reseten |
| 1800h | Transmit PDO1 Parameter | REC | | | | |
| 00h | Grösster Subindex | U8 | ro | 5h | | |
| 01h | COB-ID | U32 | rw | 180h+id | ja | PDO ID = 180h + Node-ID |
| 02h | PDO Type | U8 | rw | FEh | ja | FEh=UserDefiniert, zyklisch |
| 05h | EventTimer | U16 | rw | 203h | ja | Zykluszeit in ms |
| 1801h | Transmit PDO2 Parameter | REC | | | | |
| 00h | Grösster Subindex | U8 | ro | 5h | | |
| 01h | COB-ID | U32 | rw | 280h+id | ja | PDO ID = 280h + Node-ID |
| 02h | PDO Type | U8 | rw | 2h | ja | 2h= Synchron Betrieb |
| 05h | EventTimer | U16 | rw | 100h | ja | Zykluszeit in ms |
| 1A00h | Transmit PDO1 Mapping | ARR | | | | |



| Objekt Sub-Index | Name | Format | Zugriff | Default | Save | Beschreibung |
|---------------------|----------------------------|--------|---------|---------|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 00h | Grösster Subindex | U8 | ro | 1h | | |
| 01h | Inhalt des PDO1 | U32 | ro | 6004h | | ReadOnly, obwohl von CiA als ReadWrite |
| 1A01h | Transmit PDO2 Mapping | ARR | | | | |
| 00h | Grösster Subindex | U8 | ro | 1h | | |
| 01h | Inhalt des PDO2 | U32 | ro | 6004h | | ReadOnly, obwohl von CiA als ReadWrite |
| 2100h | Baudrate | U8 | rw | 2h | ja | Nach setzen der Baudrate muss EEPROM gespeichert und neu initialisiert werden 0=10 kBit/s 1=20 kBit/s 2=50 kBit/s 3=100 kBit/s 4=125 kBit/s 5=250 kBit/s 6=500 kBit/s 7=800 kBit/s 8=1000 kBit/s |
| 2101h | Node-ID | U8 | rw | 1h | ja | Node Nummer 1..127 möglich Nach setzen der Baudrate muss EEPROM gespeichert und neu initialisiert werden. |
| 2110h | Versionssteuerung | U32 | rw | 8h | ja | Bit1 = Drehrichtungssinn (Objekt 6000h Bit0) 0 Nicht invertiert 1 Invertiert Bit2 = Skalierungsfunktion (Objekt 6000h Bit2) 0 Freigegeben 1 Gesperrt Bit3 = 0 BusOFF wird nicht zurückgesetzt 1 Wenn BusOFF wird der Bus wieder zurückgesetzt Bit5 = 0 Heartbeat-Protokoll aktiv 1 Nodeguardin-Protokoll aktiv Bit6 = 0 Beim SYNC-Telegramm wird der bereits ermittelte Positionswert ausgegeben 1 Beim SYNC-Telegramm wird Positionswert neu eingelesen Bit7 = minimaler Positions-Jitter beim SYNC 0 Positionsermittlung zyklisch 1 Position wird erst auf ein SYNC-Telegramm eingelesen (Bit6 muss aktiviert sein) → minimaler Positions-Jitter Bit8 = PDO1 Zeitverzögerung 2ms 0 1800h-5h = 6200h 1 1800h-5h = 6200h + 2ms |
| 2201h | Statistik | REC | | | | |
| 00h | Grösster Subindex | U8 | ro | 3h | | Anzahl Subindexe |
| 01h | Anzahl Position-Fehler | U32 | ro | | ja | Positionskontrolle |
| 02h | Zeit in Sekunden | U32 | ro | | ja | Zeit seit letztem Reset |
| 03h | Anzahl TimerReset Watchdog | U32 | ro | | ja | TimerWatchDog |
| 2300h | Customer EEPROM Bereich | ARR | | | | In diesem Objekt können beliebige Daten abgespeichert werden |
| 00h | Grösster Subindex | U8 | ro | 7h | | |
| 01h | Data0 | U16 | rw | 0h | ja | |
| 02h | Data1 | U16 | rw | 0h | ja | |
| 03h | Data2 | U16 | rw | 0h | ja | |
| 04h | Data3 | U16 | rw | 0h | ja | |
| 05h | Data4 | U16 | rw | 0h | ja | |
| 06h | Data5 | U16 | rw | 0h | ja | |
| 07h | Data6 | U16 | rw | 0h | ja | |
| 2800h | PDO1-Zusatz / EventTrigger | U8 | rw | 0h | ja | Wiederholungszähler für PDO1 |



| Objekt Sub-Index | Name | Format | Zugriff | Default | Save | Beschreibung |
|---------------------|--------------------------------|--------|---------|-----------------------------------------------------------------------|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2801h | PDO2-Zusatz / EventTrigger | U8 | rw | 0h | ja | Wiederholungszähler für PDO2 |
| 6000h | Operating Parameter | U16 | rw | 4h | ja | Bit0 = Position steigend, Blick auf Welle 0 CW 1 CCW Bit2 = 0 Skalierungsfunktion aus 1 Skalierungsfunktion ein |
| 6001h | Auflösung | U32 | rw | 2000h 1000h 40000h | ja | Auflösung in Schritte/Umdrehung: 13Bit = Procoder, BFX 12Bit = MAGRES 18bit = Digitalizer |
| 6002h | Gesamtmessbereich in Schritten | U32 | rw | 20000000h 2000h 40000000h 1000h (1)00000000h 40000h | ja | Gesamtmessbereich in Schritten 29Bit = Procoder Multiturn 13Bit = Procoder Singleturn, BFX 30Bit = MAGRES Multiturn 12Bit = MAGRES Singleturn 32Bit = Digitalizer Multiturn 18Bit = Digitalizer Singleturn |
| 6003h | Preset Wert in Schritten | U32 | rw | 0h | ja | Preset in Schritten → Offset |
| 6004h | Position in Schritten | U32 | ro | | | Positionswert inkl. Offset in Schritten |
| 6200h | Cyclic Timer für PDO1 | U16 | rw | 203h | ja | In ms, identisch Objekt 1800h, Subindex 5 |
| 6500h | Operating Status | U16 | ro | 4h | | Bit0 = Position steigend, Blick auf Welle 0 CW 1 CCW Bit2 = 0 Skalierungsfunktion aus 1 Skalierungsfunktion ein |
| 6501h | Max. Auflösung | U32 | ro | 2000h 4000h 40000h | | Max Auflösung in Schritte/Umdrehung: 13Bit = Procoder, BFX 14Bit = MAGRES 18Bit = Digitalizer |
| 6502h | Gesamtmessbereich in Schritten | U32 | ro | 20000000h 2000h (1)00000000h 4000h (1)00000000h 40000h | | (ist ausserhalb der Spezifikation von CiA) Gesamtmessbereich in Schritten: 29Bit = Procoder Multiturn 13Bit = Procoder Singleturn, BFX 32Bit = MAGRES Multiturn 14Bit = MAGRES Singleturn 32Bit = Digitalizer Multiturn 18Bit = Digitalizer Singleturn |
| 6503h | Alarmer | U16 | ro | 0h | | Folgende Alarmer werden ausgewertet: Bit0 = Positions-Error |
| 6504h | Unterstützte Alarmer | U16 | ro | 1h | | Folgende Alarmer werden unterstützt: Bit0 = Positions-Error |
| 6505h | Warnungen | U16 | ro | 0h | | Folgende Warnungen werden ausgewertet: Multiturn Drehgeber: Bit2 = CPU watchdog status Bit4 = Batterieladung Singleturn Drehgeber: Bit2 = CPU watchdog status |
| 6506h | Unterstützte Warnungen | U16 | ro | 14h 04h | | Folgende Warnungen werden unterstützt: Multiturn Drehgeber: Bit2 = CPU Watchdog Status Bit4 = Batterieladung Singleturn Drehgeber: Bit2 = CPU Watchdog Status |
| 6507h | Profil & Software-Version | U32 | ro | werkseitig | | Byte 0..1: Profil-Version = 2.01 = 0201h Byte 2..3: Software-Version = 1.05 = 0105h |
| 6508h | Operating Time | U32 | ro | 0h | | Zeit in 1/10 Stunden seit letztem Reset |
| 6509h | Offset | U32 | ro | 0h | | Offset aus Preset berechnet → 6003h |
| 650Bh | Serie Nummer | U32 | ro | werkseitig | ja | Seriennummer Objekt 1018h-4h |

5. Diagnose und Wissenswertes

5.1. Fehlerdiagnose Feldbus-Kommunikation

- Falls der Drehgeber über den CANopen-Bus nicht angesprochen werden kann, sollten Sie als erstes die Anschlüsse überprüfen.

Sind die Anschlüsse in Ordnung, sollte als nächstes der Feldbusbetrieb getestet werden. Dazu wird ein CAN-Monitor benötigt, welcher die CANopen-Kommunikation aufzeichnet und die Telegramme darstellt.

- Nun sollte der Drehgeber beim Aus- und wieder Einschalten der Spannungsversorgung eine BootUp-Message absetzen.

Sollte keine BootUp-Meldung erscheinen, prüfen Sie, ob die Baudraten des Antriebs, des CAN-Monitors und des Bussystems übereinstimmen.

- Wenn Sie Schwierigkeiten haben die Verbindung zu einem Teilnehmer aufzunehmen, prüfen Sie die Knotennummer und die Baudrate.

Die Baudrate muss überall gleich eingestellt werden. Die Knotennummer (Node-ID, Knotenadresse) muss zwischen 1 und 127 liegen. Jeder Busteilnehmer muss eindeutig mit einer Node-ID definiert werden. D.h. es darf auf keinen Fall mehrere Male die selbe Node-ID zugeordnet werden.

Node-ID und Baudrate können auch bequem über den LSS-Dienst eingestellt werden.

5.2. Fehlerdiagnose über Feldbus

Der Drehgeber verfügt über mehrere Objekte und Meldungen, welche den Status oder Fehlerzustände des Drehgebers umschreiben:

- Objekt 1001h: Dieses Objekt ist ein Error-Register für den Fehlerzustand des Gerätes.
- Objekt 1003h: In diesem Objekt werden die letzten acht Fehlercodes und Warnungen gespeichert.
- Objekt Emergency (80h + Node-ID): Hochpriorie Fehlermeldung eines Teilnehmers mit Error code und Error register.
- SDO Abort Message: Falls die SDO-Kommunikation nicht korrekt abläuft, enthält die SDO-Antwort einen Abort code.

Objekt 1001h Error register

In diesem Register wird das Vorhandensein eines Gerätefehlers sowie dessen Art angezeigt.

Bit 0: Generic error

Die restlichen Bits werden durch unseren Drehgeber nicht unterstützt.

Objekt 1003h Predefined error field

In diesem Objekt werden die acht zuletzt aufgetretenen Error codes aus den Objekten 6503h und 6505h gespeichert, wobei der letzte Error im Subindex1 und der älteste Error unter Subindex8 eingetragen ist.

Objekt Emergency

Fehlermeldung eines Teilnehmers.

SDO Abort Message

Erfolgt die SDO-Kommunikation nicht problemlos, wird als SDO-Antwort ein Abort code gesendet:

| | |
|-----------|-----------------------------------------|
| 05040001h | : Command Byte wird nicht unterstützt |
| 06010000h | : Falscher Zugriff auf ein Objekt |
| 06010001h | : Lesezugriff auf Write Only |
| 06010002h | : Schreibzugriff auf Read Only |
| 06020000h | : Objekt wird nicht unterstützt |
| 06090011h | : Subindex wird nicht unterstützt |
| 06090030h | : Wert ausserhalb der Limite |
| 06090031h | : Wert zu gross |
| 08000000h | : Genereller Error |
| 08000020h | : Falsche Speichersignatur ("save") |
| 08000021h | : Daten können nicht gespeichert werden |

Serviceadresse

Baumer Electric AG
Postfach, Hummelstrasse 17
CH-8501 Frauenfeld
Service-Tel.-Nr.: +41(0)52 728 11 22 (Montag-Freitag 8.00-17.00 h)
sales.ch@baumerelectric.com
www.baumerelectric.com

5.3. Wissenswertes zum Sensor**Node-ID neu setzen**

1. Die Node-ID wird mit dem Baumer-spezifischen Objekt 2100h neu gesetzt.
2. Nach dem Setzen der Node-ID muss diese mit dem Objekt 1010h im EEPROM gespeichert werden.
3. Beim nächsten Initialisieren wird sich der Sensor mit der neuen Node-ID melden.

Baudrate neu setzen

1. Die Baudrate wird mit dem Baumer-spezifischen Objekt 2101h neu gesetzt.
2. Nach dem Setzen der Baudrate muss diese mit dem Objekt 1010h im EEPROM gespeichert werden.
3. Beim nächsten Initialisieren wird sich der Sensor auf der neuen Baudrate melden.
4. ! NICHT VERGESSEN DEN MASTER AUF DIE NEUE BAUDRATE EINSTELLEN !

Abschirmung

Da der Drehgeber je nach Einbaulage nicht immer auf einem definierten Erdpotential liegt, sollte der Drehgeber-Flansch zusätzlich immer mit Erdpotential verbunden werden. Grundsätzlich sollte der Drehgeber über eine abgeschirmte Leitung angeschlossen werden.

Wenn möglich sollte der Kabelschirm beidseitig aufgelegt werden. Es ist darauf zu achten, dass keine Ausgleichströme über den Drehgeber abgeleitet werden.

6. Applikationen

6.1. Objekte setzen und lesen

Um ein Objekt (SDO) zu überschreiben oder zu lesen werden immer zwei Telegramme gesendet.

Objekt setzen

Zuerst sendet der Master den zu setzenden Wert. Anschliessend sendet der Drehgeber die Bestätigung.

Wert (ba) wird gesendet:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 2Bh | 00h | 23h | 3h | a | b | x | x |

Bestätigung:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 60h | 00h | 23h | 3h | 0 | 0 | 0 | 0 |

Objekt lesen

Zuerst sendet der Master eine Aufforderung des gewünschten Objekts. Dann sendet der Drehgeber den geforderten Wert.

Anfrage vom Master:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 40h | 04h | 60h | 0h | x | x | x | x |

Antwort (dcba) des Drehgebers auf die Anfrage:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 43h | 04h | 60h | 0h | a | b | c | d |

Inbetriebnahme

Wenn der Drehgeber an den Bus angeschlossen wird, meldet er sich mit einer BootUp-Meldung. Nun muss der Drehgeber an seine Umgebung angepasst und konfiguriert werden.

Node-ID und Baudrate ändern mit LSS

Node-ID und Baudrate können geändert werden, ohne den Drehgeber über diese anzusprechen zu müssen. Mit dem LSS-Dienst werden die Sensoren über ProductCode, RevisionNr, VendorID und Seriennummer angesprochen und konfiguriert.

Node-ID (Knotennr.) ändern

Die Node-ID kann im Objekt 2101h zwischen 1 und 127 geändert werden. Anschliessend sollte ein Speichervorgang mittels Objekt 1010h durchgeführt werden. Beim nächsten Initialisieren meldet sich der Drehgeber mit der neuen Node-ID an.

Baudrate ändern

Die Baudrate kann im Objekt 2100h geändert werden. Es wird ein Index ins Objekt geschrieben, nicht die effektive Baudrate:

| | Baudrate |
|---|------------|
| 0 | 10 kBaud |
| 1 | 20 kBaud |
| 2 | 50 kBaud |
| 3 | 100 kBaud |
| 4 | 125 kBaud |
| 5 | 250 kBaud |
| 6 | 500 kBaud |
| 7 | 800 kBaud |
| 8 | 1000 kBaud |

Nun muss die Baudrate noch über Objekt 1010-1 gespeichert werden. Beim nächsten Initialisieren meldet sich der Drehgeber auf der neuen Baudrate an. Vorher sollte man aber noch die Baudrate des Masters ändern.

6.2. Konfiguration

Position setzen

Wert wird gesendet:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 23h | 03h | 60h | 0h | a | b | c | d |

Bestätigung:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 60h | 03h | 60h | 0h | 0 | 0 | 0 | 0 |

Drehrichtung und Skalierung ändern

Die Drehrichtung kann auf CW (Uhrzeigersinn) oder CCW (Gegenuhrzeigersinn) eingestellt werden. Zusätzlich wird im gleichen Objekt (6000h) die Skalierung ein oder ausgeschaltet. Bei eingeschalteter Skalierung werden die eingestellten Auflösungen verwendet. Wird jedoch die Skalierung ausgeschaltet, arbeitet der Drehgeber mit den maximalen Auflösungen (6501h und 6502h).

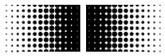
Bit 0: 0 -> CW (Uhrzeigersinn) Wert: 0
 1 -> CCW (Gegenuhrzeigersinn) Wert: 1
 Bit 2: 0 -> Skalierung aus Wert: 0
 1 -> Skalierung ein Wert: 4

Gegenuhrzeigersinn und Skalierung ein:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 23h | 00h | 60h | 0h | 5h | x | x | x |

Bestätigung:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 60h | 00h | 60h | 0h | 0 | 0 | 0 | 0 |



Singleturnauflösung ändern

Im Objekt 6001h kann die Singleturnauflösung konfiguriert werden. Zum Beispiel 1024 (10bit) Schritte pro Umdrehung (1024 = 400h):

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 23h | 01h | 60h | 0h | 00 | 04 | 00 | 00 |

Bestätigung:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 60h | 01h | 60h | 0h | 0 | 0 | 0 | 0 |

Gesamtauflösung ändern

Im Objekt 6002h kann die Gesamtauflösung eingestellt werden. Aus der Gesamtauflösung und der Singleturnauflösung resultiert die Anzahl Umdrehungen. Beispiel: Die Singleturnauflösung beträgt 10 Bit (1024 Schritte) und die Gesamtauflösung wird auf 22 Bit (4194304) eingestellt, somit ergeben sich 4096 (12bit) Umdrehungen à 1024 (10bit) Schritte.

Gesamtauflösung auf 4194304 einstellen (4194304 = 400000h)

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 23h | 02h | 60h | 0h | 00 | 00 | 40 | 00 |

Bestätigung:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 60h | 02h | 60h | 0h | 0 | 0 | 0 | 0 |

Einstellungen ins EEPROM speichern

Über das Objekt 1010h wird das Speichern untenstehender Objekte in den nichtflüchtigen Speicher (EEPROM) ausgelöst. Um ein unabsichtliches Speichern zu verhindern muss die Botschaft "save" in den Subindex 1 geschrieben werden.

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 23h | 10h | 10h | 01h | 73 's' | 61 'a' | 76 'v' | 65 'e' |

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 60h | 10h | 10h | 01h | 0 | 0 | 0 | 0 |

6.3. Betrieb

NMT-Zustände

Nachdem der Drehgeber initialisiert wurde, befindet er sich im **Pre-Operational Mode**. In diesem Zustand können SDO gelesen und geschrieben werden.

Um die PDO-Kommunikation noch zu starten, müssen Sie einen **NMT-Start** senden. Dann befindet sich der Drehgeber im **Operational Mode**. Nun werden gewünschte PDO's gesendet. Zudem können SDO gelesen und geschrieben werden.

Wenn der Drehgeber mit einem **NMT-Stop** gestoppt wird, befindet sich der Drehgeber im **Stopped Mode**. In diesem Zustand ist nur noch NMT-Kommunikation möglich, also auch Heartbeat.

Durch einen **NMT-Reset** wird der Drehgeber wieder initialisiert und befindet sich erneut im **Pre-Operational Mode**.

Position lesen

Anfrage vom Master:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 40h | 04h | 60h | 0 | 0 | 0 | 0 | 0 |

Antwort (dcba) des Drehgebers auf die Anfrage:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 43h | 04h | 60h | 0 | a | b | c | d |

PDO's konfigurieren

Nach folgender Tabelle können die PDO's konfiguriert werden:

| 1800h | | 2800h | Kurzbeschreibung |
|-------|------|-------|-----------------------------------------------------------------------|
| Sub2 | Sub5 | | |
| FEh | 3ms | 0 | Zyklisches Senden alle 3ms |
| FEh | 5ms | 2 | Alle 5ms wird das PDO doppelt gesendet, falls eine Änderung vorliegt. |
| FEh | 0ms | xxx | PDO senden ausgeschaltet |
| 3 | xxx | 0 | Bei jedem dritten Sync-Telegramm senden |
| 3 | xxx | 2Bh | Auf jedes dritte Sync-Telegramm, aber gesamthaft nur 43mal (=2Bh). |

Heartbeat Time festlegen

Um die Kommunikationsfähigkeit zu überwachen muss im Objekt 1017h "Producer Heartbeat Time" die Zeit des Herzschlages definiert werden. Sobald der Wert bestätigt wurde, beginnt der Dienst zu senden. Beispiel: Alle 100ms soll der Drehgeber einen Heartbeat senden (100 = 64h):

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 |
|--------------|-----|----------|----------|----------|----------|--------|--------|
| 600h+Node-ID | 8 | 2Bh | 17h | 10h | 0h | 64h | 0h |

Bestätigung:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 |
|--------------|-----|----------|----------|----------|----------|--------|--------|
| 580h+Node-ID | 8 | 60h | 17h | 10h | 0h | 0 | 0 |

| COB-ID | Data/ Remote | Byte 0 |
|--------|--------------|--------|
| 701h | d | 7Fh |

Die Heartbeat-Meldungen bestehen aus der COB-ID und einem Byte. In diesem Byte wird der NMT-Zustand überliefert.

- 0: BootUp-Event
- 4: Stopped
- 5: Operational
- 127: Pre-Operational

D.h. der Drehgeber befindet sich im Pre-Operational Modus (7Fh = 127).

7. Anschlussbelegung und Inbetriebnahme der Bushaube

7.1. Mechanischer Anbau

Wellen-Drehgeber

- Drehbergerhäuse an den Befestigungsbohrungen flanschseitig mit drei Schrauben (quadratischer Flansch mit 4 Schrauben) montieren. Gewindedurchmesser und Gewindetiefe beachten.
- Alternativ kann der Drehgeber mit Befestigungsexzentern in jeder Winkelposition montiert werden, siehe Zubehör.
- Antriebswelle und Drehgeberwelle über eine geeignete Kupplung verbinden. Die Wellenenden dürfen sich nicht berühren. Die Kupplung muss Verschiebungen durch Temperatur und mechanisches Spiel ausgleichen. Zulässige axiale oder radiale Achsbelastung beachten. Geeignete Verbindungen siehe Zubehör.
- Befestigungsschrauben fest anziehen.

Hohlwellen-Drehgeber

- Klemmringbefestigung
Drehgeber auf die Antriebswelle vollständig aufstecken und den Klemmring fest anziehen.
- Drehmomentstift des Drehgebers
Drehgeber über die Antriebswelle schieben und Drehmomentstift in das kundenseitige Justierelement einführen.
- Justierteil mit Gummifederelement
Drehgeber über die Antriebswelle schieben und Zylinderstift in das kundenseitig montierte Justierteil (mit Gummifederelement) einführen.
- Justierwinkel
Drehgeber über die Antriebswelle schieben. Justierwinkel in Gummifederelement des Drehgebers einführen und den Justierwinkel kundenseitig an der Anlagefläche befestigen.
- Ansatzschraube
Drehgeber über die Antriebswelle schieben und kundenseitig montierte Ansatzschraube in Gummifederelement des Drehgebers einführen.
- Kupplungsfeder
Kupplungsfeder mit den Schrauben an den Befestigungslöchern des Drehgeber-Gehäuses montieren. Drehgeber über die Antriebswelle schieben und Kupplungsfeder an der Anlagefläche befestigen.

7.2. Elektrischer Anschluss

Bushaube ausschliesslich im ESD Beutel lagern und transportieren. Bushaube muss vollständig am Gehäuse anliegen und fest verschraubt sein.

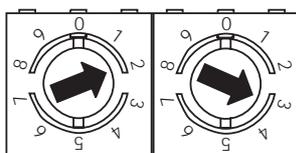
Zum elektrischen Anschluss Bushaube folgendermassen abziehen:

- Beide Befestigungsschrauben der Bushaube lösen
- Bushaube vorsichtig lockern und axial abziehen

7.2.1. Teilnehmeradresse (Node ID) einstellen

Die Einstellung der Teilnehmeradresse erfolgt über das EEPROM. Die Node-ID (Teilnehmeradresse) wird im Objekt 2101h definiert. Zusätzlich gibt es die Möglichkeit, über zwei Drehschalter in der Bushaube die Teilnehmeradresse dezimal einzustellen. Wenn die Drehschalter auf 0 stehen, wird die Node-ID aus dem EEPROM verwendet. Sobald die Schalter auf einen Wert eingestellt sind, wird dieser eingestellte Wert als Teilnehmeradresse verwendet. Die maximale Teilnehmerzahl ist 99.

- Teilnehmeradresse dezimal mit beiden Drehschaltern 1 und 2 einstellen (Werkseinstellung 01).



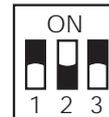
Beispiel: 23

7.2.2. Baudrate einstellen

Die Baudrate ist im Objekt 2100h festgelegt. Zusätzlich gibt es auch hier die Möglichkeit, die Baudrate mittels DIP-Schalter einzustellen. Die Einstellung der Baudrate erfolgt binär über Schalter 1 bis 3 des 3-poligen DIP Schalters in der Bushaube. Die verwendete Baudrate aus dem EEPROM wird ignoriert, sobald die Drehschalter für die Teilnehmeradresse nicht auf 0 eingestellt sind.

| Baudrate | Einstellung DIP Schalter | | |
|-------------|--------------------------|-----|-----|
| | 1 | 2 | 3 |
| 10 kBit/s | OFF | OFF | OFF |
| 20 kBit/s | OFF | OFF | ON |
| 50 kBit/s * | OFF | ON | OFF |
| 125 kBit/s | OFF | ON | ON |
| 250 kBit/s | ON | OFF | OFF |
| 500 kBit/s | ON | OFF | ON |
| 800 kBit/s | ON | ON | OFF |
| 1 MBit/s | ON | ON | ON |

* Werkseinstellung:



7.2.3. Abschlusswiderstand

Ist der angeschlossene Drehgeber das letzte Gerät in der Busleitung, muss der Bus mit einem Widerstand abgeschlossen werden. Der Widerstand ist in der Bushaube und wird über den einpoligen DIP Schalter zugeschaltet. Abschlusswiderstand muss beim letzten Teilnehmer mit dem 1-poligen DIP-Schalter auf „ON“ geschaltet werden (Werkseinstellung OFF).



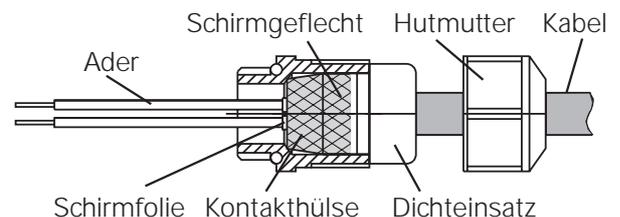
ON = Letzter Teilnehmer
OFF = Teilnehmer X



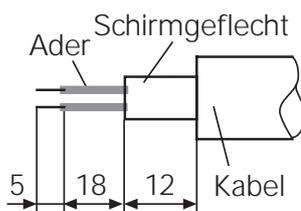
Schalter 1: ON = Letzter Teilnehmer
OFF = Teilnehmer X
Schalter 2: ohne Funktion

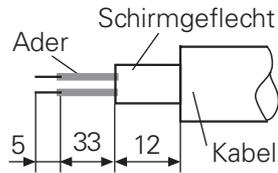
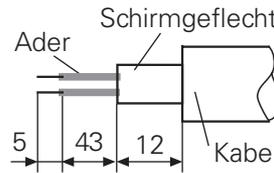
7.2.4. Anschluss Bushaube

- Hutmutter der Kabelverschraubung lösen
- Hutmutter und Dichteinsatz mit Kontakthülse auf den Kabelmantel schieben.
- Kabelmantel und Adern abisolieren, Schirmfolie, falls vorhanden, kürzen (s. Bild)
- Schirmgeflecht um ca. 90° umbiegen
- Dichteinsatz mit Kontakthülse bis an das Schirmgeflecht schieben. Dichteinsatz mit Kontakthülse und Kabel bündig in die Kabelverschraubung einführen und Hutmutter verschrauben



Für Standard Drehgeber

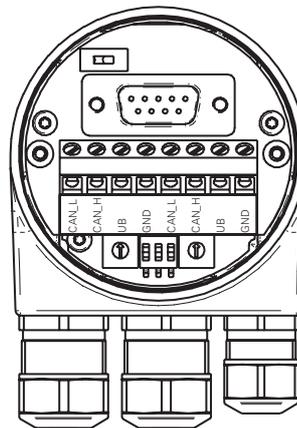


Für BISD und BIMD
Busleitung

Betriebsspannungsleitung


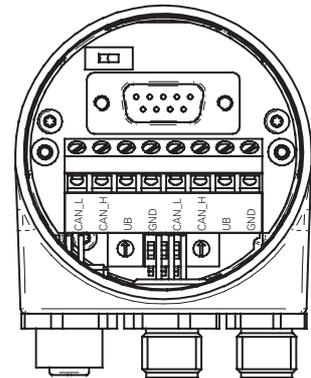
- Klemmen mit gleicher Bezeichnung sind intern miteinander verbunden.
- Für die Betriebsspannung ausschliesslich Kabelverschraubung 3 verwenden. Für die Busleitungen können frei wählbar Kabelverschraubung 1 oder 2 verwendet werden. Zulässige Kabelquerschnitte beachten.
- Adern auf dem kürzesten Weg von der Kabelverschraubung an die Klemmleiste einführen. Zulässiger Aderquerschnitt beachten. Isolierte Aderendhülsen verwenden.
- Überkreuzungen der Datenleitungen mit der Leitung der Betriebsspannung muss vermieden werden.

Bushaube – axial


1 2 3

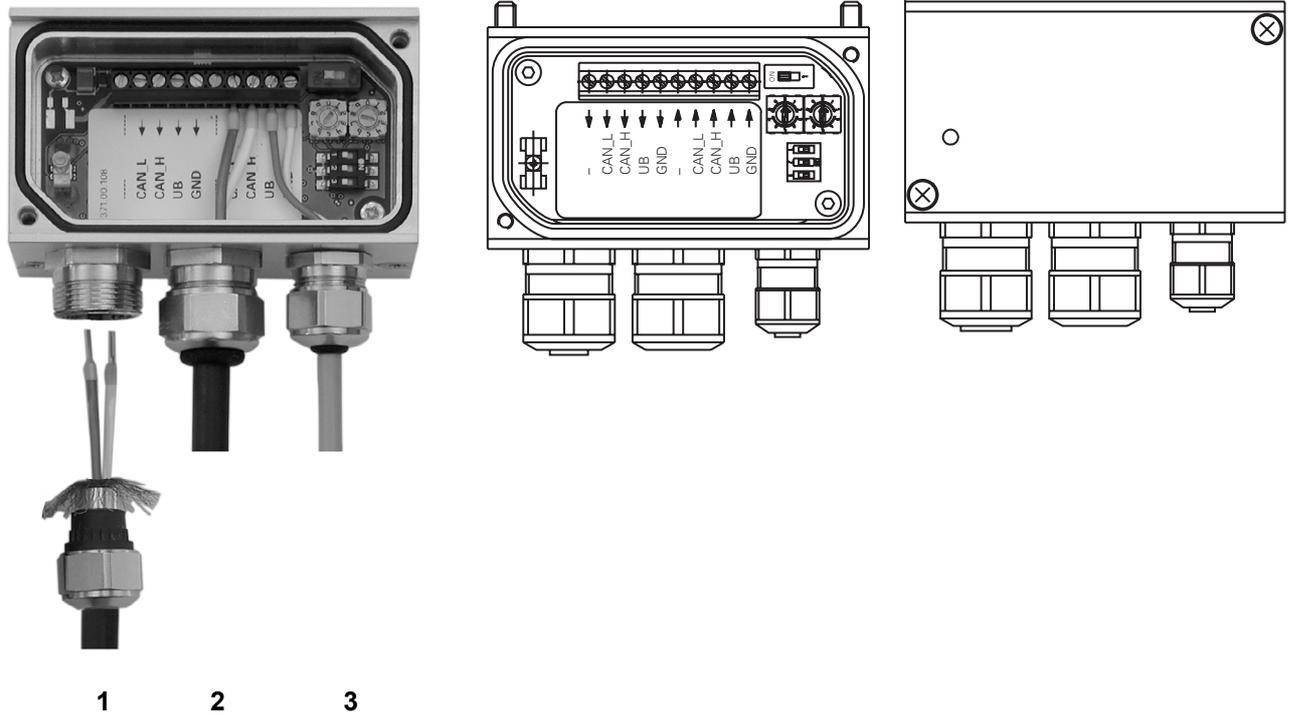


Kabelverschraubung

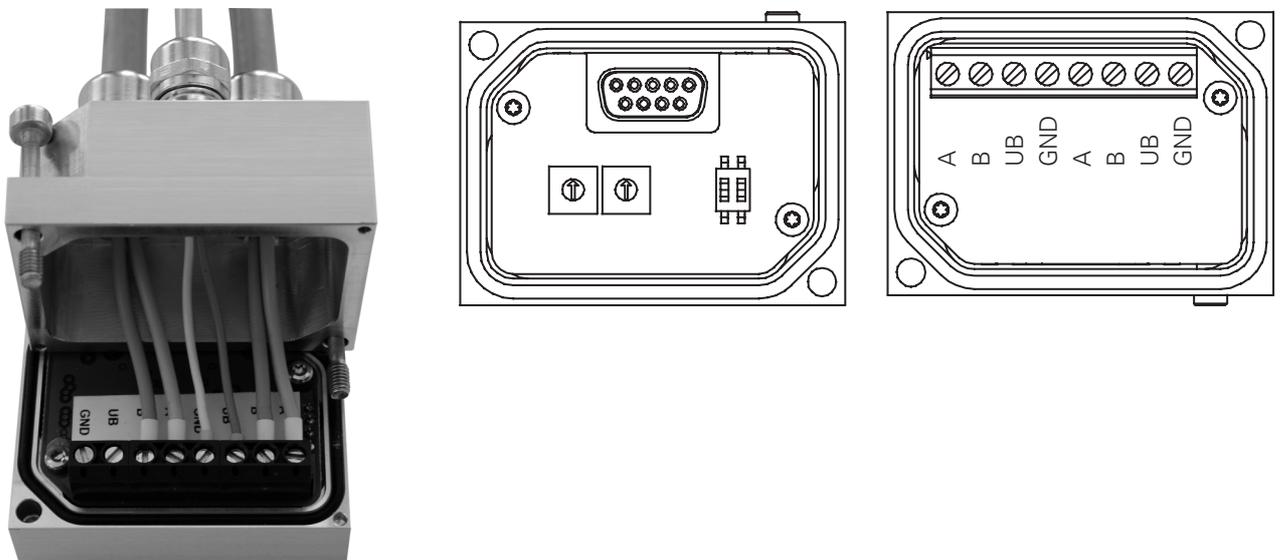


M12-Stecker

Bushaube – radial



Bushaube – radial (für BISS und BIMD)



7.2.5. Anschlussbelegung der Klemmen

| Klemme | Erklärung |
|--------|--------------------------------|
| CAN_L | CAN Bus Signal (dominant Low) |
| CAN_H | CAN Bus Signal (dominant High) |
| UB | Betriebsspannung 10...30 VDC |
| GND | Masseanschluss für UB |

Klemmen mit gleicher Bezeichnung sind intern miteinander verbunden

- Bushaube vorsichtig auf den D-SUB Stecker vom Basisgeber aufstecken, dann erst über den Dichtgummi drücken und nicht verkanten. Bushaube muss vollständig am Basisgeber anliegen.
- Beide Befestigungsschrauben gleichsinnig fest anziehen.

Drehgebergehäuse und Schirmgeflecht des Anschlusskabels sind nur dann optimal verbunden, wenn die Bushaube vollständig auf dem Basisgeber aufliegt (Formschluss).

7.3. Anzeigeelemente

Auf der Rückseite der Bushaube ist eine DUO-LED integriert.

| LED grün | LED rot | Status |
|----------|---------|--------------------------------------|
| aus | aus | Betriebsspannung nicht angeschlossen |
| blinkt | aus | Preoperational Mode |
| ein | aus | Operational Mode |
| ein | ein | Stopped/Prepared Mode |
| aus | blinkt | Warning |
| aus | ein | Error |

8. Anschluss und Inbetriebnahme bei integrierter Feldbusschnittstelle

8.1.1. Teilnehmeradresse und Baudrate einstellen

Die Node-ID kann im Objekt 2101h zwischen 1 und 127 geändert werden

Node-ID auf 23h einstellen:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 2Fh | 01h | 21h | 0h | 23h | Xx | Xx | Xx |

Bestätigung:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 60h | 01h | 21h | 0h | 0h | 0h | 0h | 0h |

Die Baudrate kann im Objekt 2100h geändert werden. Es wird ein Index ins Objekt geschrieben, nicht die effektive Baudrate:

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

Baudrate auf 250 kBit/s (5h) einstellen:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 2Fh | 00h | 21h | 0h | 05h | Xx | Xx | Xx |

Bestätigung:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 60h | 00h | 21h | 0h | 0h | 0h | 0h | 0h |

Nun müssen die neuen Einstellungen noch gespeichert werden. Über das Objekt 1010h wird das Speichern untenstehender Objekte in den nichtflüchtigen Speicher (EEPROM) ausgelöst. Um ein unabsichtliches Speichern zu verhindern muss die Botschaft "save" in den Subindex 1 geschrieben werden.

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 600h+Node-ID | 8 | 23h | 10h | 10h | 01h | 73 's' | 61 'a' | 76 'v' | 65 'e' |

Bestätigung:

| COB-ID | DLC | Kommando | Objekt L | Objekt H | Subindex | Data 0 | Data 1 | Data 2 | Data 3 |
|--------------|-----|----------|----------|----------|----------|--------|--------|--------|--------|
| 580h+Node-ID | 8 | 60h | 10h | 10h | 01h | 0h | 0h | 0h | 0h |

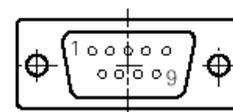
Beim nächsten Initialisieren meldet sich der Drehgeber auf der neuen Baudrate mit der neuen Knotennummer an. Vorher sollte man aber noch die Baudrate des Masters ändern.

8.1.2. Abschlusswiderstand

Um einen störungsfreien Betrieb mit hohen Datenübertragungsraten bis 1Mbit/s zu gewährleisten, ist es zwingend den Bus richtig abzuschliessen. Anfang und Ende des Busses müssen mit einem Widerstand (120Ω , $\frac{1}{4} W$) abgeschlossen werden.

8.1.3. Anschlussbelegung D-SUB-Stecker

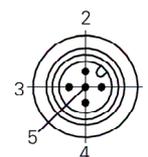
| Pin Nr. | Signal | Beschreibung | Kabelfarbe |
|---------|---------|--------------------------------|------------|
| 1 | | n.c. | |
| 2 | CAN_L | CAN Bus-Signal (dominant Low) | gelb |
| 3 | CAN_GND | Masseanschluss Bus-Signale | grau |
| 4 | | n.c. | |
| 5 | | n.c. | |
| 6 | 0V | Betriebsspannung | weiss |
| 7 | CAN_H | CAN Bus-Signal (dominant High) | grün |
| 8 | | n.c. | |
| 9 | +Vs | Betriebsspannung 10...30 VDC | braun |



Sicht auf Drehgeber

8.1.4. Anschlussbelegung Stecker M12

| Pin Nr. | Signal | Beschreibung | |
|---------|---------|----------------------------------|--|
| 1 | | n.c. | |
| 2 | +Vs | Betriebsspannung 10...30 VDC | |
| 3 | CAN_GND | CAN Ground / 0V Betriebsspannung | |
| 4 | CAN_H | CAN Bus-Signal (dominant High) | |
| 5 | CAN_L | CAN Bus-Signal (dominant Low) | |



Sicht auf Drehgeber