

Handbuch

Absolute Drehgeber mit CANopen (mit Bushaube)

Firmware Version ab 1.00

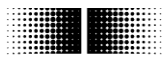
Baumer IVO GmbH & Co. KG
Dauchinger Strasse 58-62
DE-78056 Villingen-Schwenningen
Phone +49 7720 942-0
Fax +49 7720 942-900
info.de@baumerivo.com
www.baumer.com

10.17 · 174.01.022/11
Irrtum sowie Änderungen in
Technik und Design vorbehalten.



Inhalt

| | Seite |
|--|-----------|
| 1. Einleitung | 3 |
| 1.1. Lieferumfang | 3 |
| 1.2. Produktzuordnung | 4 |
| 2. Sicherheits- und Betriebshinweise | 5 |
| 3. Produktfamilie | 6 |
| 4. CAN-Bus und CANopen-Kommunikation | 7 |
| 4.1. CAN-Bus | 7 |
| 4.1.1. CAN-Bus-Eigenschaften | 7 |
| 4.2. CANopen | 8 |
| 4.3. CANopen-Kommunikation | 9 |
| 4.3.1. Kommunikationsprofil | 9 |
| 4.3.2. CANopen Meldungsaufbau | 9 |
| 4.3.3. Servicedaten-Kommunikation | 10 |
| 4.3.4. Prozessdaten-Kommunikation | 11 |
| 4.3.5. Emergency-Dienst | 13 |
| 4.3.6. Netzwerkmanagement-Dienste | 14 |
| 4.3.7. Layer Setting Services | 18 |
| 4.4. Drehgeber Profil | 21 |
| 4.4.1. Drehgeber-Objekte Übersicht | 21 |
| 4.4.2. Ausführliche Objektliste (DS-301) | 26 |
| 5. Diagnose und Wissenswertes | 41 |
| 5.1. Fehlerdiagnose Feldbus-Kommunikation | 41 |
| 5.2. Fehlerdiagnose über Feldbus | 41 |
| 5.3. Wissenswertes zum Sensor | 42 |
| 6. Applikationen | 43 |
| 6.1. Objekte setzen und lesen | 43 |
| 6.2. Konfiguration | 44 |
| 6.3. Betrieb | 46 |
| 6.4. Inbetriebnahme über CAN | 48 |
| 7. Anschlussbelegung und Inbetriebnahme | 50 |
| 7.1. Mechanischer Anbau | 50 |
| 7.2. Elektrischer Anschluss | 50 |
| 7.2.1. Teilnehmeradresse (Node ID) einstellen | 50 |
| 7.2.2. Baudrate einstellen | 51 |
| 7.2.3. Abschlusswiderstand | 51 |
| 7.2.4. Anschluss Bushaube | 51 |
| 7.2.5. Anschlussbelegung | 54 |
| 7.3. Anzeigeelemente (Statusanzeige) | 54 |



Haftungsausschluss

Diese Schrift wurde mit grosser Sorgfalt zusammengestellt. Fehler lassen sich jedoch nicht immer vollständig ausschliessen. Baumer IVO GmbH & Co. KG übernimmt daher keine Garantien irgendwelcher Art für die in dieser Schrift zusammengestellten Informationen. In keinem Fall haftet Baumer IVO GmbH & Co. KG oder der Autor für irgendwelche direkten oder indirekten Schäden, die aus der Anwendung dieser Informationen folgen.

Wir freuen uns jederzeit über Anregungen, die der Verbesserung dieses Handbuchs dienen können.

Created by:

Baumer IVO GmbH & Co. KG

Villingen-Schwenningen, Germany

1. Einleitung

1.1. Lieferumfang

Bitte prüfen Sie vor der Inbetriebnahme die Vollständigkeit der Lieferung.

Je nach Ausführung und Bestellung können zum Lieferumfang gehören:

Basisgeber, Bushaube, CD mit Beschreibungsdateien und Handbuch (auch zum Download verfügbar).



1.2. Produktzuordnung

Wellen-Drehgeber

| Produkt | Produkt-Code | Device Name | Eds-Datei | Produktfamilie |
|---------|--------------|-------------|-------------|----------------------------------|
| GBAMW | 0x0F | GBAM | GBAMW_H.eds | multivoPlus - Singleturn |
| GBMMW | 0x0E | GBMM | GBMMW_H.eds | multivoPlus - Multiturn |
| GBLMW | 0x0E | GBMM | GBMMW_H.eds | multivoPlus - Multiturn |
| GCAMW | 0x0D | GCAM | GCAMW_H.eds | magtivo® - Singleturn |
| GCMMW | 0x0C | GCMM | GCMMW_H.eds | magtivo® - Multiturn |
| GEMMW | 0x0A | GXMM | GXMMW_H.eds | multivo® - Multiturn (Edelstahl) |
| GXAMW | 0x0B | GXAM | GXAMW_H.eds | multivo® - Singleturn |
| GXMMW | 0x0A | GXMM | GXMMW_H.eds | multivo® - Multiturn |
| GXMLW | 0x0A | GXMM | GXMMW_H.eds | multivo® - Multiturn |

Endwellen-Drehgeber

| Produkt | Produkt-Code | Device Name | Eds-Datei | Produktfamilie |
|---------|--------------|-------------|-------------|--------------------------|
| GBAMS | 0x0F | GBAM | GBAMW_H.eds | multivoPlus - Singleturn |
| GBMMS | 0x0E | GBMM | GBMMW_H.eds | multivoPlus - Multiturn |
| GBLMS | 0x0E | GBMM | GBMMW_H.eds | multivoPlus - Multiturn |
| GCAMS | 0x0D | GCAM | GCAMW_H.eds | magtivo® - Singleturn |
| GCMMS | 0x0C | GCMM | GCMMW_H.eds | magtivo® - Multiturn |
| GXAMS | 0x0B | GXAM | GXAMW_H.eds | multivo® - Singleturn |
| GXMMS | 0x0A | GXMM | GXMMW_H.eds | multivo® - Multiturn |
| GXLMS | 0x0A | GXMM | GXMMW_H.eds | multivo® - Multiturn |

Hohlwellen-Drehgeber

| Produkt | Produkt-Code | Device Name | Eds-Datei | Produktfamilie |
|---------|--------------|-------------|-------------|----------------------------------|
| G0AMH | 0x0B | GXAM | GXAMW_H.eds | multivo® - Singleturn |
| G0MMH | 0x0A | GXMM | GXMMW_H.eds | multivo® - Multiturn |
| G0LMH | 0x0A | GXMM | GXMMW_H.eds | multivo® - Multiturn |
| G1AMH | 0x0B | GXAM | GXAMW_H.eds | multivo® - Singleturn |
| G1MMH | 0x0A | GXMM | GXMMW_H.eds | multivo® - Multiturn |
| G2AMH | 0x0B | GXAM | GXAMW_H.eds | multivo® - Singleturn |
| G2MMH | 0x0A | GXMM | GXMMW_H.eds | multivo® - Multiturn |
| GBAMH | 0x0F | GBAM | GBAMW_H.eds | multivoPlus - Singleturn |
| GBLMH | 0x0E | GBMM | GBMMW_H.eds | multivoPlus - Multiturn |
| GBMMH | 0x0E | GBMM | GBMMW_H.eds | multivoPlus - Multiturn |
| GEMMH | 0x0A | GXMM | GXMMW_H.eds | multivo® - Multiturn (Edelstahl) |



2. Sicherheits- und Betriebshinweise

Bestimmungsgemässer Gebrauch

- Der Drehgeber ist ein Präzisionsmessgerät, das der Erfassung von Positionen und/oder Geschwindigkeiten dient. Er liefert Messwerte als elektronische Ausgangssignale für das Folgegerät. Er darf nur zu diesem Zweck verwendet werden. Sofern dieses Produkt nicht speziell gekennzeichnet ist, darf es nicht für den Betrieb in explosionsgefährdeter Umgebung eingesetzt werden.
- Eine Gefährdung von Personen, eine Beschädigung der Anlage oder von Betriebseinrichtungen durch den Ausfall oder Fehlfunktion des Drehgebers muss durch geeignete Sicherheitsmassnahmen ausgeschlossen werden.

Qualifikation des Personals

- Einbau und Montage des Drehgebers darf ausschliesslich durch eine Fachkraft für Elektrik und Feinmechanik erfolgen.
- Betriebsanleitung des Maschinenherstellers ist zu beachten.

Wartung

- Der Drehgeber ist wartungsfrei und darf nicht geöffnet beziehungsweise mechanisch oder elektrisch verändert werden. Ein Öffnen des Drehgebers kann zu Verletzungen führen.

Entsorgung

- Der Drehgeber enthält elektronische Bauelemente und je nach Typ eine Batterie. Bei einer Entsorgung müssen die örtlichen Umweltrichtlinien beachtet werden.

Montage

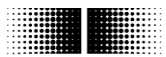
- Vollwelle: Keine starre Verbindung von Drehgeberwelle und Antriebswelle vornehmen. Antriebs- und Drehgeberwelle über eine geeignete Kupplung verbinden.
- Hohlwelle: Vor Montage des Drehgebers, Klemmring vollständig öffnen. Fremdkörper sind in ausreichendem Abstand zur Statorkupplung zu halten. Die Statorkupplung darf ausser an den Befestigungspunkten des Drehgebers und der Maschine nicht anstehen.

Elektrische Inbetriebnahme

- Keine Verdrahtungsarbeiten unter Spannung vornehmen
- Den elektrischen Anschluss unter Spannung nicht aufstecken oder entfernen
- Die gesamte Anlage EMV-gerecht installieren. Einbauumgebung 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 Schutzterde (PE) anschliessen. Geschirmte Kabel, auch für die Stromversorgung, verwenden. Schirmgeflecht muss mit der Kabelverschraubung oder Stecker verbunden sein. Anzustreben ist ein beidseitiger Anschluss an Schutzterde (PE), Gehäuse über den mechanischen Anbau, Kabelschirm über die nachfolgenden angeschlossenen Geräte.

Zusätzliche Informationen

- Das Handbuch ist eine Ergänzung zu weiteren Dokumentationen (z.B. Katalog, Datenblatt oder Montageanleitung).



3. Produktfamilie

Die Produktfamilie ist modular aufgebaut. Basis-Drehgeber und Bushauben können je nach Anforderungen an den Drehgeber und nach dem gewählten Bussystem beliebig kombiniert werden. Die Basis-Drehgeber unterscheiden sich in Auflösung, Genauigkeit, den Umgebungsbedingungen und dem Abtastsystem.

Bushaube

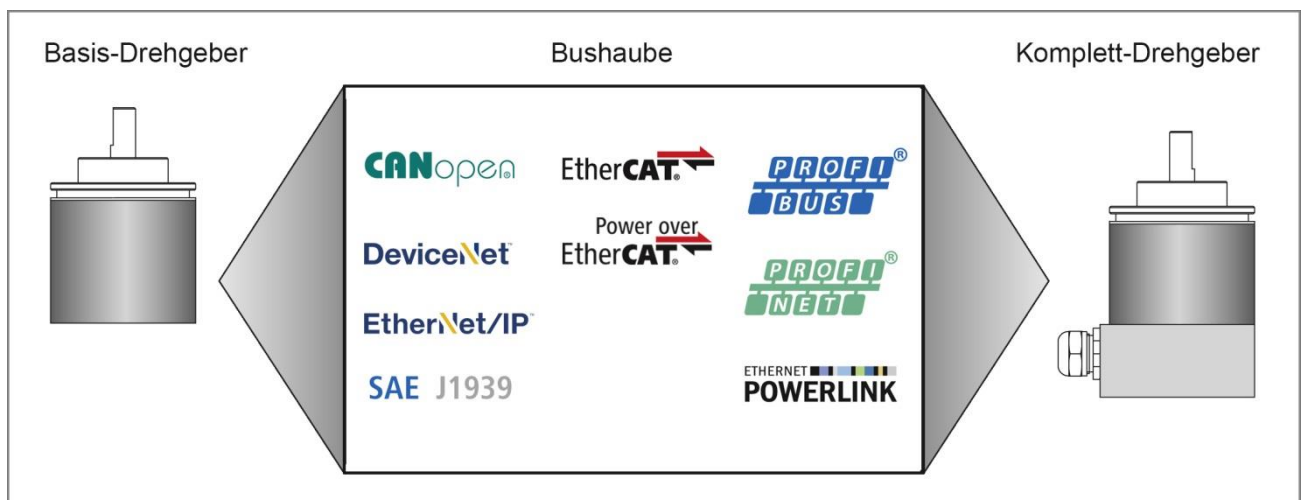
In der Bushaube ist die gesamte Elektronik der Messwertaufbereitung und des Kommunikationssystems (Feldbus oder Realtime-Ethernet) integriert.

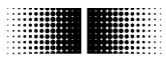
Die Bushauben unterscheiden sich durch die jeweiligen Bus-Schnittstellen.

Schnittstellen sind: CANopen®, DeviceNet, EtherCAT, Ethernet/IP, Profibus-DP, Profinet, Powerlink, Power over EtherCAT, SAE J1939, SSI.

Alle Drehgeber sind über die Bus-Schnittstelle parametrierbar.

Funktionsprinzip:





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.

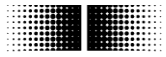
4.1.1. CAN-Bus-Eigenschaften

- Datenrate von 1 MBit/s bei einer Netzausdehnung bis zu 40 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 Nachrichtennummer (Identifier) gekennzeichnet. Alle Netzknoten prüfen anhand des Identifier, 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 selbstständig 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 Identifier 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 Netzwerk-Teilnehmern 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 geeigneten Funktionen enthalten. CANopen stellt somit eine für die Anwendung optimierte Teilmenge von CAL dar und ermöglicht dadurch vereinfachten Systemaufbau und den Einsatz vereinfachter 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, Daten-Typ, 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 | xx | xx | xx | xx | xx | xx | 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 5..8 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, 5 mal 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 3 ms |
| Feh | 5ms | 2 | Alle 5ms wird das PDO doppelt gesendet, falls eine Änderung vorliegt. |
| Feh | 0ms | 0 | PDO senden ausgeschaltet |
| 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 | | - |

Byte 0..1: Error Codes

| Error Code (hex) | Meaning |
|------------------|---|
| 0000 | Error Reset or No Error |
| 1000 | Generic Error |
| 5530 | EEProm error (ab V1.04) |
| 6010 | Software reset (Watchdog) (ab V1.04) |
| 7320 | Position error (ab V1.04) |
| 7510 | Internal communication error (ab V1.04) |
| 8130 | Life Guard error or Hearbeat error (ab V1.04) |
| FF00 | Battery low (ab V1.04) |

Byte 2: Error-Register

| Bit | Meaning |
|-----|----------------------------------|
| 0 | Generic Error |
| 4 | Communication error (ab V1.04) |
| 7 | manufacturer specific (ab V1.04) |

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 verwendet



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.

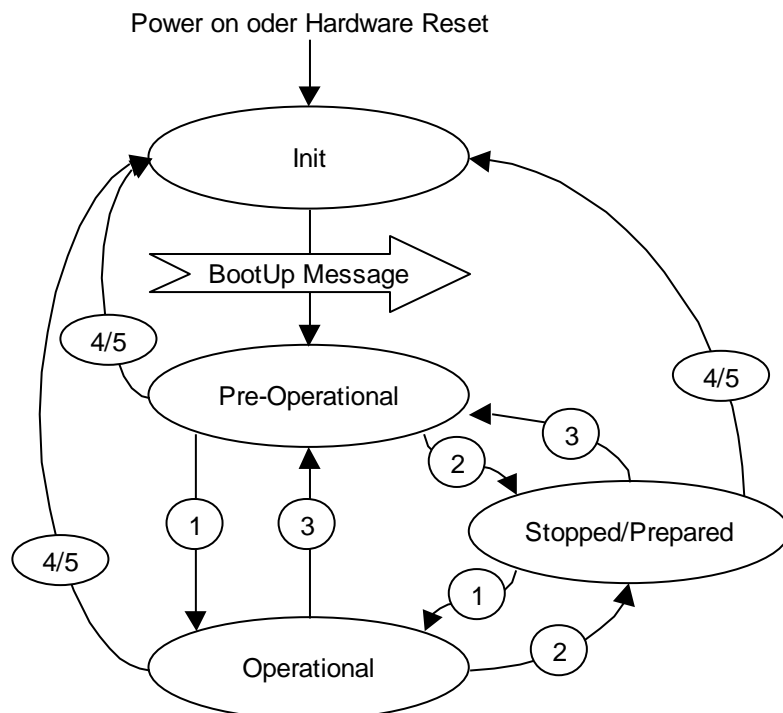
Kommando Byte

| 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 CAN-Bus. Danach geht der Drehgeber automatisch in den Zustand PreOperational 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 Reset-Befehl wird der Drehgeber neu initialisiert.
Reset Remote Node (4):

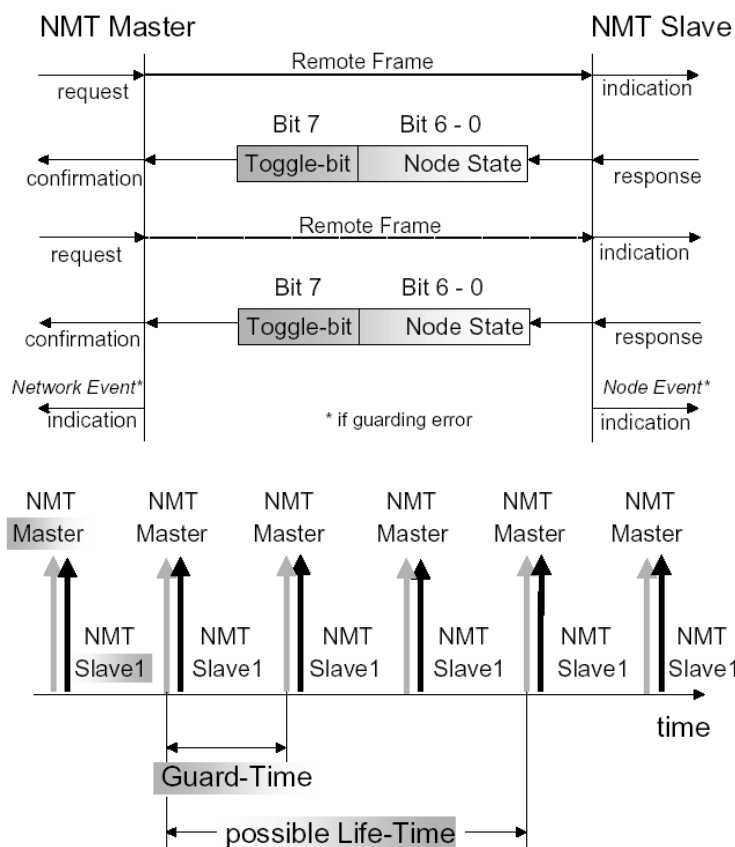
| COB-ID | Kommando Byte | Knoten Nummer |
|--------|---------------|---------------|
| 0 | 81h | 0..127 |

Reset Kommunikation (5):

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



Node und Life Guarding



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 Überwachungs-dienst mit einem Remote-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 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) |

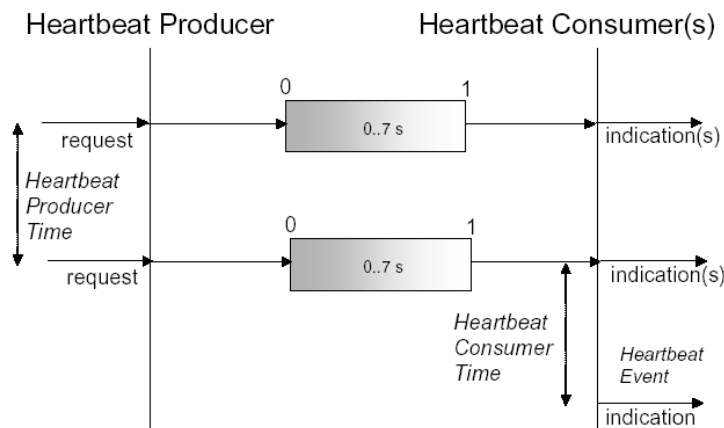
Mögliche NMT-Zustände der Teilnehmer:

- 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.



Heartbeat-Protokoll



Wahlweise kann das neue Heartbeat-Protokoll verwendet werden. Heartbeat ist aktiv, wenn im Objekt 2110h Bit 5 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).

Achtung : Nur einer der beiden oben beschriebenen Knotenüberwachungsmechanismen kann aktiv sein.

Default: Heartbeat
Optional: NodeGuarding (siehe Objekt 2110)



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 : ECh

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 → | 11h | 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 | ProduktCode | 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 unterstützen 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 | "GXMM" "GXAM" "GCMM" "GCAM" "GDMM" "GDAM" "GBMM" "GBAM" | ja | Devicename = "GXMM" multivo Multiturn "GXAM" multivo Singleturn "GCMM" magtivo Multiturn "GCAM" magtivo Singleturn "GDMM" activo Multiturn "GDAM" activo Singleturn "GBMM" multivoPlus Multiturn "GBAM" multivoPlus 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 | | | | |



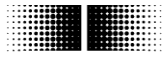
| Objekt Sub-Index | Name | Format | Zugriff | Default | Save | Beschreibung |
|---------------------|----------------------------------|--------|---------|--|------|--|
| 00h | Größter Subindex | U8 | ro | 4h | | |
| 01h | Alle Parameter speichern | U32 | rw | | | =“save“ (0x73617665) zum speichern |
| 02h | Communication Parameters | U32 | rw | | | =“save“ (0x73617665) zum speichern |
| 03h | Application Parameters | U32 | rw | | | =“save“ (0x73617665) zum speichern |
| 04h | Manuf. Specific Parameters | U32 | rw | | | =“save“ (0x73617665) zum speichern |
| 1011h | Restore Default Parameters | ARR | | | | |
| 00h | Größter Subindex | U8 | ro | 4h | | |
| 01h | Alle Parameter | U32 | rw | | | =“load“ (0x6C6F6164) zum laden |
| 02h | Communication Parameters | U32 | rw | | | =“load“ (0x6C6F6164) zum laden |
| 03h | Application Parameters | U32 | rw | | | =“load“ (0x6C6F6164) zum laden |
| 04h | Manufacturer Specific Parameters | U32 | rw | | | =“load“ (0x6C6F6164) 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 | ECh | ja | Von CiA vergebene Vendor ID |
| 02h | Product Code | U32 | ro | 0Ah 0Bh 0Ch 0Dh 0Eh 0Fh | ja | Product Code: 0Ah = multivo Multiturn 0Bh = multivo Singleturn 0Ch = magtivo Multiturn 0Dh = magtivo Singleturn 0Eh = activo/multivoPlus Multiturn 0Fh = activo/multivoPlus Singleturn |
| 03h | Revision Number | U32 | ro | werkseitig | ja | Produkt Revisionsnummer |
| 04h | Serial Number | U32 | ro | werkseitig | ja | Eindeutige fortlaufende Seriennummer |
| 1029h | Error behaviour | ARR | | | | Ab V1.04 |
| 00h | Größter Subindex | U8 | ro | 1h | | |
| 01h | Communication error | U8 | rw | 1h | ja | 0h = Wechsel in den Pre-Operational Mode 1h = kein Mode-Wechsel 2h = Wechsel in den Stop Mode 3h = Knoten reset |
| 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 | | | | |
| 00h | Größter Subindex | U8 | ro | 1h | | |
| 01h | Inhalt des PDO1 | U32 | ro | 60040020h | | |



| Objekt Sub-Index | Name | Format | Zugriff | Default | Save | Beschreibung |
|---------------------|----------------------------|--------|---------|-----------|------|--|
| 1A01h | Transmit PDO2 Mapping | ARR | | | | |
| 00h | Größter Subindex | U8 | ro | 1h | | |
| 01h | Inhalt des PDO2 | U32 | ro | 60040020h | | |
| 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 | Manufacturer_Options | 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 Nodeguarding-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 Bit9 = Verhalten bei schreiben auf Objekt Auflösung/Gesamtauflösung 0 Offset löschen 1 Offset nicht löschen (ab Version V1.08) Bit10 = Verhalten bei Reset Node (ab V1.09) 0 HW Reset 1 Init NMT state |
| 2201h | Statistik | REC | | | | |
| 00h | Größter Subindex | U8 | ro | 3h | | |
| 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 | |



| Objekt Sub-Index | Name | Format | Zugriff | Default | Save | Beschreibung |
|---------------------|-----------------------------------|--------|---------|--|------|---|
| 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 |
| 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 = multivo 12Bit = magtivo 18Bit = activo/multivoPlus |
| 6002h | Gesamtmessbereich in Schritten | U32 | rw | 20000000h 2000h 4000000h 1000h 80000000h 40000h | ja | Gesamtmessbereich in Schritten 29Bit = multivo Multiturn 13Bit = multivo Singleturn 26Bit = magtivo Multiturn 12Bit = magtivo Singleturn 31Bit = activo/multivoPlus Multiturn 18Bit = activo/multivoPlus 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 1000h 40000h | | Max Auflösung in Schritte/Umdrehung: 13Bit = multivo 12Bit = magtivo 18Bit = activo/multivoPlus |
| 6502h | Gesamtmessbereich in Schritten | U32 | ro | 20000000h 2000h 4000000h 1000h 80000000h 40000h | | 29Bit = multivo Multiturn 13Bit = multivo Singleturn 26Bit = magtivo Multiturn 12Bit = magtivo Singleturn 31Bit = activo/multivoPlus Multiturn 18Bit = activo/multivoPlus Singleturn |
| 6503h | Alarme | U16 | ro | 0h | | Folgende Alarme werden ausgewertet: Bit0 = Positions-Error |
| 6504h | Unterstützte Alarme | U16 | ro | 1h | | Folgende Alarme 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 |



| Objekt Sub-Index | Name | Format | Zugriff | Default | Save | Beschreibung |
|---------------------|---------------|--------|---------|------------|------|-------------------------------------|
| 6509h | Offset | U32 | ro | 0h | | Offset aus Preset berechnet → 6003h |
| 650Bh | Serien Nummer | U32 | ro | werkseitig | ja | Seriennummer Objekt 1018h-4h |
| | | | | | | |



4.4.2. Ausführliche Objektliste (DS-301)

Objekt 1000 Device Typ

| | | | | |
|--------------|--|---------------------|-------------|-------|
| SubIndex | 0 | | | |
| Daten-Typ | Unsigned 32 | | | |
| Zugriff | ReadOnly | | | |
| Default | Multiturn: 00020196h Singleturn: 00010196h | | | |
| EEPROM | No | | | |
| Beschreibung | Information über Geräteprofil und Gerätetyp | | | |
| Werte | Multiturn: | | | |
| | Data0 = Profil LOW | Data1 = Profil HIGH | Data2 = Typ | Data3 |
| | 96 | 01 | 02 | 00 |
| | Data 0, 1 = 96h 01h = 0196h = DSP-406 = DeviceProfil für Drehgeber | | | |
| | Data 2, 3 = 02h 00h = Multiturn, absolut | | | |
| | Singleturn: | | | |
| | Data0 = Profil LOW | Data1 = Profil HIGH | Data2 = Typ | Data3 |
| | 96 | 01 | 02 | 00 |
| | Data 0, 1 = 96h 01h = 0196h = DSP-406 = DeviceProfil für Drehgeber | | | |
| | Data 2, 3 = 01h 00h = Singleturn, absolut | | | |

Objekt 1001 Error Register

| | |
|--------------|---|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadOnly |
| Default | 0h |
| EEPROM | No |
| Beschreibung | Aktueller FehlerCode |
| Werte | Bit0 = Generic error Bit4 = Communication error (overrun, ...) Bit7 = Manufacturer specific |

Objekt 1003 Vordefiniertes Error Feld

CiA (CAN in Automation) definiert hier etwa 200 verschiedene Error Code. In diesem Dokument werden nur die für den Sensor relevanten Error Code beschrieben.

Dieses Objekt speichert die letzten 8 aufgetretenen Fehler oder Warnungen.

| | |
|--------------|---|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadWrite |
| Default | 0 |
| EEPROM | No |
| Beschreibung | Lesen: Anzahl Fehler oder Warnungen Schreiben von 0: Fehler zurücksetzen |
| Werte | 0..8 |

| | |
|--------------|--|
| SubIndex | 1..8 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 0 |
| EEPROM | No |
| Beschreibung | Aufgetretene Fehler oder Warnungen wobei SubIndex 1 der letzte, SubIndex 2 der vorletzte, ..., Eintrag ist |
| Werte | Noch nicht definiert |

**Objekt 1005 COB-ID SYNC Message**

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadWrite |
| Default | 80h |
| EEPROM | Yes |
| Beschreibung | Definiert COB-ID des Synchronisations Objekt (SYNC) |
| Werte | Bit 31 not defined Bit 30 1=Sensor generiert SYNC Meldungen, 0=generiert keine SYNC Meldung Bit 29 1=29 Bit SYNC COB-ID (CAN 2.0B), 0=28 Bit SYNC COB-ID (CAN 2.0A) Bit 28..11 Bit 28..11 der 29 Bit SYNC COB-ID Bit 10..0 Bit 10..0 der SYNC COB-ID |

Objekt 1008 Hersteller Device Name

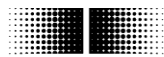
| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | Abhängig vom verwendeten Basisgeber |
| EEPROM | No |
| Beschreibung | GeräteBezeichnung in ASCII |
| Werte | Data 0..3: "GXMM" = 47h 58h 4Dh 4Dh → multivo Multiturn "GXAM" = 47h 58h 41h 4Dh → multivo Singleturn "GCMM" = 47h 43h 4Dh 4Dh → magtivo Multiturn "GCAM" = 47h 43h 41h 4Dh → magtivo Singleturn "GDMM" = 47h 44h 4Dh 4Dh → activo Multiturn "GDAM" = 47h 44h 41h 4Dh → activo Singleturn "GBMM" = 47h 42h 4Dh 4Dh → multivoPlus Multiturn "GBAM" = 47h 42h 41h 4Dh → multivoPlus Singleturn |

Objekt 1009 Hersteller Hardware Version

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | |
| EEPROM | No |
| Beschreibung | Hardware-Version in ASCII |
| Werte | Data 0..3 Beispiel: 31h 2Eh 30h 30h = "1.00" |

Objekt 100A Hersteller Software Version

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | |
| EEPROM | No |
| Beschreibung | Software-Version in ASCII |
| Werte | Data 0..3 Beispiel: 31h 2Eh 30h 30h = "1.00" |



Objekt 100C Guard Time

| | |
|--------------|------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadWrite |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | Timer für Nodeguarding in ms |
| Werte | 0...65535 |

Objekt 100D Life Time Factor

| | |
|--------------|---|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadWrite |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | Dieser Faktor multipliziert mit der Guard Time ergibt die Life time |
| Werte | 0...256 |



Objekt 1010 Parameter 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 | 01 | 73h 's' | 61h 'a' | 76h 'v' | 65h 'e' |

Im EEPROM gespeicherte Objekte:

| Objekt | SubIndex | Beschreibung | Default Wert (nach Objekt 1011) |
|--------|----------|---|--|
| 1005h | 0h | Sync ID | 80h |
| 1008h | 0h | Device Name | "GXMM" → multivo Multiturn "GXAM" → multivo Singleturn "GCMM" → magtivo Multiturn "GCAM" → magtivo Singleturn "GDMM" → activo Multiturn "GDAM" → activo Singleturn "GBMM" → multivoPlus Multiturn "GBAM" → multivoPlus Singleturn |
| 100Ch | 0h | Guard Time | 0h |
| 100Dh | 0h | Life Time Factor | 0h |
| 1014h | 0h | Emergency COB-ID | 80h+Node-ID |
| 1016h | 1 | Consumer heartbeat time | 10000h |
| 1017h | 0h | Producer Heartbeat time | 0h (disabled) |
| 1018h | 1h | VendorID | ECh |
| 1018h | 2h | Product Code | 0Ah → multivo Multiturn 0Bh → multivo Singleturn 0Ch → magtivo Multiturn 0Dh → magtivo Singleturn 0Eh → activo/multivoPlus Multiturn 0Fh → activo/multivoPlus Singleturn |
| 1018h | 4h | Serial Number | xyz |
| 1029h | 1h | Error Behavior | 1 |
| 1800h | 1h | PDO1 ID | 180h+Node-ID |
| 1800h | 2h | PDO1 Type | FEh → asynchron, zyklisch |
| 1800h | 5h | PDO1 EventTimer asynchron Betrieb | 203h ms |
| 1801h | 1h | PDO2 ID | 280h+Node-ID |
| 1801h | 2h | PDO2 Type | 2h → synchron |
| 1801h | 5h | PDO2 Refresh-Zeit für zyklisches Senden | 100h ms |
| 2100h | 0h | Baudrate | 2h = 50 kBaud |
| 2101h | 0h | Node-ID | 1h |
| 2110h | 0h | Manufacturer_Options | 0x00000008 |
| 2201h | 1h | Anzahl Positionsfehler | 0h |
| 2201h | 2h | Betriebszeit Total in Sekunden | 0h |
| 2201h | 3h | Anzahl TimerReset vom WatchDog | 0h |
| 2300h | 1h | Kundenspezifischer EEPROM Bereich Data0 | 0h |
| 2300h | 2h | Kundenspezifischer EEPROM Bereich Data1 | 0h |
| 2300h | 3h | Kundenspezifischer EEPROM Bereich Data2 | 0h |
| 2300h | 4h | Kundenspezifischer EEPROM Bereich Data3 | 0h |
| 2300h | 5h | Kundenspezifischer EEPROM Bereich Data4 | 0h |
| 2300h | 6h | Kundenspezifischer EEPROM Bereich Data5 | 0h |
| 2300h | 7h | Kundenspezifischer EEPROM Bereich Data6 | 0h |
| 2300h | 8h | Kundenspezifischer EEPROM Bereich Data7 | 0h |
| 2800h | 0h | PDO1-Zusatz (EventTrigger) | 0h |
| 2801h | 0h | PDO2-Zusatz (EventTrigger) | 0h |
| 6000h | 0h | Operating Parameter | 0004h |
| 6001h | 0h | Anzahl Schritte pro Umdrehung | 2000h → multivo 1000h → magtivo 40000h → activo/multivoPlus |



| | | | |
|-------|----|--------------------------------|--|
| 6002h | 0h | Gesamtmessbereich in Schritten | 20000000h → multivo Multiturn 2000h → multivo Singleturn 4000000h → magtivo Multiturn 1000h → magtivo Singleturn 80000000h → activo/multivoPlus Multiturn 40000h → activo/multivoPlus Singleturn |
| 6003h | 0h | Preset Wert in Schritten | 0h |
| 6200h | 0h | Zyklischer Timer für PDO1 | 203h (siehe Objekt 1800-5) |
| 6509h | 0h | Offset | 0h |
| 650Bh | 0h | Serie Nummer | xyz (siehe Objekt 1018-4) |

Objekt 1011 Restore Parameter

Über das Objekt 1011h werden die Werte im RAM mit den Default-Werten (siehe Objekt 1010h) überschrieben. Zudem wird der Inhalt des EEPROM als ungültig markiert. Dies bedeutet, dass bis zum nächsten Speichern der Daten im EEPROM, jeweils die Default Werte geladen werden.
Um ein unabsichtliches Überschreiben zu verhindern muss die Botschaft „load“ 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 | 11h | 10h | 01 | 6Ch 'l' | 6Fh 'o' | 61h 'a' | 64h 'd' |

Objekt 1014 COB-ID Emergency Message

| | |
|--------------|---------------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadWrite |
| Default | 80h+Node-ID |
| EEPROM | Yes |
| Beschreibung | Definiert COB-ID des Emergency Objekt |
| Werte | 80h + Node-ID |

Objekt 1016 Consumer heartbeat Time

| | |
|--------------|--------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | Read only |
| Default | 1 |
| EEPROM | No |
| Beschreibung | Größter unterstützter SubIndex |
| Werte | 1 |

| | |
|--------------|---|
| SubIndex | 1 |
| Daten-Typ | Unsigned 32 |
| Zugriff | Read write |
| Default | 10000h |
| EEPROM | Yes |
| Beschreibung | Consumer heartbeat time |
| Werte | Bit 0..15 Consumer heartbeat time in ms Bit 16..23 Node ID |

Objekt 1017 Producer Heartbeat Time

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadWrite |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | Definiert die Wiederholzeit des Überwachungsdienstes Heartbeat |
| Werte | 0 = Disabled 1..65535 = Wiederholzeit in ms |



Objekt 1018 Identity Object

| | |
|--------------|--------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadOnly |
| Default | 4 |
| EEPROM | No |
| Beschreibung | Größter unterstützter Subindex |
| Werte | 4 |

| | |
|--------------|---|
| SubIndex | 1 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | ECh |
| EEPROM | Yes |
| Beschreibung | Von CiA vergebene VendorID für Baumer IVO GmbH & Co. KG |
| Werte | ECh (im Internet unter www.can-cia.de) |
| SubIndex | 2 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 0Ah 0Bh 0Ch 0Dh 0Eh 0Fh |
| EEPROM | Yes |
| Beschreibung | Product Code |
| Werte | 0Ah → multivo Multiturn 0Bh → multivo Singleturn 0Ch → magtivo Multiturn 0Dh → magtivo Singleturn 0Eh → activo/multivoPlus Multiturn 0Fh → activo/multivoPlus Singleturn |

| | | | | |
|--------------|---|--------------------------|----------------------|-----------------------|
| SubIndex | 3 | | | |
| Daten-Typ | Unsigned 32 | | | |
| Zugriff | ReadOnly | | | |
| Default | | | | |
| EEPROM | No | | | |
| Beschreibung | Revisions Nummer des Sensors | | | |
| Werte | Version der aktuellen Software = xxyy (xx=Version, yy=Laufnummer) | | | |
| | Data 0 = Laufnummer LOW | Data 1 = Laufnummer HIGH | Data 2 = Version LOW | Data 3 = Version HIGH |
| | siehe Typenschild | | | |

| | |
|--------------|--|
| SubIndex | 4 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 0 |
| EEPROM | Yes |
| Beschreibung | Fortlaufende eindeutige Seriennummer des Sensors |
| Werte | Wird im Werk während des Endtests definiert |



Objekt 1029 Error Behavior (ab Firmware Version V1.04)

| | |
|--------------|------------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadOnly |
| Default | 1 |
| EEPROM | No |
| Beschreibung | Größter unterstützter Subindex |
| Werte | 1 = Größter unterstützter SubIndex |

| | |
|--------------|--|
| SubIndex | 1 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadWrite |
| Default | 1 |
| EEPROM | Yes |
| Beschreibung | Verhalten nach Communication error |
| Werte | 0h = Wechsel in den Pre-Operational Mode 1h = kein Mode-Wechsel 2h = Wechsel in den Stop Mode 3h = Knoten reset |

Objekt 1800 PDO1 Parameter

| | |
|--------------|--------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 5 |
| EEPROM | No |
| Beschreibung | Größter unterstützter Subindex |
| Werte | 5 |

| | |
|--------------|--|
| SubIndex | 1 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadWrite |
| Default | 180h + Node-ID |
| EEPROM | Yes |
| Beschreibung | COB-ID des PDO |
| Werte | 180h + Node-ID |
| SubIndex | 2 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadWrite |
| Default | FEh |
| EEPROM | Yes |
| Beschreibung | PDO Type |
| Werte | 1..n..F0h = PDO hat synchrone Charakteristik (auf jedes n-te SYNC-Telegramm wird das PDO gesendet) FEh = PDO hat asynchrone Charakteristik (PDO's werden zyklisch in Abhängigkeit vom EventTimer und EventTrigger gesendet) |

| | |
|--------------|---|
| SubIndex | 5 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadWrite |
| Default | 203h |
| EEPROM | Yes |
| Beschreibung | Event Timer für Prozess Daten Objekt |
| Werte | 0 = zyklisches Senden ausgeschaltet 1..n..65535 =Wiederholzeit zyklisches Senden beträgt n ms. |



Objekt 1801 PDO2 Parameter

Siehe Objekt 1800h mit Ausnahme SubIndex1 COB-ID ist hier 280h + Node-ID

Objekt 1A00 PDO1 Mapping

| | |
|--------------|--------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadOnly |
| Default | 0 |
| EEPROM | No |
| Beschreibung | Größter unterstützter Subindex |
| Werte | 1 |

| | |
|--------------|--|
| SubIndex | 1 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly (von CiA als ReadWrite definiert) |
| Default | 6004h |
| EEPROM | No |
| Beschreibung | Beschreibt den Inhalt der PDO1-Meldung |
| Werte | 6004h = Position |

Objekt 1A01 PDO2 Mapping

| | |
|--------------|--------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadOnly |
| Default | 0 |
| EEPROM | No |
| Beschreibung | Größter unterstützter Subindex |
| Werte | 1 |

| | |
|--------------|--|
| SubIndex | 1 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly (von CiA als ReadWrite definiert) |
| Default | 6004h |
| EEPROM | No |
| Beschreibung | Beschreibt den Inhalt der PDO2-Meldung |
| Werte | 6004h = Position |

Objekt 2100 Baudrate

| | |
|--------------|---|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadWrite |
| Default | 2 = 50 kBaud |
| EEPROM | Yes |
| Beschreibung | Baudrate des Sensors lesen oder neu setzen. → Nach dem Setzen müssen Parameter mit dem Objekt 1010h im EEPROM gespeichert und danach der Sensor neu initialisiert werden |
| Werte | 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 |

**Objekt 2101 Node-ID**

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadWrite |
| Default | 1 |
| EEPROM | Yes |
| Beschreibung | Node-ID des Sensors lesen oder neu setzen. → Nach dem Setzen müssen Parameter mit dem Objekt 1010h im EEPROM gespeichert und danach der Sensor neu initialisiert werden |
| Werte | 1..127 |

Objekt 2110 Manufacturer_Options

| | |
|--------------|---|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadWrite |
| Default | 8h |
| EEPROM | Yes |
| Beschreibung | Einstellungen, um die Kompatibilität zu älteren Sensoren zu gewährleisten bzw. um kundenspezifische Einstellungen vorzunehmen Objekt wird nicht über EDS Datei unterstützt. Einstellungen sollten nur vom Hersteller vorgenommen werden. Kundenseitige Veränderungen bitte strikt nach untenstehender Tabelle vornehmen. |
| Werte | Bit1 = Drehrichtungssinn (Objekt 6000h Bit0) (ab V1.04) 0 Nicht invertiert 1 Invertiert Bit2 = Skalierungsfunktion (Objekt 6000h Bit2) (ab V1.04) 0 Freigegeben 1 Gesperrt Bit3 = 0 BusOFF wird nicht zurückgesetzt 1 Wenn BusOFF wird der Bus wieder zurückgesetzt Bit5 = 0 Heartbeat-Protokoll aktiv 1 Nodeguarding-Protokoll aktiv Bit6 = 0 Beim SYNC-Telegramm wird der bereits ermittelte Positionswert ausgegeben (ab V1.04) 1 Beim SYNC-Telegramm wird Positionswert neu eingelesen Bit7 = minimaler Positions-Jitter beim SYNC (ab V1.04) 0 Positionsermittlung zyklisch 1 Position wird erst auf ein SYNC-Telegramm eingelesen (Bit6 muss aktiviert sein) → minimaler Positions-Jitter Bit8 = PDO1 Zeitverzögerung 2ms (ab V1.04) 0 1800h-5h = 6200h 1 1800h-5h = 6200h + 2ms Bit9 = Verhalten bei schreiben auf Objekt Auflösung/Gesamtauflösung 0 Offset löschen 1 Offset nicht löschen (ab Version V1.08) Bit10 = Verhalten bei Reset Node (ab V1.09) 0 HW Reset 1 Init NMT state |



Objekt 2201 Statistik

| | |
|--------------|---------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadOnly |
| Default | 3h |
| EEPROM | No |
| Beschreibung | Grösster unterstützter Subindex |
| Werte | 3 |

| | |
|--------------|---------------------------------------|
| SubIndex | 1 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | Anzahl der Positions-Fehler insgesamt |
| Werte | 0...4294967295 |

| | |
|--------------|---|
| SubIndex | 2 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | Betriebszeit in Sekunden Total (Objekt 6508h Zeit seit letztem Reset) |
| Werte | 0... 4294967295 |

| | |
|--------------|----------------------------|
| SubIndex | 3 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | WatchDog TimerReset Zähler |
| Werte | 0... 4294967295 |

Objekt 2300 Customer EEPROM Bereich

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadOnly |
| Default | 8h |
| EEPROM | No |
| Beschreibung | In diesem Objekt können beliebige Daten abgespeichert werden |
| Werte | 8 |

| | |
|--------------|--|
| SubIndex | 1...8 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadWrite |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | Pro Subindex kann ein 16 Bit Wert abgespeichert werden (speichern im EEPROM über Objekt 1010h) |
| Werte | 0 |

**Objekt 2800 PDO1-Zusatz (EventTrigger)**

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadWrite |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | Der Wert EventTrigger bestimmt, wie oft der gleiche PDO-Wert übertragen wird |
| Werte | 0 = PDO-Zähler ist ausgeschaltet → ständiges übermitteln (Zeitbasis vom EventTimer) 1..n..255 = der gleiche PDO-Wert wird n-mal gesendet (Zeitbasis vom EventTimer) |

Objekt 2801 PDO2-Zusatz (EventTrigger)

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 8 |
| Zugriff | ReadWrite |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | Der Wert EventTrigger bestimmt, wie oft der gleiche PDO-Wert übertragen wird |
| Werte | 0 = PDO-Zähler ist ausgeschaltet → ständiges übermitteln (Zeitbasis vom EventTimer) 1..n..255 = der gleiche PDO-Wert wird n-mal gesendet (Zeitbasis vom EventTimer) |

Objekt 6000 Operating Parameter

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadWrite |
| Default | 4 |
| EEPROM | Yes |
| Beschreibung | Betriebsparameter |
| Werte | Bit 0 Drehrichtung = 0 → Uhrzeigersinn; 1 → Gegenuhrzeigersinn Bit 2 Skalierungsfunktion = 0 → max. Auflösung; 1 → gespeicherte Auflösung |

Objekt 6001 Auflösung

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadWrite |
| Default | 2000h = 8192 = 13Bit → multivo 1000h = 4096 = 12Bit → magtivo 40000h = 262144 = 18Bit → activo/multivoPlus |
| EEPROM | Yes |
| Beschreibung | Anzahl Schritte pro Umdrehung frei wählbar. ! Offsetwert wird bei einer Änderung der Auflösung zurück gesetzt ! |
| Werte | 1..n.. Max. Anzahl Schritte pro Umdrehung (siehe Objekt 6501) 1..n..8192 → multivo 1..n..4096 → magtivo 1..n..262144 → activo/multivoPlus |

**Objekt 6002 Gesamtmeßbereich**

| | |
|--------------|---|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadWrite |
| Default | 20000000h = 536870912 = 29Bit → multivo Multiturn 2000h = 8192 = 13Bit → multivo Singleturn 4000000h = 67108864 = 26Bit → magtivo Multiturn 1000h = 4096 = 12Bit → magtivo Singleturn 80000000h = 2147483648 = 31Bit → activo/multivoPlus Multiturn 40000h = 262144 = 18Bit → activo/multivoPlus Singleturn |
| EEPROM | Yes |
| Beschreibung | Gesamtmeßbereich in Schritten frei wählbar. Daraus resultiert: Anzahl der Umdrehungen = $\frac{\text{Gesamtmeßbereich}}{\text{Auflösung}}$ Hinweis beim Betrieb des Multiturn-Drehgebers: Wenn die Anzahl der Umdrehungen auf den Wert ungleich 2^n (1, 2, 4,...65536) programmiert ist, muss nach Überfahren des Gebernulpunktes im stromlosen Zustand, neu programmiert werden. |
| Werte | 1... Gesamtmeßbereich in Schritte (siehe Objekt 6502) 1...536870912 → multivo Multiturn 1...8192 → multivo Singleturn 1...67105564 → magtivo Multiturn 1...4096 → magtivo Singleturn 1...2147483648 → activo/multivoPlus Multiturn 1...262144 → activo/multivoPlus Singleturn |

Objekt 6003 Preset Wert

| | |
|--------------|---|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadWrite |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | Frei wählbarer Positionswert. Preset und interne Position ergeben Offset (→ Objekt 6509h) |
| Werte | 0..aktueller Gesamtmeßbereich-1 (Objekt 6002h) |

Objekt 6004 Position in Inkremente (Schritte)

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | |
| EEPROM | No |
| Beschreibung | Aktuelle Position inkl. Offset |
| Werte | 0..aktueller Gesamtmeßbereich-1 (Objekt 6002h) |

Objekt 6200 Cyclic Timer für PDO1

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadWrite |
| Default | 203h |
| EEPROM | Yes |
| Beschreibung | EventTimer für Prozess Daten Objekt (siehe Objekt 1800-5) |
| Werte | 0 = zyklisches Senden ausgeschaltet 1...65535 = Wiederholzeit zyklisches Senden beträgt n ms. |

**Objekt 6500 Operating Status**

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadOnly |
| Default | 4h |
| EEPROM | No |
| Beschreibung | Betriebsdaten welche mit Objekt 6000h geschrieben werden |
| Werte | Bit 0 Drehrichtung = 0 → Uhrzeigersinn; 1 → Gegenuhrzeigersinn Bit 2 Skalierungsfunktion = 0 → max. Auflösung; 1 → gespeicherte Auflösung |

Objekt 6501 Max. Auflösung in Schritten

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 2000h = 8192 = 13Bit → multivo 1000h = 4096 = 12Bit → magtivo 40000h = 262144 = 18Bit → activo/multivoPlus |
| EEPROM | No |
| Beschreibung | Maximale Singleturn-Auflösung in Schritten |
| Werte | 2000h = 8192 = 13Bit → multivo 1000h = 4096 = 12Bit → magtivo 40000h = 262144 = 18Bit → activo/multivoPlus |

Objekt 6502 Gesamtmessbereich in Schritten

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 20000000h = 536870912 = 29Bit → multivo Multiturn 2000h = 8192 = 13Bit → multivo Singleturn 4000000h = 67108864 = 26Bit → magtivo Multiturn 1000h = 4096 = 12Bit → magtivo Singleturn 80000000h = 2147483648 = 31Bit → activo/multivoPlus Multiturn 40000h = 262144 = 18Bit → activo/multivoPlus Singleturn |
| EEPROM | No |
| Beschreibung | Maximaler Gesamtmessbereich (Der Daten-Typ U32 in diesem Objekt entspricht nicht dem CiA-Profil) |
| Werte | 20000000h = 536870912 = 29Bit → multivo Multiturn 2000h = 8192 = 13Bit → multivo Singleturn 4000000h = 67108864 = 26Bit → magtivo Multiturn 1000h = 4096 = 12Bit → magtivo Singleturn 80000000h = 2147483648 = 31Bit → activo/multivoPlus Multiturn 40000h = 262144 = 18Bit → activo/multivoPlus Singleturn |

Objekt 6503 Alarme

| | |
|--------------|------------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadOnly |
| Default | 0h |
| EEPROM | No |
| Beschreibung | Alarmmeldungen gem. Objekt 6504h |
| Werte | Bit 0 = 1 → Positions-Fehler aktiv |



Objekt 6504 Unterstützte Alarmer

| | |
|--------------|---|
| SubIndex | 0 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadOnly |
| Default | 1h |
| EEPROM | No |
| Beschreibung | Vom Objekt 6503 unterstützte Alarmmeldungen |
| Werte | Bit 0 = Positions-Fehler |

Objekt 6505 Warnungen

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadOnly |
| Default | 0h |
| EEPROM | No |
| Beschreibung | Warnungen gem. Objekt 6506h |
| Werte | Multiturn: Bit 2 = 1 → CPU Watchdog geresetet Bit 4 = 1 → Batterie-Ladung zu tief Singleturn: Bit 2 = 1 → CPU Watchdog geresetet |

Objekt 6506 Unterstützte Warnungen

| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 16 |
| Zugriff | ReadOnly |
| Default | Multiturn: 14h Singleturn: 04h |
| EEPROM | No |
| Beschreibung | Vom Objekt 6505h unterstützte Warnungen |
| Werte | Multiturn: Bit 2 = CPU Watchdog Status Bit 4 = Batterie-Ladung Singleturn: Bit 2 = CPU Watchdog Status |

Objekt 6507 Profil und SoftwareVersion

| | | | | |
|--------------|--|--------------------------------|---------------------------------|----------------------------------|
| SubIndex | 0 | | | |
| Daten-Typ | Unsigned 32 | | | |
| Zugriff | Read Only | | | |
| Default | | | | |
| EEPROM | No | | | |
| Beschreibung | Version des Profils und der aktuellen Software | | | |
| Werte | Version der aktuellen Software = xxyy | | | |
| | Data0 = Profil Version LOW | Data1 = Profil Version HIGH | Data2 = Software Version LOW | Data3 = Software Version HIGH |
| | (xx = Software Version, yy = Profil Version) | | | |
| | Data 0,1 = 01h 02h = 0201h = Profil Version Data 2,3 = siehe Typenschild = Software Version | | | |

**Objekt 6508 Operating Time**

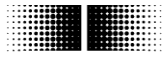
| | |
|--------------|--|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 0h |
| EEPROM | No |
| Beschreibung | Betriebszeit in 1/10 Stunden, seit dem letzten Reset des Sensors |
| Werte | 0..n..4294967295 = n * 6 Minuten Betriebszeit ohne Reset |

Objekt 6509 Offset

| | |
|--------------|---------------------------------------|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | 0h |
| EEPROM | Yes |
| Beschreibung | Aus Preset (→ Objekt 6003h) errechnet |
| Werte | 0..aktueller Gesamtmessbereich-1 |

Objekt 650B Seriennummer

| | |
|--------------|---|
| SubIndex | 0 |
| Daten-Typ | Unsigned 32 |
| Zugriff | ReadOnly |
| Default | xyz |
| EEPROM | Yes |
| Beschreibung | Fortlaufende Seriennummer |
| Werte | 0..4294967295 = Ist direkt verknüpft mit der Seriennummer vom Endtest (siehe Objekt 1018-4) |



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 Drehgebers, 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 dieselbe NodeID 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.

Siehe separate Objektbeschreibung

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 |

5.3. Wissenswertes zum Sensor

Node-ID neu setzen

1. Die Node-ID wird mit dem Baumer IVO spezifischen Objekt 2101h 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 IVO spezifischen Objekt 2100h 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 ansprechen 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)
 1 -> CCW (Gegenuhrzeigersinn)
 Bit 2: 0 -> Skalierung aus
 1 -> Skalierung ein

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 4096 (12 Bit) Schritte pro Umdrehung (4096 = 1000h):

| 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 12 Bit (4096 Schritte) und die Gesamtauflösung wird auf 24 Bit (16777216) eingestellt, somit ergeben sich 4096 (12 Bit) Umdrehungen à 4096 (12 Bit) Schritte.

Gesamtauflösung auf 16777216 einstellen (16777216 = 1000000h)

| 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' |

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 | 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 | 0 | PDO senden ausgeschaltet |
| FEh | 0ms | xxx | PDO senden ausgeschaltet |
| 3 | xxx | 0 | Bei jedem dritten Sync-Telegramm senden |
| 3 | xxx | 2Bh | Auf jedes dritte Sync-Telegramm, aber insgesamt 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).



6.4. Inbetriebnahme über CAN

Einfache Inbetriebnahme des CANopen Drehgebers über CAN (Layer 2)

Beispiel: Drehgeber mit eingestellter Knotennummer 1

Verwendetes Tool: CANAnalyser32 von Fa. IXXAT

| Nr | ID (hex) | Name | Description | RTR | Data (hex) |
|----------|----------|------|------------------------------|-----|-------------------------|
| 1 (byt) | | | | 0 | |
| 2 (byt) | 601 | SDO | Gesamtmessbereich lesen | 0 | 40 02 60 00 |
| 3 (byt) | 601 | SDO | Gesamtmessbereich setzen | 0 | 22 02 60 00 00 00 00 10 |
| 4 (byt) | 601 | SDO | Auflösung lesen | 0 | 40 01 60 00 |
| 5 (byt) | 601 | SDO | Auflösung setzen | 0 | 22 01 60 00 00 10 00 00 |
| 6 (byt) | 601 | SDO | Position lesen | 0 | 40 04 60 00 |
| 7 (byt) | 601 | SDO | Preset setzen Position 0 | 0 | 22 03 60 00 00 00 00 00 |
| 8 (byt) | 601 | SDO | Cyclic timer lesen | 0 | 40 00 62 00 00 |
| 9 (byt) | 601 | SDO | Cyclic timer setzen 5 ms | 0 | 28 00 62 00 05 00 00 00 |
| 10 (byt) | | | | 0 | |
| 11 (byt) | 601 | SDO | Knotennummer lesen | 0 | 40 01 21 00 |
| 12 (byt) | 601 | SDO | Knotennummer setzen auf 2 | 0 | 28 01 21 00 02 00 00 00 |
| 13 (byt) | | | | 0 | |
| 14 (byt) | 601 | SDO | baudrate lesen | 0 | 40 00 21 00 00 00 00 00 |
| 15 (byt) | 601 | SDO | baudrate setzen 250Kbit/s | 0 | 28 00 21 00 05 00 00 00 |
| 16 (byt) | | | | 0 | |
| 17 (byt) | 601 | SDO | save in eeprom | 0 | 23 10 10 01 73 61 76 65 |
| 18 (byt) | 601 | SDO | restore alle parameter | 0 | 23 11 10 01 6C 6F 61 64 |
| 19 (byt) | | | | 0 | |
| 20 (byt) | 601 | SDO | alarm lesen | 0 | 40 03 65 00 00 00 00 00 |
| 21 (byt) | 601 | SDO | warnung lesen | 0 | 40 05 65 00 00 00 00 00 |
| 22 (byt) | | | | 0 | |
| 23 (byt) | 0 | NMT | set Operational Node 1 (RUN) | 0 | 01 01 |
| 24 (byt) | 0 | NMT | set Preoperational Node 1 | 0 | 80 01 |
| 25 (byt) | 0 | NMT | Stopp Node 1 | 0 | 02 01 |
| 26 (byt) | 0 | NMT | Reset Node 1 | 0 | 81 01 |
| 27 (byt) | | | | 0 | |
| 28 (byt) | 601 | SDO | Gesamtmessbereich setzen | 0 | 22 02 60 00 00 00 00 10 |

= 0x100000

= 0x1000

Erst nach Power

Off/On wirksam

Lädt Default-

Einstellungen

Siehe unter

Abschnitt
Netzwerkmanage-
ment Dienste

COB ID = 0x600 + Knotennummer

SDO Kommando

Objekt Index 6002

Objekt Subindex 00

Daten 0x10000000

Weitere Beschreibung im Abschnitt 'Servicedaten-Kommunikation'



Aufzeichnungen der CAN-Telegramme zum und vom Drehgeber

(Kommandos von Seite vorher)

| ID (hex) | Name | Data (hex) | ASCII | |
|----------|------|-------------------------|--------------|--------------------------------------|
| 701 | | 00 | | Boot up Message nach dem Einschalten |
| 601 | SDO | 40 02 60 00 | @ | |
| 581 | | 43 02 60 00 00 00 00 20 | C | |
| 601 | SDO | 22 02 60 00 00 00 00 10 | " | SDO request zum Drehgeber (Anfrage) |
| 581 | | 60 02 60 00 00 00 00 00 | . | COB ID = 0x600+Knotennummer |
| 601 | SDO | 40 01 60 00 | @ | |
| 581 | | 43 01 60 00 00 20 00 00 | C | |
| 601 | SDO | 22 01 60 00 00 10 00 00 | " | |
| 581 | | 60 01 60 00 00 00 00 00 | . | SDO response vom Drehgeber (Antwort) |
| 601 | SDO | 40 04 60 00 | @ | COB ID = 0x580+Knotennummer |
| 581 | | 43 04 60 00 C9 CA 03 00 | C . EE | |
| 601 | SDO | 22 03 60 00 00 00 00 00 | " | |
| 581 | | 60 03 60 00 00 00 00 00 | . | |
| 601 | SDO | 40 00 62 00 00 | @ . b | |
| 581 | | 4B 00 62 00 03 02 00 00 | K . b | |
| 601 | SDO | 2B 00 62 00 05 00 00 00 | + . b | |
| 581 | | 60 00 62 00 00 00 00 00 | . . b | |
| 601 | SDO | 40 01 21 00 | @ . ! | |
| 581 | | 4F 01 21 00 01 00 00 00 | O . ! | |
| 601 | SDO | 2B 01 21 00 02 00 00 00 | + . ! | |
| 581 | | 60 01 21 00 00 00 00 00 | . . ! | |
| 601 | SDO | 40 00 21 00 00 00 00 00 | @ . ! | |
| 581 | | 4F 00 21 00 02 00 00 00 | O . ! | |
| 601 | SDO | 2B 00 21 00 05 00 00 00 | + . ! | |
| 581 | | 60 00 21 00 00 00 00 00 | . . ! | |
| 601 | SDO | 23 10 10 01 73 61 76 65 | # . . . save | |
| 581 | | 60 10 10 01 00 00 00 00 | | |
| 601 | SDO | 23 11 10 01 6C 6F 61 64 | # . . . load | |
| 581 | | 60 11 10 01 00 00 00 00 | | |
| 601 | SDO | 40 03 65 00 00 00 00 00 | @ . e | |
| 581 | | 4B 03 65 00 00 00 00 00 | K . e | |
| 601 | SDO | 40 05 65 00 00 00 00 00 | @ . e | |
| 581 | | 4B 05 65 00 00 00 00 00 | K . e | Geber jetzt im Status Operational |
| 0 | NMT | 01 01 | . | |
| 181 | | 92 95 07 00 | . . . | Run, sendet zyklisch Positions-Daten |
| 181 | | 92 95 07 00 | . . . | COB ID = 0x180+Knotennummer |
| 181 | | 92 95 07 00 | . . . | Geber im Status Pre-operational |
| 181 | | 92 95 07 00 | . . . | Geber im Status Stopped |
| 0 | NMT | 80 01 | . | |
| 0 | NMT | 02 01 | . | |
| 0 | NMT | 81 01 | . | Geber führt Reset aus |
| 701 | | 00 | . | Boot up Message |
| | | | | COB ID = 0x700+Knotennummer |

7. Anschlussbelegung und Inbetriebnahme

7.1. Mechanischer Anbau

Wellen-Drehgeber

- Drehgebergehäuse 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/Endwellen-Drehgeber

- Klemmringbefestigung
Vor Montage des Drehgebers den Klemmring vollständig öffnen. Drehgeber auf die Antriebswelle 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.

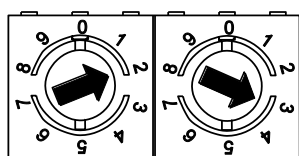
Zum elektrischen Anschluss Bushaube folgendermaßen abziehen:

- 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

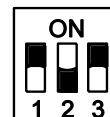


7.2.2. Baudrate einstellen

Die Baudrate ist im Objekt 2100h festgelegt. Zusätzlich gibt es 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 Schalter 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:

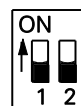


7.2.3. Abschlusswiderstand

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 DIP Schalter auf „ON“ geschaltet werden (Werkseinstellung OFF).



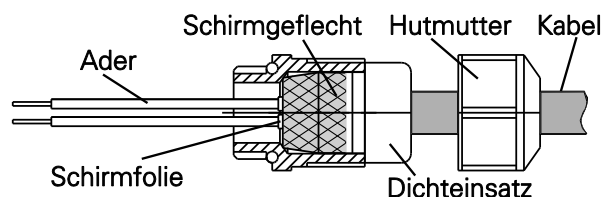
ON = Letzter Teilnehmer
OFF = Teilnehmer X



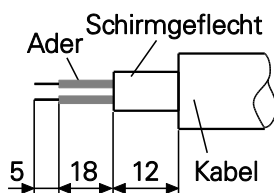
Schalter 1: ON = Letzter Teilnehmer
OFF = Teilnehmer X
Schalter 2: ohne Funktion

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



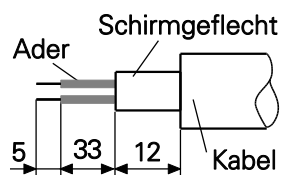
Für Standard Drehgeber



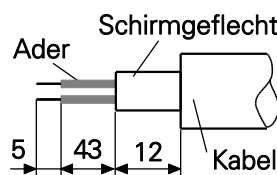


Für G0AMH, G0MMH, GBAMH und GBMMH

Busleitung



Betriebsspannungsleitung

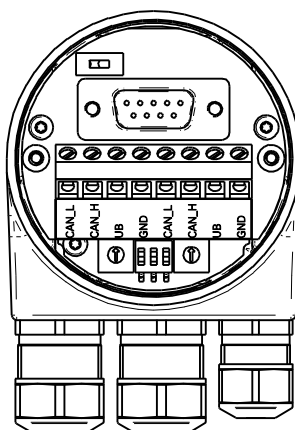


- Klemmen mit gleicher Bezeichnung sind intern miteinander verbunden.
- Für die Betriebsspannung ausschließlich Kabelverschraubung 3 verwenden. Für die Busleitungen können frei wählbar Kabelverschraubung 1 oder 2 verwendet werden. Zulässige Kabelquerschnitte beachten.
- Adern auf dem kürzesten Weg von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

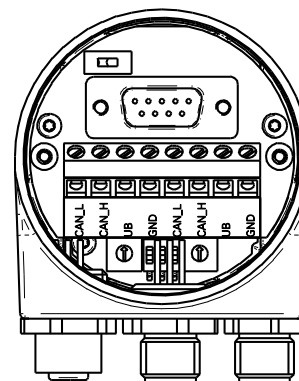
Bushaube – Welle/Endwelle



1 2 3



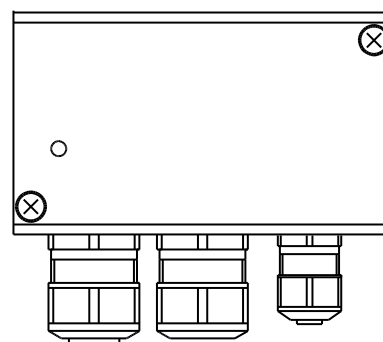
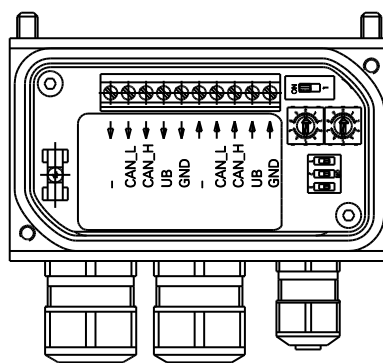
Kabelverschraubung



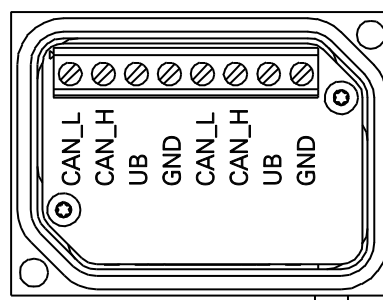
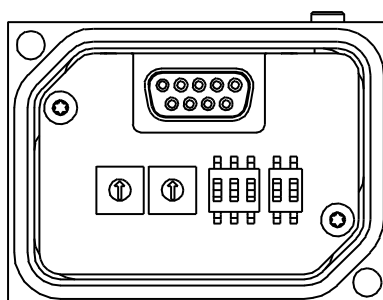
M12-Stecker



Bushaube – Hohlwelle G1 und G2

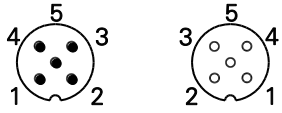


Bushaube – Hohlwelle G0 und GB





7.2.5. Anschlussbelegung

| Pin | Klemme | Beschreibung | M12-Stecker (Stift/Buchse) |
|-----|--------|--------------------------------|---|
| 1 | GND | Masseanschluss für UB |  |
| 2 | UB | Betriebsspannung 10...30 VDC | |
| 3 | GND | Masseanschluss für UB | |
| 4 | CAN_H | CAN Bus signal (dominant High) | |
| 5 | CAN_L | CAN Bus signal (dominant Low) | |

Klemmen mit gleicher Bezeichnung sind intern verbunden und funktionsidentisch. Diese internen Klemmverbindungen UB-UB und GND-GND dürfen mit max. je 1 A belastet werden.

- 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.
- 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 (Statusanzeige)

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 |