

---

# Benutzerhandbuch

---

Magnetische Absolutwert-Drehgeber

mit **CANopen** - Schnittstelle

**WV58M, WH58M**



## Inhaltsverzeichnis

<b>1. Allgemeine Hinweise</b>	<b>4</b>
1.1. Definitionen	4
1.2. Dokumentation	5
1.3. Bestimmungsgemäße Verwendung	5
<b>2. Produktfamilie magnetische Winkelkodierer</b>	<b>5</b>
<b>3. Allgemeine Angaben CAN-Bus</b>	<b>5</b>
3.1. CAN-Bus-Eigenschaften	6
3.2. CANopen	6
3.3. Das Encoder-Geräte-Profil (CiA Draft Standard 406)	7
<b>4. Datenübertragung nach dem CANopen-Kommunikationsmodell</b>	<b>7</b>
4.1. CANopen Meldungsaufbau	8
4.1.1. Funktions-Code	8
4.1.2. Knotennummer (Node-ID)	9
4.2. Übertragung der Prozess-Daten	9
4.2.1. Synchrone Datenübertragung	9
4.2.2. asynchrone Datenübertragung	10
4.3. Übertragung der SDO-Daten (Parametrierung)	10
4.4. Emergency-Dienst	12
4.5. Netzwerkmanagement-Dienste (NMT)	14
4.5.1. Beschreibung der NMT-Kommandos	14
4.5.2. Kommando Byte	14
4.5.3. NMT-Status	14
4.5.4. Die verschiedenen NMT-Zustände	15
4.5.5. Zustandswechsel	15
4.5.6. Heartbeat	16
<b>5. Objektverzeichnis</b>	<b>17</b>
5.1. Objektübersicht	17
5.2. Detaillierte Objektbeschreibung	19
5.2.1. Objekt 1000h (Device Type)	19
5.2.2. Objekt 1001h (Error Register)	19
5.2.3. Object 1002h (Manufacturer Status Register)	20
5.2.4. Object 1003h (Pre-defined Error Field)	20
5.2.5. Object 1005h (COB-ID SYNC message)	21
5.2.6. Object 1008h (Manufacturer Device Name)	21
5.2.7. Object 1009h (Manufacturer Hardware Version)	21
5.2.8. Object 100Ah (Manufacturer Software Version)	22
5.2.9. Object 1010h (Store Parameters)	22
5.2.10. Object 1011h (Load Default Parameters)	23
5.2.11. Object 1014h (COB-ID Emergency Object)	24
5.2.12. Object 1017h (Producer Heartbeat Time)	24
5.2.13. Object 1018h (Identity Object)	25



5.2.14.	Object 1800h (Transmit PDO1 Parameter, asynchrone Betriebsart)	25
5.2.15.	Object 1801h (Transmit PDO2 Parameter, synchrone Betriebsart)	26
5.2.16.	Object 1A00h (Transmit PDO1 Mapping Parameter)	27
5.2.17.	Object 1A01h (Transmit PDO2 Mapping Parameter)	28
5.2.18.	Object 2001h (Manufacturer Offset)	28
5.2.19.	Object 2002h (Geberwert Nullsetzen)	29
5.2.20.	Object 2800h (Sende-Wiederholungszähler für PDO1)	29
5.2.21.	Object 2801h (Sende-Wiederholungszähler für PDO2)	30
5.2.22.	Object 6000h (Operating Parameters)	30
5.2.23.	Object 6001h (Measuring Units per Revolution [Geberauflösung])	31
5.2.24.	Object 6002h (Total Measuring Range [Gesamtschrittzahl])	31
5.2.25.	Object 6003h (Preset value)	32
5.2.26.	Object 6004h (Position value)	32
5.2.27.	Object 6200h (Zyklus Timer)	33
5.2.28.	Object 6500h (Operating Status)	33
5.2.29.	Object 6501h (SingleTurn resolution)	34
5.2.30.	Object 6502h (Number of distinguishable revolutions)	34
5.2.31.	Object 6503h (Alarms)	34
5.2.32.	Object 6504h (Supported Alarms)	35
5.2.33.	Object 6505h (Warnings)	35
5.2.34.	Object 6506h (Supported Warnings)	36
5.2.35.	Object 6507h (Profile and Software Version)	36
5.2.36.	Object 6508h (Operating Time)	36
5.2.37.	Object 6509h (Gebernulldungswert)	37
5.2.38.	Object 650Ah (Module Identification)	37
5.2.39.	Object 650Bh (Serial Number)	38
<b>6.</b>	<b>Einstell- und Diagnoseelemente</b>	<b>38</b>
6.1.	Einstellung des Node-Identifizier (Node-ID)	38
6.2.	Einstellung der Baudrate	39
6.3.	Diagnose LED's	39
<b>7.</b>	<b>Inbetriebnahme</b>	<b>40</b>
7.1.	Einschalten der Versorgungsspannung	40
7.2.	Senden des Positionswertes	40
7.2.1.	Asynchrone (zyklische) Übertragung	40
7.2.2.	Synchrone Übertragung	41
7.3.	Beenden der Positionswertübertragung	41

## 1. Allgemeine Hinweise

Dieses Benutzerhandbuch ist gültig ab Firmwareversion **3.00!** Es beschreibt die Software, Parametrierung und Inbetriebnahme des Drehgebers.

### 1.1. Definitionen



Dieses Symbol steht bei Textstellen, die besonders zu beachten sind, damit der ordnungsgemäße Einsatz gewährleistet ist und Gefahren ausgeschlossen werden.



Dieses Symbol gibt wichtige Hinweise für den sachgerechten Umgang mit dem Drehgeber. Das Nichtbeachten dieser Hinweise kann zu Störungen an dem Drehgeber oder in der Umgebung führen.



Dieses Symbol zeigt Handlungs-Anweisungen an.

<b>CAL</b>	CAN Application Layer. Anwendungsschicht (Schicht 7) im CAN Kommunikations-Modell
<b>CAN</b>	Controller Area Network
<b>CiA</b>	CAN in Automation. Internationaler Verein der Anwender und Hersteller von CAN-Produkten.
<b>COB</b>	Communication Object. Transporteinheit im CAN Netzwerk (CAN Nachricht). Daten werden innerhalb eines COB's über das Netzwerk gesendet.
<b>COB-ID</b>	COB-Identifizier. Eindeutige Kennung einer CAN-Nachricht. Der Identifizier bestimmt die Priorität des COB's im Netzwerk.
<b>ID</b>	Identifizier, siehe COB-ID
<b>LSB</b>	Least Significant Bit/Byte; niederwertigstes Bit/Byte
<b>MSB</b>	Most Significant Bit/Byte; höchstwertigstes Bit/Byte
<b>NMT</b>	Network Management. Service-Element von CAL, verantwortlich für die Initialisierung, Konfiguration und Fehlerbehandlung im Netzwerk.
<b>PDO</b>	Process Data Object. Objekt für den Austausch von Prozessdaten.
<b>RTR</b>	Remote Transmission Request; Datenanforderungstelegramm
<b>SDO</b>	Service Data Object; Kommunikationsobjekt, über das der Master auf das Objektverzeichnis eines Knotens zugreifen kann.
<b>SYNC</b>	Synchronisations-Telegramm. Busteilnehmer antworten mit ihrem Prozesswert auf das SYNC-Kommando.

**Zahlenangaben** falls nicht explizit angegeben, werden dezimale Werte als Ziffern ohne Zusatz angegeben (z.B. 1234), binäre Werte werden mit **b** (z.B. 1011b), hexadezimale Werte mit **h** (z.B. 280h) hinter den Ziffern gekennzeichnet.

## 1.2. Dokumentation

Dieses Benutzerhandbuch ist für die absoluten, magnetischen Winkelkodierer WV58M bzw. WH58M gültig und soll die notwendigen Informationen zur Handhabung dieser Geräte vermitteln.

Hinweise zur Gewährleistung, Sicherheitshinweise und mechanischer Montage der Winkelkodierer WV/WH58M sind der diesen Gebern beiliegenden Benutzerinformation zu entnehmen.

## 1.3. Bestimmungsgemäße Verwendung



Die genannten Winkelkodierer sind Präzisionsmessgeräte. Sie dienen ausschließlich zur Erfassung von Winkelpositionen und Umdrehungen, der Aufbereitung und Bereitstellung der Messwerte als elektrische Ausgangssignale für das Folgegerät. Die Winkelkodierer dürfen ausschließlich zu diesem Zweck verwendet werden.

## 2. Produktfamilie magnetische Winkelkodierer

Die Produktfamilie der magnetischen, absoluten Winkelkodierer umfasst zur Zeit die folgenden 4 Typen:

- **10Bit-Singleturn (1024 Schritte/Umdrehung),**
- **12Bit-Singleturn (4096 Schritte/Umdrehung),**
- **10+12Bit-Multiturn (1024 Schritte/Umdrehung, 4096 Umdrehungen),**
- **12+12Bit-Multiturn (4096 Schritte/Umdrehung, 4096 Umdrehungen)**

Sie sind entweder als Vollwellen- oder als (Sackloch-) Hohlwellen-Ausführung im Standardmaß mit 58mm Durchmesser verfügbar. Die Winkelkodierer zeichnen sich trotz Busausführung durch eine sehr kompakte Bauform aus.

Die Winkelkodierer sind mit folgenden Schnittstellen erhältlich:

- **SN3 (serielle RS485-Schnittstelle mit SIKONETZ3-Protokoll)**
- **SSI (Synchron Serielle Schnittstelle)**
- **PB (Profibus-DP-Schnittstelle)**
- **CAN (CANopen Schnittstelle)**

Im folgenden wird ausschließlich der Winkelkodierer WV/WH58M mit CANopen-Schnittstelle behandelt.

## 3. Allgemeine Angaben 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.

### 3.1. CAN-Bus-Eigenschaften

- Busmedium ist ein paarweise verdrahtetes und abgeschirmtes Kabel,
- Multimasterfähig, d.h. mehrere CAN-Teilnehmer können gleichzeitig den Bus anfordern. Dabei setzt sich die Nachricht mit der höchsten Priorität (festgelegt durch den Identifier) durch,
- Datenrate bis 1Mbit/s zulässig (bei 40m Netz-Ausdehnung),
- Beidseitig abgeschlossenes Netzwerk,
- Theoretisch bis zu 127 Teilnehmer an einem Bus möglich, durch den Treiber bedingt aber praktisch nur bis zu 32 Teilnehmer,
- Nachrichtenorientierte Kommunikation: Die Nachricht wird mit einer Nachrichtenkennung (Identifier) gekennzeichnet. Alle Busteilnehmer prüfen an Hand des Identifiers, ob die Nachricht für sie relevant ist,
- Alle Busteilnehmer erhalten gleichzeitig jede Nachricht. Daher ist eine Synchronisation möglich,
- Der Identifier setzt die Priorität der Nachricht fest. Je niedriger der Identifier vom Wert her ist, desto höhere Priorität hat die Nachricht. Dadurch können wichtige Nachrichten schnell über den Bus übertragen werden,
- Hohe Übertragungssicherheit durch mehrere, einander ergänzende Fehlererkennungsmechanismen,
- Lokalisation von fehlerhaften bzw. ausgefallenen Busteilnehmern. Das CAN-Protokoll beinhaltet eine Funktionsüberwachung von Busteilnehmern. Diese werden, wenn sie fehlerhaft sind, in ihrer Funktion eingeschränkt oder ganz vom Netz abgekoppelt,

### 3.2. 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.

Der Winkelkodierer WV/WH58M mit CANopen-Schnittstelle erfüllt die im „CANopen Application Layer and Communication Profile“ (CiA Draft Standard 301, Version 4.02) und im „CANopen Device profile for encoders“ (CiA Draft Standard 406, Version 3.1) festgelegten Forderungen.

CANopen ermöglicht:

- einfachen Zugriff auf alle Geräte- und Kommunikationsparameter,
- Synchronisation von mehreren Geräten,
- Automatische Konfiguration von Netzwerken,
- zyklischen und ereignisgesteuerten Datenverkehr

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

- Prozessdaten-Objekte für Echtzeitdaten (PDO),

- Servicedaten-Objekte für Parameter- und Programmübertragung (SDO),
- Netzwerk-Management (NMT),
- Vordefinierte Objekte (für Synchronisation, Notfallnachricht)

Zentrales Element des CANopen-Standards ist die Beschreibung der Gerätefunktionalität über ein Objektverzeichnis. Das Objektverzeichnis ist unterteilt in einen Bereich, welcher allgemeine Angaben über das Gerät (Geräteidentifikation, Herstellername, etc.) sowie Kommunikationsparameter enthält, sowie einen Teil, der die spezifische Gerätefunktionalität beschreibt.

Die Identifizierung eines Eintrags („Objekt“) des Objektverzeichnisses erfolgt über einen 16Bit-Index und einen 8Bit-Subindex. Über diese Einträge werden die „Anwendungsobjekte“ eines Gerätes (z.B. Positionswert bei Encodern) in standardisierter Form über das Netzwerk zugänglich gemacht.

Die Funktionalität und Eigenschaften eines CANopen-Geräts können in Form eines standardisierten „elektronischen Datenblatts“ (Electronic Data Sheet, **EDS**) im ASCII-Format beschrieben werden.

Die den verschiedenen Geberausführungen zugeordneten EDS-Dateien sind unter den folgenden Datei-Bezeichnungen auf der Homepage von SIKO GmbH ([www.siko.de](http://www.siko.de)) abrufbar. Sie sind des Weiteren auf der Begleit-CD zu finden:

- Wx58MCAN\_1012.eds (EDS-Datei für 10+12Bit Multiturn)
- Wx58MCAN\_1212.eds (EDS-Datei für 12+12Bit Multiturn)
- Wx58MCAN\_10ST.eds (EDS-Datei für 10Bit-Singleturn)
- Wx58MCAN\_12ST.eds (EDS-Datei für 12Bit-Singleturn)

### 3.3. Das Encoder-Geräte-Profil (CiA Draft Standard 406)

Dieses Profil beschreibt eine herstellerunabhängige und verbindliche Festlegung der Schnittstelle für Drehgeber. Im Profil ist definiert, welche CANopen Funktionen verwendet werden und ebenso wie sie zu verwenden sind. Dieser Standard ermöglicht ein offenes und herstellerunabhängiges Bussystem.

Das Geräteprofil ist gegliedert in zwei Objekt-Klassen:

- die Standard-Klasse C1 beschreibt alle Grundfunktionen, die der Geber enthalten muss,
- die erweiterte Klasse C2 enthält eine Vielzahl von weiteren Funktionen, die von Gebern dieser Klasse entweder unterstützt werden müssen (Mandatory) oder optional sind. Geräte der Klasse C2 enthalten somit alle C1- und C2-mandatory-Funktionen, sowie, Herstellerabhängig, weitere optionale Funktionen.

Im Profil ist außerdem ein Adressbereich definiert, der mit Hersteller-eigenen Sonderfunktionen belegt werden kann.

Der WV/WH58M unterstützt die Klasse C2.

## 4. Datenübertragung nach dem CANopen-Kommunikationsmodell

Das bei CANopen zugrundeliegende Kommunikationsmodell stellt zwei Arten von Kommunikationsmechanismen zur Verfügung:

- Unbestätigte Übertragung von Daten mit einer Länge von 4Byte (**Prozessdaten-Objekte, PDO**). Diese Daten werden mit hoher Priorität übertragen (niedriger COB-Identifizierer). PDO's sind Broadcast-Nachrichten und stellen ihre Daten allen Empfängern am Bus gleichzeitig zur Verfügung.
- Bestätigte Übertragung auch längerer Datensätze (Parameter) zwischen zwei Teilnehmern mit direktem Zugriff auf die Einträge des Objektverzeichnisses des adressierten Teilnehmers (**Servicedaten-**

**Objekte, SDO).** Diese Parameter werden in der Regel azyklisch (z.B. nur einmal beim Hochfahren des Systems) übertragen und haben deshalb eine niedrige Priorität (= hoher COB-Identifizier).



Die Priorität der Nachrichten-Objekte wird über den COB-Identifizier festgelegt.

#### 4.1. CANopen Meldungsaufbau



Zur einfacheren Verwaltung der Identifier verwendet CANopen das „Pre-Defined Connection Set“. Hierbei sind alle Identifier mit Standardwerten im Objektverzeichnis definiert. Es besteht jedoch die Möglichkeit, diese Identifier über SDO-Zugriff kundenspezifisch zu ändern.

Der 11Bit-Identifier (COB-Identifizier) setzt sich aus einem 4Bit-Funktionscode und einer 7Bit-Knotennummer zusammen:

Bit-Nr.	10	9	8	7	6	5	4	3	2	1	0
Art	Funktions-Code				Knotennummer (Node-ID)						
Belegung	x	x	x	x	0	0	x	x	x	x	x

**Hinweis:** Die Bits 5 und 6 sind beim WV/WH58M immer auf 0 gesetzt. Somit sind maximal 31 unterschiedliche Knotennummern einstellbar (Knotennummer 0 nicht erlaubt!)



**Der Funktionscode gibt Aufschluss über die Art der Meldung und die Priorität. Je höher der Wert des COB-Identifizier ist, umso niedriger die Priorität.**

##### 4.1.1. Funktions-Code

Folgende Funktionscodes sind im „Pre-Defined Connection Set“ definiert (es sind nur diejenigen Funktionscodes dargestellt, die vom WV/WH58M verwendet werden):

Objekt	Funktions-Code	Resultierender COB-ID	zugeordneter Kommunikationsparameter bei Index
NMT	0000b	0	-
SYNC	0001b	128 (80h)	1005h
EMERGENCY	0001b	128 (80h) + Node-ID	1014h
PDO1 (tx) <sup>1</sup>	0011b	384 (180h) + Node-ID	1800h
PDO2 (tx) <sup>1</sup>	0101b	640 (280h) + Node-ID	1801h
SDO (tx) <sup>1</sup>	1011b	1408 (580h) + Node-ID	1200h
SDO (rx) <sup>1</sup>	1100b	1536 (600h) + Node-ID	1200h
HEARTBEAT	1110b	1792 (700h) + Node-ID	1017h

<sup>1</sup> (tx) und (rx) aus Sicht des Drehgebers

#### 4.1.2. Knotennummer (Node-ID)

Die 7Bit-Knotennummer wird Hardwaremäßig über 5 DIP-Schalter am Drehgeber eingestellt. Dieser DIP-Schalter ist nach Abnahme des Schraubverschlusses an der Geberhaube erreichbar (siehe Kap. 6.1, *Einstellung des Node-Identifizier (Node-ID)*, Seite 38).

Die 5 DIP-Schalter legen die Bits 0 bis 4 fest. Bits 5 und 6 sind immer fest mit Wert 0 vorgegeben.



Die Knotennummer 0 ist reserviert und darf von keinem Knoten verwendet werden. Resultierende Knotennummern liegen somit im Bereich von 1 .. 31. Beim WV/WH58M wird eine Einstellung der Knotennummer 0 am DIP-Schalter automatisch der Knotennummer 1 zugeordnet! Die Übernahme einer neu eingestellten Knotennummer erfolgt erst beim nächsten Reset/Power-On des Gebers.

Der Drehgeber wird ab Werk mit der Knotennummer 1 ausgeliefert.

#### 4.2. Übertragung der Prozess-Daten

Es stehen die zwei PDO-Dienste PDO1 (tx) und PDO2 (tx) zur Verfügung. Eine PDO-Übertragung kann durch verschiedene Ereignisse initiiert werden:

- asynchron (Ereignisgesteuert) durch internen Gerätetimer oder durch Positionswertänderung,
- synchron als Antwort auf ein SYNC-Telegramm,
- als Antwort auf ein RTR-Telegramm.

Beide PDO's liefern die aktuelle Position des Gebers und sind über die Objekte 1800h, 1801h, 1A00h, 1A01h, 2800h, 2801h und 6200h festgelegt.

Beim WV/WH58M ist der PDO1 der asynchronen und der PDO2 der synchronen Prozessdatenübertragung zugeordnet. PDO2 ist standardmäßig nach jedem Power-On des Gebers gesperrt und muss bei Bedarf per SDO freigeschaltet werden.

Eine Positionswertanforderung per RTR-Telegramm ist ebenfalls nur über den PDO2 möglich.

Die PDO-Nachricht hat folgenden Aufbau:

COB-ID	Prozessdaten im Binärkode			
11Bit	Byte 0 (LSB)	Byte 1	Byte 2	Byte 3 (MSB)
PDO1: 180h+Node-ID	Positionswert in 2er-Komplement-Darstellung			
PDO2: 280h+Node-ID				

##### 4.2.1. Synchroner Datenübertragung

Um die Prozessdaten synchron zu senden, muss im Objekt 1801h, Subindex 2 ein Wert zwischen 1 und F0h (=240) eingeschrieben werden.

Wenn nun der Wert 3 beträgt, wird das PDO2 auf jedes dritte SYNC-Telegramm gesendet (beim Wert 1 wird auf jedes SYNC-Telegramm gesendet), solange im Objekt 2801h eine 0 eingeschrieben ist. Ist dort zum Beispiel eine 5 eingeschrieben, wird das PDO2 nach wie vor auf jedes dritte SYNC-Telegramm geschrieben, insgesamt aber nur 5 mal. Dementsprechend folgt auf das 15. SYNC-Telegramm das letzte PDO.

Der Zähler für die Anzahl der zu übertragenden PDO's wird bei einer Positionsänderung oder durch das Kommando 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 wird der PDO2 von einem Master über das SYNC-Telegramm angefordert (SYNC-COB-ID = 80h).

Soll der PDO2 über ein RTR-Telegramm angefordert werden, so muss in Objekt 1801h, Subindex 2 der Wert 253 (= FDh) eingeschrieben sein.

#### 4.2.2. asynchrone Datenübertragung

Soll ein PDO zyklisch gesendet werden, muss ins Objekt 1800h, Subindex 5 die Zykluszeit in Millisekunden eingetragen werden. Wird der Wert 0ms geschrieben, wird der PDO nicht gesendet. Die Funktion ist ausgeschaltet. Der minimal einzustellende Wert ist 1 (= 1ms).

Eine weitere Möglichkeit bringt das Objekt 2800h: Beträgt der Wert 0, läuft das zyklische Senden wie oben beschrieben. Beträgt der Wert 1, wird zyklisch geprüft, ob eine Änderung des Positionswertes vorliegt. Wenn nicht, wird nicht gesendet. Beträgt der Wert z.B. 4, wird bei jedem Zyklus, falls eine Änderung besteht, der PDO1 viermal gesendet.



Das Senden des PDO1 auf Grund einer Positionswertänderung funktioniert nur dann, wenn die timergesteuerte Übertragung ausgeschaltet ist, d.h. wenn in Subindex 5, Objekt 1800h eine 0 eingeschrieben ist!

#### 4.3. Übertragung der SDO-Daten (Parametrierung)

Über eine SDO-Nachricht kann auf das Objektverzeichnis des Drehgebers zugegriffen werden. Alle Geräteparameter sind in diesem Objektverzeichnis unter genormten Adressen (Indizes) abgelegt und können mit SDO's beschrieben und gelesen werden. SDO's werden im „Anforderungs und Bestätigungs“-Verfahren (Request/Response) zwischen zwei Teilnehmern ausgetauscht.

Es stehen zwei SDO-Dienste zur Verfügung:



- SDO (tx) (Geber → Master): **580h** + Node-ID
- SDO (rx) (Master → Geber): **600h** + Node-ID

#### Die SDO-Identifizierer können nicht verändert werden!

SDO-Nachrichten haben folgenden Aufbau:

COB-ID	Kommando	Index		Subindex	Servicedaten (Parameter)			
SDO + Node-ID	Byte 0	Byte 1 (LSB)	Byte 2 (MSB)	Byte 3	Byte 4 (LSB)	Byte 5	Byte 6	Byte 7 (MSB)

Die Bedeutung von Index, Subindex und Daten ist dem *Kapitel 5, „Objektverzeichnis“* zu entnehmen.

Das Kommando-Byte legt die Länge der Servicedaten (Parameter) fest. Im Falle des WV/WH58M sind die folgenden Kommandobytes gültig:

Kommandobyte	Art	Funktion
23h	SDO (rx), Initiate Download Request	Parameter an Drehgeber senden (Datenlänge = 4Byte)
60h	SDO (tx), Initiate Download Response	Bestätigung der Datenübernahme an den Master
40h	SDO (rx), Initiate Upload Request	Parameter vom Drehgeber anfordern
42h	SDO (tx), Initiate Upload Response	Parameter an Master (Datenlänge = 4Byte)
80h	SDO (tx), Abort Domain Transfer	Drehgeber meldet Fehlercode an Master



- Eine Fehlermeldung (Kommando 80h) ersetzt im Fehlerfall die normale Bestätigung (Response),
- Die Fehlermeldung umfasst sowohl Kommunikations-Protokoll-Fehler als auch Objektverzeichnis-Zugriffsfehler (z.B. Schreibversuch auf Read-Only-Objekt, falscher Index, etc.).

Die Fehlercodes sind im CANopen-Profil (DS 301) bzw. im Encoder-Profil (DSP 406) beschrieben. Die beim WV/WH58M verwendeten Fehlercodes zeigt nachfolgende Tabelle:

Fehlercode	Beschreibung
06010000h	Falscher Zugriff auf ein Objekt.
06010001h	Lesezugriff auf Write-Only.
06010002h	Schreibzugriff auf Read-Only.
06020000h	Objekt existiert nicht im Objektverzeichnis.
06040043h	Allgemeine Parameter Inkompatibilität.
06070010h	Falscher Datentyp, Datenlänge falsch.
06090011h	Subindex existiert nicht.
06090030h	Wertebereich des gewählten Parameters falsch.
06090036h	Maximalwert ist kleiner als Minimalwert.
08000020h	Parameter können nicht zur Applikation übertragen oder gespeichert werden.
08000022h	Parameter können auf Grund des aktuellen Gerätestatus nicht zur Applikation übertragen oder gespeichert werden.

#### Beispiele SDO:

Anfrage eines Wertes von einem Master bei einem Slave ➔ Operating Status (Objekt 6500h):

COB-ID	Kommando	Index L	Index H	Subindex	Data 0	Data 1	Data 2	Data 3
600h + Node-ID	40h	00h	65h	00h	x	x	x	x

Antwort des Slaves auf die Anfrage:

COB-ID	Kommando	Index L	Index H	Subindex	Data 0	Data 1	Data 2	Data 3
580h + Node-ID	42h	00h	65h	00h	a	b	c	d

Schreiben eines Wertes vom Master zu einem Slave → Objekt 1800, Subindex 5 (Event Timer):

COB-ID	Kommando	Index L	Index H	Subindex	Data 0	Data 1	Data 2	Data 3
600h + Node-ID	23h	00h	18h	05h	E8h	03h	00h	00h

Antwort des Slaves auf das schreiben des Wertes:

COB-ID	Kommando	Index L	Index H	Subindex	Data 0	Data 1	Data 2	Data 3
600h + Node-ID	60h	00h	18h	05h	00h	00h	00h	00h

#### 4.4. Emergency-Dienst

Interne Gerätefehler oder Busprobleme lösen eine „Notfall“-Meldung („Emergency“-Message) aus. Das Telegramm hierzu ist wie folgt aufgebaut:

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
80h + Node-ID	Error Code		Error Register	Alarms (Objekt 6503h)		Warnings (Objekt 6505h)		00h



Steht im „Error Register“ der Wert „11h“, so ändert sich die Bedeutung der Bytes 3 – 6 im Emergency-Telegramm. Der Wert „11h“ zeigt Fehler an, die bei der Übertragung von Daten auf dem CAN-Bus aufgetreten sind (siehe Beschreibung „Error Codes“). Hierbei ist der Geber in den Zustand „Error Passive“ eingetreten.

Verringert sich die Störbelastung auf dem CAN-Bus, so wechselt der Geber automatisch wieder in den normalen, als „Error Active“, bezeichneten Zustand. Im anderen Fall, wenn die Störbelastung weiterhin zunimmt, wechselt der Geber in den Zustand „Bus Off“ und führt daran anschließend einen Neustart durch, der durch eine „Boot-Up-Message“ und einer zusätzlichen „Emergency-Message“ (Byte3 und Byte4 = 0) gekennzeichnet wird.

#### Emergency-Message im Falle von Busstörungen:

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
80h + Node-ID	Error Code		Error Register	Transmit Error Counter	Receive Error Counter	00h	00h	00h

Wie bei den SDO-Fehlermeldungen sind auch dem EMERGENCY-Objekt vordefinierte Fehlermeldungen zugeordnet. Eine Teilmenge dieser im CAN Application Layer DS301 beschriebenen Fehlercodes wird vom WV/WH58M verwendet und sind in nachfolgender Tabelle beschrieben:

**Byte 0 .. Byte 1: Error Code**

Error Codes	Beschreibung
0000h	kein Fehler
8001h	CANBus-Kommunikationsfehler: ➔ Acknowledgement Error
8002h	CANBus-Kommunikationsfehler: ➔ Form Error
8003h	CANBus-Kommunikationsfehler: ➔ CRC Error
8004h	CANBus-Kommunikationsfehler: ➔ Stuff Error
8120h	Geber befindet sich im Error Passive Mode
8140h	recovered from Bus Off
FF10h	Geberspezifischer Fehler: ➔ keine Kommunikation mit Gebergrundkarte möglich
FF20h	Geberspezifischer Fehler: ➔ Batteriewarnung; Batterieladung nähert sich der untersten, zulässigen Grenze
FF30h	Geberspezifischer Fehler: ➔ Batterie entladen
FF40h	Geberspezifischer Fehler: ➔ Checksummenfehler in der Kommunikation mit der Gebergrundkarte
FF50h	Geberspezifischer Fehler: ➔ Timeoutfehler in der Kommunikation mit der Gebergrundkarte

**Byte 2: Error Register**

Bit Nr.	Beschreibung
0	Gesetztes Bit zeigt eine generelle Fehlerbedingung an; wird bei jedem auftretenden Fehler gesetzt.
4	Kommunikationsfehler; wird bei auftreten eines CANBus-Kommunikationsfehlers gesetzt (Acknowledgement-, Form-, CRC- und Stuff-Fehler).

**Byte 3 .. Byte 4: Alarms**

Bit Nr.	Beschreibung
0	Positionswert ungültig.
12	Keine Kommunikation mit Gebergrundkarte möglich.
13	Timeoutfehler in Kommunikation mit Grundkarte.
14	Batteriewarnung
15	Batteriealarm

### Byte 5 .. Byte 6: Warnings

Bit Nr.	Beschreibung
4	Batteriezustand kritisch.
12	Störungen im Datenverkehr mit Gebergrundkarte (Checksum-Fehler).
13	Kommunikation mit Gebergrundkarte wurde manuell über DIP-Schalter 1 unterbrochen.

### Byte 7: nicht benutzt

#### 4.5. Netzwerkmanagement-Dienste (NMT)

Das Netzwerkmanagement kann in zwei Gruppen unterteilt werden:

- NMT-Dienst für die Gerätekontrolle; hiermit kann der Geber initialisiert, gestartet und gestoppt werden,
- NMT-Dienst Verbindungsüberwachung („Heartbeat“).

##### 4.5.1. Beschreibung der NMT-Kommandos

Die Kommandos werden als unbestätigte Objekte (Broadcast-Meldungen) übertragen und sind wie folgt aufgebaut:

COB-ID	Byte 1	Byte 2
0h	Kommando Byte	Knotennummer (Node-ID)

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

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

##### 4.5.2. Kommando Byte

Kommando Byte	Beschreibung	Zustandsübergang (siehe Zustandsdiagramm, Abb. 1)
01h	Start_Remote_Node; Wechsel von Zustand „Pre-Operational“ bzw. „Stopped“ in „Operational“	1
02h	Stop_Remote_Node; Wechsel in Zustand „Stopped“	2
80h	Enter_PRE-OPERATIONAL_State; Wechsel in Zustand „Pre-Operational“	3
81h	Re-Initialisierung CAN-Parameter	4
82h	Reset CAN-Karte	5

##### 4.5.3. NMT-Status

Nach dem Initialisieren befindet sich der Geber im Zustand „Pre-Operational“. In diesem Zustand können per SDO Parameter gelesen und beschrieben werden. Um PDO's anzufordern, muss der Geber zuerst in den Zustand „Operational“ geschaltet werden.

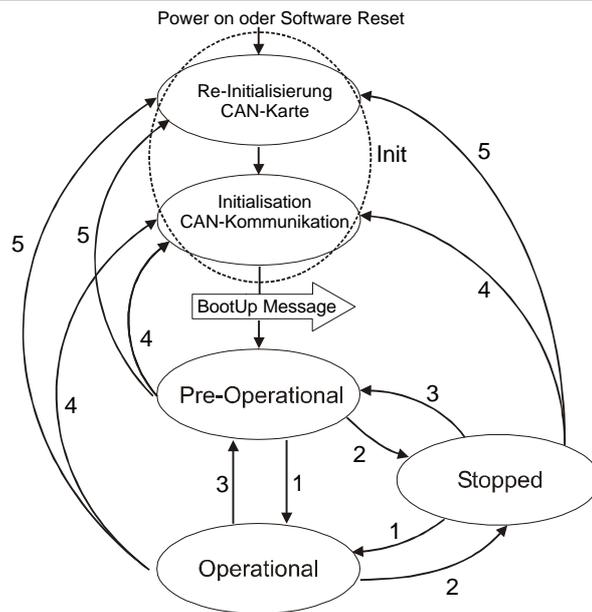


Abb. 1: CAN-Zustands-Diagramm

#### 4.5.4. Die verschiedenen NMT-Zustände

##### Init:

Nach dem initialisieren meldet sich der Geber mit einer Boot-Up-Meldung am CAN-Bus an. Danach wechselt der Geber automatisch in den Zustand „Pre-Operational“.

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

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

##### Pre-Operational Mode:

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

##### Operational Mode:

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

##### Stopped Mode:

Im Stopped Mode ist nur noch NMT-Kommunikation möglich. Es können keine SDO Parameter gelesen oder geschrieben werden.

#### 4.5.5. Zustandswechsel

##### Start Remote Node (1)

Mit dem „Start\_Remote\_Node“-Befehl wird der Geber in den Zustand „Operational Mode“ gebracht.

COB-ID	Kommando Byte	Knotennummer
0h	1h	0h .. 1Fh (0 .. 31)

**Stop Remote Node (2)**

Mit dem „Stop\_Remote\_Node“-Befehl wird der Geber in den Zustand „Stopped“ gebracht.

COB-ID	Kommando Byte	Knotennummer
0h	2h	0h .. 1Fh (0 .. 31)

**Enter\_PRE-OPERATIONAL-Mode (3)**

den Zustand „Pre-Operational“ wechseln.

COB-ID	Kommando Byte	Knotennummer
0h	80h	0h .. 1Fh (0 .. 31)

**Re-Initialisierung CAN-Parameter (4)**

COB-ID	Kommando Byte	Knotennummer
0h	81h	0h .. 1Fh (0 .. 31)

**Re-Initialisierung CAN-Karte (5)**

COB-ID	Kommando Byte	Knotennummer
0h	82h	0h .. 1Fh (0 .. 31)

**4.5.6. Heartbeat**

Zur Sicherstellung der Funktionsfähigkeit von CANopen-Netzknoten sind zwei optionale Überwachungsmechanismen vorgesehen: Ein Netzknoten kann über das sog. „Node-Guarding“ von einem übergeordneten NMT-Master überwacht werden oder alternativ seine Kommunikationsfähigkeit durch zyklisches Senden einer sog. „Heartbeat“-Nachricht bekannt geben.

Beim WV/WH58M ist ausschließlich das „Heartbeat“-Prinzip vorgesehen.

Diese Nachricht kann von einem oder mehreren anderen Netzteilnehmern empfangen werden und damit den zugeordneten Teilnehmer überwachen.

Im Objekt 1017h, „Producer Heartbeat Time“ kann die Zeit des Heartbeatintervalls hinterlegt werden. Ein Wert von 0 schaltet den Heartbeat ab.

Die Heartbeat-Meldung besteht aus der COB-ID und einem zusätzlichen Byte. In diesem Byte wird der aktuelle NMT-Zustand hinterlegt.

COB-ID	Byte 0
700h + Node-ID	NMT-Zustand

**NMT-Zustand:**

- 0: Boot-Up
- 4: Stopped
- 5: Operational
- 127: Pre-Operational

## 5. Objektverzeichnis

Im Objektverzeichnis eines CANopen-Gerätes sind alle Eigenschaften und Parameter dieses Gerätes abgelegt.



Bestimmte Parameter des Objektverzeichnisses sind spannungsausfallsicher in einem nichtflüchtigen Speicher des Gebers hinterlegt und werden beim Power-On oder Re-Initialisierung in den Arbeitsspeicher kopiert.

Der Zugriff auf das Objektverzeichnis erfolgt über die in *Kapitel 4.3 „Übertragung der SDO-Daten (Parametrierung)“* beschriebenen SDO-Dienste.

Das Objektverzeichnis ist in drei verschiedenen Bereiche gegliedert:

- Standardobjekte, die für alle CANopen-Geräte gültig sind, 1h .. 1FFFh, (CiA DS 301)
- Herstellerspezifische Objekte, 2000h .. 5FFFh
- Gerätespezifische Objekte, 6000h .. BFFFh, (CiA DS 406)

Die Adresse (Index), die auf jeden Eintrag im Objektverzeichnis zeigt, ist ebenfalls mit Ausnahme des Herstellerspezifischen Bereichs in den Profilen genormt. So ist sichergestellt, dass alle Geräte die im Profil beschriebenen Funktionen (Standard- und optionale Funktionen) immer unter dem gleichen Index liefern. Dies ist Voraussetzung für ein offenes System und für die Austauschbarkeit der Geräte.

Die Einträge im Objektverzeichnis werden durch einen 16 Bit Index adressiert. Jeder Index-Eintrag kann durch einen Sub-Index weiter untergliedert werden.

### 5.1. Objektübersicht

Index	Name	Beschreibung	siehe Seite
1000h	Device Type	gibt das Geräteprofil und den Gebertyp an	19
1001h	Error Register	zeigt Fehlerzustände des Gebers an	19
1002h	Manufacturer Status Register	zeigt den Inhalt der CAN-Bus-spezifischen „TransmitErrorCounter“ bzw. „ReceiveErrorCounter“ und die Firmware-Version der Grundkarte an	20
1003h	Pre-Defined Error Field	das Objekt speichert die 8 zuletzt aufgetretenen Fehlerzustände	20
1005h	COB-ID SYNC Message	Einstellung der COB-ID des SYNC-Objektes	21
1008h	Manufacturer Device Name	Kurzbezeichnung des Gerätetyps	21
1009h	Manufacturer Hardware Version	Hardwareversion des Gebers	21
100Ah	Manufacturer Software Version	Softwareversion des Gebers	22
1010h	Store Parameters	Das Objekt zeigt an, dass der Geber Parameter ohne Benutzereingabe nichtflüchtig abspeichert	22
1011h	Restore Parameters	Das Objekt zeigt an, dass der Geber nichtflüchtig gespeicherte Parameter automatisch lädt.	23
1014h	COB-ID Emergency Object	COB-ID des Emergency-Objekts	24

Index	Name	Beschreibung	siehe Seite
1017h	Producer Heartbeat Time	Einstellung der Zykluszeit des Heartbeat-Timers	24
1018h	Identity Objekt	enthält die von CiA vergebene Herstellernummer	25
1800h	Transmit PDO1 Communication Parameter	Transmit PDO für asynchrone Betriebsart (Timer- bzw. Positionswert-Gesteuert)	25
1801h	Transmit PDO2 Communication Parameter	Transmit PDO für synchrone Betriebsart, einschließlich Positionswertausgabe über RTR	26
1A00h	Transmit PDO1 Mapping Parameter		27
1A01h	Transmit PDO2 Mapping Parameter		28
2001h	Manufacturer Offset	Herstellerspezifischer Offsetwert (wird Geberintern zum Positionswert hinzuaddiert)	28
2002h	Geber Nullsetzen	Positionswert auf Wert 0 setzen (Voraussetzung: Presetwert = 0)	29
2800h	Sendewiederholungszähler für PDO1	gibt an, wie oft der PDO1 gesendet wird	29
2801h	Sendewiederholungszähler für PDO2	gibt an, wie oft der PDO2 gesendet wird	30
6000h	Operating Parameters	Einstellung von Drehrichtung und Skalierungsfunktion	30
6001h	Measuring units per Revolution	Parametrierung der Auflösung in Schritte / Umdrehung des Gebers	31
6002h	Total measuring range in measuring units	Parametrierung des Gesamtmessbereich des Gebers	31
6003h	Preset Value	Parametrierung eines Preset- (Kalibrier) Werts	32
6004h	Position Value	Positionswert (verrechnet mit Preset- und Manufacturer Offset-Wert)	32
6200h	Cycle Timer PDO1	Wert in ms, identisch mit Objekt 1800h, Subindex 5	33
6500h	Operating Status	zeigt die aktuell eingestellte Drehrichtung und Skalierungsfunktion an	33
6501h	Singleturn Resolution	zeigt die maximal mögliche Auflösung in Schritten / Umdrehung an	34
6502h	Number of distinguishable Revolutions	zeigt die maximal mögliche Anzahl Umdrehungen an	34
6503h	Alarms	Anzeige von Fehlerzuständen	34
6504h	Supported Alarms	gibt an, welche Alarmmeldungen unterstützt werden	35
6505h	Warnings	Anzeige von Warnungen	35
6506h	Supported Warnings	gibt an, welche Warnungen unterstützt werden	36
6507h	Profile and Software Version	zeigt die Versionsnummer des verwendeten Geräteprofils und die Versionsnummer der Geber-Firmware an	36

Index	Name	Beschreibung	siehe Seite
6508h	Operating Time	gibt den Wert FFFFFFFFh aus (Funktion wird momentan noch nicht unterstützt)	36
6509h	Offset Value	Entspricht dem Geber-Nullpunktwert	37
650Ah	Module Identification	über Subindices sind Gerätespezifische Parameter darstellbar (Manufacturer Offset, Manufacturer min position value, Manufacturer max position value)	37
650Bh	Serial Number	gibt den Wert FFFFFFFFh aus (Funktion wird momentan noch nicht unterstützt)	38

## 5.2. Detaillierte Objektbeschreibung

### 5.2.1. Objekt 1000h (Device Type)

<b>Subindex</b>	<b>00h</b>			
<b>Beschreibung</b>	Information über Gerätetyp und Geräteprofil			
<b>Zugriff</b>	ro			
<b>Datentyp</b>	UNSIGNED 32			
<b>EEPROM</b>	no			
<b>Default</b>	Multiturn: <b>00030196h</b> Singleturn: <b>00010196h</b>			
<b>Dateninhalt</b>	<b>Geräteprofil-Nummer</b>		<b>Gebertyp</b>	
	Byte 0	Byte 1	Byte 2	Byte 3
	<b>96h</b>	<b>01h</b>	<b>03h</b>	<b>00h</b>

**0196h** (= 406): CANopen Device Profile for Encoders, Version 3.01

**0003h:** Single Turn-Winkelkodierer, absolut, mit Batteriegepufferten elektronischen Umdrehungszähler (Multiturn)

**0001h:** Single Turn-Winkelkodierer, absolut

### 5.2.2. Objekt 1001h (Error Register)

<b>Subindex</b>	<b>00h</b>	
<b>Beschreibung</b>	Aufretende Gerätefehler werden hier angezeigt	
<b>Zugriff</b>	ro	
<b>Datentyp</b>	UNSIGNED 8	
<b>EEPROM</b>	no	
<b>Default</b>	no	
<b>Dateninhalt</b>	<b>Bit</b>	<b>Bedeutung</b>
	<b>0</b>	gesetztes Bit zeigt das auftreten irgendeiner Fehlerbedingung an
	<b>4</b>	gesetztes Bit zeigt Kommunikationsfehler auf dem CAN-Bus an (Acknowledgement-, Form-, CRC- und Stuffbit-Fehler)
	<b>1-3, 5-7</b>	nicht verwendet

### 5.2.3. Object 1002h (Manufacturer Status Register)

<b>Subindex</b>	<b>00h</b>			
<b>Beschreibung</b>	Die Zählerstände der Register „Transmit Error Counter“ und „Receive Error Counter“ können über dieses Objekt gelesen werden. Die Inhalte dieser Register geben Aufschluss über die am Montageort des Gebers herrschenden Übertragungsstörungen. Zusätzlich wird in den Bytes 2 und 3 der Versionsstand der Grundkarten-Firmware ausgegeben.			
<b>Zugriff</b>	ro			
<b>Datentyp</b>	UNSIGNED 32			
<b>EEPROM</b>	no			
<b>Default (Multiturn-Geber)</b>	02070000h			
<b>Default (Singleturn-Geber)</b>	02080000h			
<b>Dateninhalt</b>	Byte 0	Byte 1	Byte 2	Byte 3
	Receive Error Counter	Transmit Error Counter	Firmwarestand Grundkarte LOW	Firmwarestand Grundkarte HIGH

Details zu den genannten Zählern sind den einschlägigen CAN-Bus-Publikationen zu entnehmen.

### 5.2.4. Object 1003h (Pre-defined Error Field)

- Dieses Objekt speichert die 8 zuletzt aufgetretenen Fehlerzustände.
- Der Eintrag unter Subindex 0 zeigt die Anzahl der gespeicherten Fehler an.
- Jeder neu hinzugekommene Fehlerzustand wird unter Subindex 1 gespeichert. Vorangegangene Fehlermeldungen rutschen in der Position um eine Stelle nach unten.
- Die gesamte Fehlerliste wird durch Schreiben des Wertes 0 bei Subindex 0 gelöscht.
- Die Einträge in der Fehlerliste besitzen das Format wie unter *Kap. 4.4, Emergency-Dienst* beschrieben.

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Anzahl der gespeicherten Fehlermeldungen
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“)
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	no
<b>Default</b>	0
<b>Wertebereich</b>	<b>0 – 8</b>

<b>Subindex</b>	<b>01h .. 08h</b>
<b>Beschreibung</b>	Aufgetretene Fehlermeldungen
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	no
<b>Default</b>	0

### 5.2.5. Object 1005h (COB-ID SYNC message)

<b>Subindex</b>	<b>00h</b>	
<b>Beschreibung</b>	Definiert die COB-ID des Synchronisations Objekts (SYNC)	
<b>Zugriff</b>	rw (beschreibbar nur im Zustand „Pre-Operational“)	
<b>Datentyp</b>	UNSIGNED 32	
<b>EEPROM</b>	yes	
<b>Default</b>	<b>80h</b>	
<b>Dateninhalt</b>	<b>Bit 31</b>	nicht definiert
	<b>Bit 30</b>	0: Geber generiert keine SYNC-Meldung, 1: Geber generiert SYNC-Meldungen
	<b>Bit 29</b>	0: 11Bit-Identifizier (CAN 2.0A) 1: 29Bit-Identifizier (CAN 2.0B)
	<b>Bit 28..11</b>	0: falls Bit 29 = 0, X: falls Bit 29 = 1: Bits 28 – 11 des 29Bit SYNC-COB-ID
	<b>Bit 10..0</b>	X: Bits 10 – 0 des SYNC-COB-ID

### 5.2.6. Object 1008h (Manufacturer Device Name)

<b>Subindex</b>	<b>00h</b>			
<b>Beschreibung</b>	Geberkurzbezeichnung in ASCII			
<b>Zugriff</b>	const			
<b>Datentyp</b>	Visible_String			
<b>EEPROM</b>	no			
<b>Default</b>	W58M			
<b>Dateninhalt</b>	Byte 0	Byte 1	Byte 2	Byte 3
	<b>57h (,W')</b>	<b>35h (,5')</b>	<b>38h (,8')</b>	<b>4Dh (,M')</b>

### 5.2.7. Object 1009h (Manufacturer Hardware Version)

<b>Subindex</b>	<b>00h</b>			
<b>Beschreibung</b>	Hardwareversion in ASCII			
<b>Zugriff</b>	const			
<b>Datentyp</b>	Visible_String			
<b>EEPROM</b>	no			
<b>Default</b>	„1.00“			
<b>Dateninhalt</b>	Byte 0	Byte 1	Byte 2	Byte 3
	<b>31h (,1')</b>	<b>2Ehh (,.)'</b>	<b>30h (,0')</b>	<b>30h (,0')</b>

### 5.2.8. Object 100Ah (Manufacturer Software Version)

<b>Subindex</b>	<b>00h</b>			
<b>Beschreibung</b>	Softwareversion in ASCII			
<b>Zugriff</b>	const			
<b>Datentyp</b>	Visible_String			
<b>EEPROM</b>	no			
<b>Default</b>	„3.00“			
<b>Dateninhalt</b>	Byte 0	Byte 1	Byte 2	Byte 3
	<b>33h (,3')</b>	<b>2Eh (,.)</b>	<b>30h (,0')</b>	<b>30h (,0')</b>

### 5.2.9. Object 1010h (Store Parameters)

Dieses Objekt dient ausschließlich als Hinweis, dass der Geber bestimmte Parameter automatisch ins interne EEPROM abspeichert. Das „Store-Parameter“-Kommando ist hierzu nicht nötig!

<b>Subindex</b>	<b>00h</b>		
<b>Beschreibung</b>	beschreibt die Anzahl der in Subindex 1 stehenden Einträge		
<b>Zugriff</b>	ro		
<b>Datentyp</b>	UNSIGNED 8		
<b>EEPROM</b>	no		
<b>Default</b>	1h		

<b>Subindex</b>	<b>01h</b>		
<b>Beschreibung</b>	beschreibt das Verhalten des Gebers, wie Parameter im EEPROM abgespeichert werden.		
<b>Zugriff</b>	ro		
<b>Datentyp</b>	UNSIGNED 32		
<b>EEPROM</b>	no		
<b>Default</b>	2h		
<b>Dateninhalt</b>	<b>Bit 31-2</b>	0	
	<b>Bit 1</b>	0:	Geber speichert Parameter nicht automatisch
		1:	Geber speichert Parameter automatisch nach Schreibzugriff auf entsprechendes Objekt
<b>Bit 0</b>	0:	Geber speichert Parameter nicht per Befehl	
	1:	Geber speichert Parameter auf Kommando	

Die Parameter, die im EEPROM gespeichert werden, sind in nachfolgender Tabelle dargestellt:

Objekt	Subindex	Beschreibung	Defaultwert
1005h	0h	SYNC-ID	80h
1014h	0h	EMCY-ID	80h + Node-ID
1017h	0h	Producer Heartbeat Time	0h
1800h	1h	PDO1-ID	40000180h + Node-ID
1800h	5h	PDO1 Event Timer	0h
1801h	1h	PDO2-ID	80000280h + Node-ID
1801h	2h	PDO2 Transmission Type	1h
2001h	0h	Manufacturer Offset	0h
2800h	0h	PDO1 Sende-Wiederholungszähler	0h
2801h	0h	PDO1 Sende-Wiederholungszähler	0h
6000h	0h	Operating Status	0h
6001h	0h	Geberauflösung	10Bit-Geber: 1024 12Bit-Geber: 4096
6002h	0h	Gesamtmessbereich	10Bit-Geber: 4194304 12Bit-Geber: 16777216
6003h	0h	Presetwert	0h
6200h	0h	PDO1 Event Timer	siehe Objekt 1800-5

### 5.2.10. Object 1011h (Load Default Parameters)

Über dieses Objekt wird der Geber auf seine Default-Werte gesetzt (siehe 5.2.9). Um eine gewisse Sicherheit gegen unbeabsichtigtes laden der Defaultwerte zu haben, muss in Subindex 1h der String „load“ geschrieben werden:

COB-ID	Kommando	Index Low	Index High	Subindex	Data 0 (LSB)	Data 1	Data 2	Data 3 (MSB)
600h+ Node-ID	23h	11h	10h	01h	‚l‘ (6Ch)	‚o‘ (6Fh)	‚a‘ (61h)	‚d‘ (64h)

Ein Lesezugriff auf die entsprechenden Subindizes ergibt die nachfolgend dargestellten Werte:

<b>Subindex</b>	00h
<b>Beschreibung</b>	zeigt den größten, unterstützten Subindex an
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	no
<b>Default</b>	1h

<b>Subindex</b>	<b>01h</b>	
<b>Beschreibung</b>	Es werden alle Default-Werte geladen	
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“)	
<b>Datentyp</b>	UNSIGNED 32	
<b>EEPROM</b>	no	
<b>Default</b>	0h	
<b>Dateninhalt</b>	<b>Bit 31-1</b>	0
	<b>Bit 0</b>	0: Geber lässt ein laden von Default-Parameter nicht zu. 1: Geber lässt das laden von Default-Parameter zu.

### 5.2.11. Object 1014h (COB-ID Emergency Object)

<b>Subindex</b>	<b>00h</b>	
<b>Beschreibung</b>	Definiert die COB-ID des Emergency Objekts (EMCY)	
<b>Zugriff</b>	rw (beschreibbar nur im Zustand „Pre-Operational“)	
<b>Datentyp</b>	UNSIGNED 32	
<b>EEPROM</b>	yes	
<b>Default</b>	<b>80h + Node-ID</b>	
<b>Dateninhalt</b>	<b>Bit 31</b>	0: EMCY-Objekt existiert / ist gültig 1: EMCY-Objekt existiert nicht / ungültig
	<b>Bit 30</b>	immer 0
	<b>Bit 29</b>	0: 11Bit-Identifizier (CAN 2.0A) 1: 29Bit-Identifizier (CAN 2.0B)
	<b>Bit 28..11</b>	0: falls Bit 29 = 0, X: falls Bit 29 = 1: Bits 28 – 11 des 29Bit EMCY-COB-ID
	<b>Bit 10..0</b>	X: Bits 10 – 0 des EMCY-COB-ID

### 5.2.12. Object 1017h (Producer Heartbeat Time)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	definiert die Zykluszeit des Heartbeat-Überwachungsdienstes
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“)
<b>Datentyp</b>	UNSIGNED 16
<b>EEPROM</b>	yes
<b>Default</b>	0h
<b>Wertebereich</b>	10 .. 65535 (Ah .. FFFFh); Der Zahlenwert entspricht einem vielfachen von 1ms.  Durch schreiben des Werts 0 wird der Dienst ausgeschaltet.  Das schreiben von Werten im Bereich 1 .. 9 lösen eine Fehlermeldung aus!

### 5.2.13. Object 1018h (Identity Object)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Anzahl Einträge
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	no
<b>Default</b>	2h

<b>Subindex</b>	<b>01h</b>
<b>Beschreibung</b>	von der CiA vergebene Hersteller-Identifikationsnummer (Vendor-ID) für die Fa. SIKO GmbH (siehe <a href="http://www.can-cia.org">www.can-cia.org</a> )
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	no
<b>Default</b>	<b>195h</b>

<b>Subindex</b>	<b>02h</b>			
<b>Beschreibung</b>	zeigt in ASCII-codiert die Gebervariante an.			
<b>Zugriff</b>	ro			
<b>Datentyp</b>	UNSIGNED 32			
<b>EEPROM</b>	no			
<b>Default</b>	10+12Bit-Variante: „1012“ 12+12Bit-Variante: „1212“ 10Bit-SingleTurn: „10ST“ 12Bit-SingleTurn: „12ST“			
<b>Dateninhalt (Beispiel: 12+12Bit-Variante)</b>	Byte 0	Byte 1	Byte 2	Byte 3
	32h (,2')	31h (,1')	32h (,2')	31h (,1')
<b>Beispiel: 10+12Bit-Variante</b>	32h (,2')	31h (,1')	30h (,0')	31h (,1')
<b>Beispiel: 10Bit-Singleturn</b>	54h (,T')	53h (,S')	30h (,0')	31h (,1')
<b>Beispiel: 12Bit-Singleturn</b>	54h (,T')	53h (,S')	32h (,2')	31h (,1')

### 5.2.14. Object 1800h (Transmit PDO1 Parameter, asynchrone Betriebsart)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	größter unterstützter Subindex
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	no
<b>Default</b>	5h

<b>Subindex</b>	<b>01h</b>
<b>Beschreibung</b>	COB-ID des PDO1
<b>Zugriff</b>	rw (beschreibbar nur im Zustand „Pre-Operational“)
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	yes
<b>Default</b>	<b>40000180h + Node-ID</b> Bit30 = 1: RTR für diesen PDO nicht freigegeben, Bit ist immer gesetzt

<b>Subindex</b>	<b>02h</b>
<b>Beschreibung</b>	Transmission Type
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	no
<b>Default</b>	<b>FEh (254)</b> PDO hat asynchrone Charakteristik (PDO's werden in Abhängigkeit vom „Event Timer“ gesendet). Dieser Wert ist nicht änderbar!

<b>Subindex</b>	<b>03h (wird nicht verwendet, Zugriff erzeugt Fehlermeldung)</b>
-----------------	--

<b>Subindex</b>	<b>04h (wird nicht verwendet, Zugriff erzeugt Fehlermeldung)</b>
-----------------	--

<b>Subindex</b>	<b>05h</b>
<b>Beschreibung</b>	Event Timer
<b>Zugriff</b>	rw (beschreibbar nur im Zustand „Pre-Operational“)
<b>Datentyp</b>	UNSIGNED 16
<b>EEPROM</b>	yes
<b>Wertebereich</b>	1 .. 65535 (1h .. FFFFh); Der Zahlenwert entspricht einem vielfachen von 1ms. Durch schreiben des Werts 0 wird der Dienst ausgeschaltet. Der Inhalt dieses Objektes ist identisch mit dem Objekt 6200h.

### 5.2.15. Object 1801h (Transmit PDO2 Parameter, synchrone Betriebsart)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	größter unterstützter Subindex
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	no
<b>Default</b>	5h

<b>Subindex</b>	<b>01h</b>	
<b>Beschreibung</b>	COB-ID des PDO2	
<b>Zugriff</b>	rw (beschreibbar nur im Zustand „Pre-Operational“)	
<b>Datentyp</b>	UNSIGNED 32	
<b>EEPROM</b>	yes	
<b>Default</b>	<b>80000280h + Node-ID</b>	Bit31 = 1: PDO2 ist nach Power-On (Init) immer gesperrt; muss explizit per SDO-Dienst frei geschaltet werden.

<b>Subindex</b>	<b>02h</b>	
<b>Beschreibung</b>	Transmission Type	
<b>Zugriff</b>	rw (beschreibbar nur im Zustand „Pre-Operational“)	
<b>Datentyp</b>	UNSIGNED 8	
<b>EEPROM</b>	yes	
<b>Default</b>	<b>1h</b>	PDO hat synchrone Charakteristik;
<b>Wertebereich</b>	<b>1h .. n .. F0h (240)</b>	auf jedes n-te SYNC-Kommando wird das PDO gesendet, abhängig noch vom Wert in Objekt 2801h.
	<b>FDh (253):</b>	Geber antwortet auf RTR-Anforderung.

<b>Subindex</b>	<b>03h (wird nicht verwendet, Zugriff erzeugt Fehlermeldung)</b>
-----------------	--

<b>Subindex</b>	<b>04h (wird nicht verwendet, Zugriff erzeugt Fehlermeldung)</b>
-----------------	--

<b>Subindex</b>	<b>05h (wird nicht verwendet, Zugriff erzeugt Fehlermeldung)</b>
-----------------	--

#### 5.2.16. Object 1A00h (Transmit PDO1 Mapping Parameter)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Anzahl der gemappten Objekte
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	no
<b>Default</b>	1h

<b>Subindex</b>	<b>01h</b>
<b>Beschreibung</b>	Beschreibt den Inhalt der PDO1-Meldung
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	no
<b>Default</b>	<b>60040020h</b>

### 5.2.17. Object 1A01h (Transmit PDO2 Mapping Parameter)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Anzahl der gemappten Objekte
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	no
<b>Default</b>	1h

<b>Subindex</b>	<b>01h</b>
<b>Beschreibung</b>	Beschreibt den Inhalt der PDO2-Meldung
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	no
<b>Default</b>	<b>60040020h</b>

### 5.2.18. Object 2001h (Manufacturer Offset)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Mit dem Offset ist es möglich, den skalierten Wertebereich zu verschieben. Der Offsetwert wird im Geber zum Positionswert hinzuaddiert. Es sind positive als auch negative Werte zugelassen. <b>HINWEIS:</b> Dieses Objekt ist bei den Single-Turn-Ausführungen des WV/WH58M <b>nicht</b> verfügbar!
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“, wenn das „Scaling Bit [siehe Objekt 6000h] gesetzt ist)
<b>Datentyp</b>	SIGNED 32
<b>EEPROM</b>	yes
<b>Default</b>	0h
<b>Wertebereich</b>	Der minimale bzw. maximale einzugebende Wert hängt von den in Objekt 650Ah, Subindex 2 bzw. Subindex 3 eingetragenen Werte ab. Diese bestimmen sich wiederum aus dem parametrisierten Wert der Gesamtschrittzahl: unterer_Grenzwert = - 1/2 Gesamtschrittzahl, oberer_Grenzwert = 1/2 Gesamtschrittzahl – 1 <b>unterer_Grenzwert &lt; Offset &lt; oberer_Grenzwert</b>



Die Ausführungszeit des Befehls „Manufacturer Offset“ beträgt ca. 36ms. Während dieser Verarbeitungszeit wird der Positionswert **nicht** aktualisiert!

### 5.2.19. Object 2002h (Geberwert Nullsetzen)

<b>Subindex</b>	00h
<b>Beschreibung</b>	Mit diesem Objekt ist es möglich den Geberwert zu „nullen“, d.h. den Positionswert auf 0 zu setzen (Voraussetzung: Presetwert = 0).
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“)
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	no
<b>Default</b>	no
<b>Wertebereich</b>	0 .. 1; schreiben des Wertes 1 auf Subindex 0 setzt den Positionswert auf 0. Ein erneutes Nullsetzen kann erst erfolgen wenn zuvor eine 0 geschrieben wurde.

Beispiel:

COB-ID	Kommando	Index Low	Index High	Subindex	Data 0 (LSB)	Data 1	Data 2	Data 3 (MSB)
600h+ Node-ID	23h	02h	20h	00h	01h	00h	00h	00h

daran anschließend muss das folgende Befehlstelegramm gesendet werden:

COB-ID	Kommando	Index Low	Index High	Subindex	Data 0 (LSB)	Data 1	Data 2	Data 3 (MSB)
600h+ Node-ID	23h	02h	20h	00h	00h	00h	00h	00h



Die Ausführungszeit des „Nullsetzbefehls“ beträgt ca. 40ms. Während dieser Bearbeitungszeit wird der Positionswert **nicht** aktualisiert!

### 5.2.20. Object 2800h (Sende-Wiederholungszähler für PDO1)

<b>Subindex</b>	00h
<b>Beschreibung</b>	Der Wert des Sendewiederholungszähler für PDO1 bestimmt, wie oft dieser PDO gesendet wird (siehe Kap. 4.2.2, <i>asynchrone Datenübertragung</i> ).
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“)
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	yes
<b>Default</b>	0h
<b>Wertebereich</b>	0 .. 100 (64h) Wert = 0: Wiederholungszähler ist ausgeschaltet

### 5.2.21. Object 2801h (Sende-Wiederholungszähler für PDO2)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Der Wert des Sendewiederholungszähler für PDO2 bestimmt, wie oft dieser PDO gesendet wird (siehe <i>Kap. 4.2.1, synchrone Datenübertragung</i> ).
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“)
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	yes
<b>Default</b>	0h
<b>Wertebereich</b>	<b>0 .. 100 (64h)</b> Wert = 0: Wiederholungszähler ist ausgeschaltet

### 5.2.22. Object 6000h (Operating Parameters)

<b>Subindex</b>	<b>00h</b>				
<b>Beschreibung</b>	Über dieses Objekt wird die Codefolge des Gebers und die Skalierungsfunktion beeinflusst.				
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“)				
<b>Datentyp</b>	UNSIGNED 16				
<b>EEPROM</b>	yes				
<b>Default</b>	0h				
<b>Bitdefinition</b>		Bit 14 .. Bit 3	Bit 2	Bit 1	Bit 0
	Funktion	nicht verwendet	Skalierung	nicht verwendet	Codefolge
	Bit = 0	-	gesperrt	-	Drehrichtung I (CW)
	Bit = 1	-	freigegeben	-	Drehrichtung E (CCW)

#### Erläuterungen zu den Funktionen:



Drehrichtung I:	aufsteigende Positionswerte bei Drehung des Gebers <u>im</u> Uhrzeigersinn (Clockwise, CW, Blick auf Geberwelle),
Drehrichtung E:	aufsteigende Positionswerte bei Drehung des Gebers <u>entgegen</u> dem Uhrzeigersinn (Counter Clockwise, CCW, Blick auf Geberwelle)
Skalierung gesperrt:	Der Geber arbeitet mit seiner vollen Auflösung (1024 Schritte/Umdrehung und 4096 Umdrehungen bzw. 4096 Schritte/Umdrehung und 4096 Umdrehungen)
Skalierung freigegeben:	Der Geber kann mit den Objekten 6001h (Measuring units per revolution), 6002h (Total Measuring range), 6003h (Preset) und 2001h (Manufacturer Offset) parametrisiert werden.

Das Setzen des Skalierungsbits hat zur Folge, dass der Preset- und der Manufacturer Offset-Wert auf 0 zurückgesetzt wird.

Das Skalierungsbit wird nichtflüchtig gespeichert. Somit sind vorgenommene Einstellungen nach Wiedereinschalten des Gebers weiterhin vorhanden.

Wird das Skalierungsbit von 1 auf 0 zurückgesetzt, werden mit den Objekten 6001h und 6002h vorgenommene Einstellungen mit den Defaultwerten überschrieben (Auflösung = 1024 bzw. 4096 Schritte/Umdrehung und 4096 Umdrehungen). Die Werte für Preset und ManufacturerOffset werden nicht verändert.



Die Ausführungszeit des Befehls „Operating Parameters“ ist von der Kombination der oben angegebenen Bits abhängig:

schreiben Bit0 = 0, Bit2 = 0: ca. 190ms

schreiben Bit0 = 1, Bit2 = 0: ca. 190ms

schreiben Bit0 = 0, Bit2 = 1: ca. 77ms

schreiben Bit0 = 1, Bit2 = 1: ca. 77ms

Während dieser Verarbeitungszeit wird der Positionswert **nicht** aktualisiert!

### 5.2.23. Object 6001h (Measuring Units per Revolution [Geberauflösung])

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Dieser Parameter stellt die gewünschte Auflösung pro Umdrehung ein.
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“, wenn das „Scaling Bit [siehe Objekt 6000h] gesetzt ist)
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	yes
<b>Default</b>	WV/WH58M-10Bit: <b>1024</b> WV/WH58M-12Bit: <b>4096</b>
<b>Wertebereich</b>	1 .. 1024 (WV/WH58M-10Bit) 1 .. 4096 (WV/WH58M-12Bit)



Ein eventuell eingestellter Preset- und/oder ManufacturerOffset-Wert wird bei Änderung der Auflösung auf 0 zurückgesetzt!



Die Ausführungszeit des Befehls „Measuring Units per Revolution“ beträgt ca. 160ms. Während dieser Verarbeitungszeit wird der Positionswert **nicht** aktualisiert!

### 5.2.24. Object 6002h (Total Measuring Range [Gesamtschrittzahl])

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Dieser Parameter stellt die Gesamtanzahl der Messschritte ein.
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“, wenn das „Scaling Bit [siehe Objekt 6000h] gesetzt ist)
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	yes
<b>Default</b>	WV/WH58M-10Bit: <b>4194304</b> WV/WH58M-12Bit: <b>16777216</b>
<b>Wertebereich</b>	4096 .. 4194304 (WV/WH58M-10Bit) 4096 .. 16777216 (WV/WH58M-12Bit)



Der parametrierbare Wert „Gesamtschrittzahl“ muss der Bedingung  $2^X \cdot \text{Geberauflösung}$  genügen ( $X = 1 \dots 12$  bei 12Bit-Geber bzw.  $1 \dots 10$  bei 10Bit-Geber).

Ein eventuell eingestellter Preset- und/oder ManufacturerOffset-Wert wird bei Änderung der Gesamtschrittzahl auf 0 zurückgesetzt!

Der Versuch, bei einem SingleTurn-Geber auf dieses Objekt zu schreiben, wird mit einer Fehlermeldung beantwortet (Fehlercode: 06040043h).



Die Ausführungszeit des Befehls „Total Measuring Range“ beträgt ca. 160ms. Während dieser Verarbeitungszeit wird der Positionswert **nicht** aktualisiert!

### 5.2.25. Object 6003h (Preset value)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Der Positionswert des Gebers wird auf diesen Preset-Wert eingestellt.
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“, wenn das „Scaling Bit [siehe Objekt 6000h] gesetzt ist)
<b>Datentyp</b>	SIGNED 32
<b>EEPROM</b>	yes
<b>Default</b>	0h
<b>Wertebereich</b>	<p>Der Wertebereich des Presetwertes hängt von den mit Objekten 6001h und 6002h vorgenommenen Einstellungen ab. In der Defaulteinstellung dieser beiden Objekte umfasst der Wertebereich folgenden Bereich:</p> <p>WV/WH58M-10Bit: <b>-2097152 .. 0 .. +2097151</b></p> <p>WV/WH58M-12Bit: <b>-8388608 .. 0 .. +8388607</b></p> <p>Der maximal darzustellende Wertebereich kann mit dem Objekt 650Ah, Subindex 2 und Subindex 3 gelesen werden und ist immer von den mit den Objekten 6001h und 6002h vorgenommenen Einstellungen abhängig!</p>



Beim setzen des Skalierungsbits, sowie bei Änderungen der Geberauflösung bzw. der Gesamtschrittzahl wird der Presetwert auf 0 zurückgesetzt!

Die Ausführungszeit des Befehls „Preset Value“ beträgt ca. 160ms. Während dieser Verarbeitungszeit wird der Positionswert **nicht** aktualisiert!

### 5.2.26. Object 6004h (Position value)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Dieses Objekt liefert den mit Skalierungsfaktoren, Preset- und Manufacturer Offset verrechneten Positionswert des Gebers.
<b>Zugriff</b>	ro
<b>Datentyp</b>	SIGNED 32
<b>EEPROM</b>	no

Der Positionswert des WV/WH58M berechnet sich nach folgender Formel:

$$\text{Positionswert} = (\text{Geberwert} - \text{Gebernulungswert}) * \text{RF} + \text{Presetwert} + \text{ManufacturerOffset}$$

Geberwert: von der Gebersensorik errechneter Absolutwert,  
 Gebernulungswert: Absolutwert zum Zeitpunkt der Nullung,  
 RF: Rechen- (Skalierungs-) Faktor = Geberauflösung / 1024; (10Bit),  
 Rechen- (Skalierungs-) Faktor = Geberauflösung / 4096; (12Bit),  
 Presetwert: siehe Objekt 6003h,  
 ManufacturerOffset: siehe Objekt 2001h

Der Gesamtmessbereich wird beim WV/WH58M in einen negativen und positiven Wertebereich aufgeteilt:

$$-1/2 \text{ Gesamtmessbereich} \dots 0 \dots +1/2 (\text{Gesamtmessbereich} - 1)$$

Die Darstellung des Positionswertes erfolgt demzufolge im 2er-Komplementformat in einer vorzeichenbehafteten 32 Bit-Zahl.

#### 5.2.27. Object 6200h (Zyklus Timer)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Definiert die Zykluszeit, mit der der PDO1 ausgegeben wird. Der Wert ist fest mit dem unter Objekt 1800h, Subindex 5 angegebenen Wert verknüpft (identisch).  Die Timer-gesteuerte Ausgabe wird aktiv, sobald eine Zykluszeit innerhalb des Wertebereichs parametrisiert wurde und der Geber in den Operational Mode geschaltet wurde.
<b>Zugriff</b>	rw (beschreibbar im Zustand „Pre-Operational“ und „Operational“)
<b>Datentyp</b>	UNSIGNED 16
<b>EEPROM</b>	no
<b>Default</b>	0h
<b>Wertebereich</b>	0: Zyklustimer ist ausgeschaltet, 1 .. 65535: Zykluszeit in ms

#### 5.2.28. Object 6500h (Operating Status)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Das Objekt zeigt die mit Objekt 6000h programmierten Einstellungen an.
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 16
<b>EEPROM</b>	yes
<b>Default</b>	no
<b>Bitdefinition</b>	siehe Objekt 6000h

### 5.2.29. Object 6501h (SingleTurn resolution)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Das Objekt zeigt die maximal mögliche Auflösung des Gebers an.
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	no
<b>Default</b>	WV/WH58M (10+12Bit): <b>1024</b> WV/WH58M (10Bit ST): <b>1024</b> WV/WH58M (12+12Bit): <b>4096</b> WV/WH58M (12Bit ST): <b>4096</b>

### 5.2.30. Object 6502h (Number of distinguishable revolutions)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Das Objekt zeigt die maximal mögliche Anzahl Umdrehungen des Gebers an.
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 16
<b>EEPROM</b>	no
<b>Default</b>	<b>4096</b>

### 5.2.31. Object 6503h (Alarms)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Zusätzlich zu den Fehlern, die über die Emergency-Messages gemeldet werden, bietet dieses Objekt weitere, Geberspezifische Fehlermeldungen. Im Fehlerfall wird das zugehörige Bit auf 1 gesetzt.
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 16
<b>EEPROM</b>	no
<b>Default</b>	0h

Bitdefinition:

Bit	Funktion	Wert = 0	Wert = 1
0	Positionsfehler	kein Fehler	Positionswert ungültig
1..11	nicht verwendet	-	-
12	Verbindung zur Grundkarte	Verbindung vorhanden	keine Grundkarte erkannt
13	Timeoutfehler in Verbindung zur Grundkarte	keine Störung	Verbindungsstörungen
14	Batteriewarnung	Batteriespannung OK	Batteriespannung nähert sich dem untersten, tolerierbaren Wert
15	Batteriefehler	Batteriespannung OK bzw. im noch tolerierbaren Bereich	Batterie ist entladen

### 5.2.32. Object 6504h (Supported Alarms)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Das Objekt zeigt an, welche Alarmmeldungen unterstützt werden.
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 16
<b>EEPROM</b>	no
<b>Default</b>	<b>F001h</b>

Bit 0:	Positionsfehler
Bit 12:	Verbindung zur Grundkarte
Bit 13:	Kommunikationsfehler mit Grundkarte (Timeout)
Bit 14:	Batteriewarnung
Bit 15:	Batteriealarm

### 5.2.33. Object 6505h (Warnings)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Warnmeldungen zeigen an, dass Toleranzen interner Geberparameter überschritten sind. Bei einer Warnmeldung kann der Positionswert, anders als bei einer Alarmmeldung, trotzdem gültig sein.
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 16
<b>EEPROM</b>	no
<b>Default</b>	0h

Bitdefinition:

Bit	Funktion	Wert = 0	Wert = 1
0 .. 3	nicht verwendet	-	-
4	Batteriewarnung	Batteriespannung OK	Batteriespannung nähert sich dem untersten, tolerierbaren Wert
5 .. 11	nicht verwendet	-	-
12	Störungen im Datenverkehr mit der Grundkarte	keine Störungen	Checksummenfehler aufgetreten
13	Schalterstellung DIP-Schalter 1	Schalter steht auf OFF	Schalter steht auf ON; Kommunikation mit Grundkarte wurde manuell unterbrochen
14, 15	nicht verwendet	-	-

### 5.2.34. Object 6506h (Supported Warnings)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Das Objekt zeigt an, welche Warnmeldungen unterstützt werden.
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 16
<b>EEPROM</b>	no
<b>Default</b>	<b>3010h</b>

- Bit 4: Batterie Warnung wird unterstützt  
 Bit 12: Kommunikationswarnung (Checksummenfehler)  
 Bit 13: Zustand DIP-Schalter 1

### 5.2.35. Object 6507h (Profile and Software Version)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Das Objekt zeigt das verwendete Geberprofil (CANopen Device profile for encoders) und die Versionsnummer des Firmware-Standes an.
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	no
<b>Default</b>	<b>03000301h</b>

Dateninhalt:

Profil Version		Firmware Version	
Byte 0 (Low)	Byte 1 (High)	Byte 2 (Low)	Byte 3 (High)
01h	03h	00h	03h

### 5.2.36. Object 6508h (Operating Time)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Betriebszeitähler (nicht im Geber implementiert)
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	no
<b>Default</b>	<b>FFFFFFFFh</b> (zeigt an, dass die Funktion nicht unterstützt wird)

### 5.2.37. Object 6509h (Gebernulungswert)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Über diese Objekt wird die Differenz zwischen Geberwert und dem skalierten und mit Preset- und/oder ManufacturerOffset verrechneten Positionswert ausgegeben.
<b>Zugriff</b>	ro
<b>Datentyp</b>	SIGNED 32
<b>EEPROM</b>	yes

$$\text{Gebernulungswert} = \text{Geberwert} - \frac{(\text{Positionswert} - \text{Presetwert} - \text{ManufacturerOffset})}{\text{Skalierungsfaktor}}$$

### 5.2.38. Object 650Ah (Module Identification)

Über dieses Objekt kann der Manufacturer Offsetwert (Subindex 1), der kleinste (Subindex 2) sowie der größte (Subindex 3) darstellbare Positionswert ausgelesen werden.

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Enthält die Anzahl weiterer Sub-Indizes.
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 8
<b>EEPROM</b>	no
<b>Default</b>	<b>3h</b>

<b>Subindex</b>	<b>01h</b>
<b>Beschreibung</b>	Herstellerspezifischer Offsetwert (dieser wird zum Positionswert hinzuaddiert).
<b>Zugriff</b>	ro
<b>Datentyp</b>	SIGNED 32
<b>EEPROM</b>	yes
<b>Default</b>	<b>0h</b>

<b>Subindex</b>	<b>02h</b>
<b>Beschreibung</b>	Minimal darstellbarer Positionswert (wird durch die Skalierungsobjekte 6001h und 6002h beeinflusst).
<b>Zugriff</b>	ro
<b>Datentyp</b>	SIGNED 32
<b>EEPROM</b>	no
<b>Default</b>	WV/WH58M (10+12Bit): <b>-2097152</b> WV/WH58M (10Bit ST): <b>0</b> WV/WH58M (12+12Bit): <b>-8388608</b> WV/WH58M (12Bit ST): <b>0</b>

<b>Subindex</b>	<b>03h</b>
<b>Beschreibung</b>	Maximal darstellbarer Positionswert (wird durch die Skalierungsobjekte 6001h und 6002h beeinflusst).
<b>Zugriff</b>	ro
<b>Datentyp</b>	SIGNED 32
<b>EEPROM</b>	no
<b>Default</b>	WV/WH58M (10+12Bit): <b>2097151</b> WV/WH58M (10Bit ST): <b>1024</b> WV/WH58M (12+12Bit): <b>8388607</b> WV/WH58M (12Bit ST): <b>4096</b>

### 5.2.39. Object 650Bh (Serial Number)

<b>Subindex</b>	<b>00h</b>
<b>Beschreibung</b>	Liefert die Seriennummer des Gebers (wird beim WV/WH58M nicht unterstützt).
<b>Zugriff</b>	ro
<b>Datentyp</b>	UNSIGNED 32
<b>EEPROM</b>	no
<b>Default</b>	<b>FFFFFFFFh</b> (Funktion ist nicht implementiert)

## 6. Einstell- und Diagnoseelemente

### 6.1. Einstellung des Node-Identifiers (Node-ID)

Nach Abnahme des sich auf der Geberhaube befindlichen Schraubverschlusses sind ein 8-poliger DIP-Schalter sowie zwei Diagnose-LED's (gelb und grün) sichtbar.

Zur Einstellung des Node-Identifiers dienen die Schalter 4 bis 8. Der einstellbare Bereich liegt zwischen 1 und 31. Die Einstellung der Schalter 4 bis 8 auf ON ist zwar möglich, wird aber Geberintern in die Node-ID 1 umgewandelt, weil der Identifier 0 nicht erlaubt ist!

Die Kodierung der Node-ID mit den Schaltern 4 bis 8 erfolgt im Binärformat. Die nachfolgende Tabelle verdeutlicht dies:

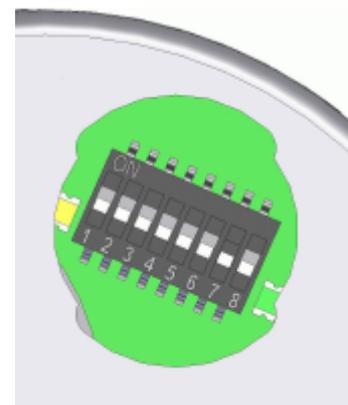


Abb. 2: DIP-Schalter und Diagnose-LED's

Schalter 4	Schalter 5	Schalter 6	Schalter 7	Schalter 8	eingestellte Node-ID
ON	ON	ON	ON	ON	1 (!)
ON	ON	ON	ON	OFF	1
ON	ON	ON	OFF	ON	2
ON	ON	ON	OFF	OFF	3
:	:	:	:	:	:

Schalter 4	Schalter 5	Schalter 6	Schalter 7	Schalter 8	eingestellte Node-ID
OFF	OFF	OFF	ON	OFF	29
OFF	OFF	OFF	OFF	ON	30
OFF	OFF	OFF	OFF	OFF	31



Die DIP-Schaltereinstellungen werden nur beim Starten des Gebers (Power-On oder bei Neuinitialisierung über ein NMT-Kommando) gelesen. Eine Änderung der Schalterstellungen im Betrieb des Gebers hat keine Auswirkungen. (Ausnahme: DIP-Schalter 1)

Standardeinstellung bei Geberauslieferung ist Node-ID 1 (DIP-Schalter 8 = OFF).

## 6.2. Einstellung der Baudrate

Mit den DIP-Schaltern 2 und 3 erfolgt die Einstellung der CAN-Baudrate. Es stehen 4 verschiedene Baudraten zur Auswahl. Die Zuordnung zeigt die nachfolgende Tabelle:

Schalter 2	Schalter 3	Baudrate
ON	ON	125 kBit/s
ON	OFF	250 kBit/s
OFF	ON	500 kBit/s
OFF	OFF	1000 kBit/s



Die DIP-Schaltereinstellungen werden nur beim Starten des Gebers (Power-On oder bei Neuinitialisierung über ein NMT-Kommando) gelesen. Eine Änderung der Schalterstellungen im Betrieb des Gebers hat keine Auswirkungen. (Ausnahme: DIP-Schalter 1)

Standardeinstellung bei Geberauslieferung ist die Baudrate 125 kBit/s.



DIP-Schalter 1 ist ausschließlich für Servicezwecke vorgesehen und muss in Stellung ON verbleiben.

## 6.3. Diagnose LED's

Neben dem DIP-Schalter sind jeweils eine grüne und eine gelbe LED angeordnet. Hiermit werden verschiedene Betriebszustände signalisiert:

	LED	Bedeutung
Power (grün)	AUS	Spannungsversorgung fehlt
	EIN	Spannungsversorgung ist OK
Status (gelb)	blinkend (Tastverhältnis 1:4)	Geber befindet sich im Pre-Operational-Mode
	EIN	Geber befindet sich im Operational-Mode
	schnell blinkend (Tastverhältnis 1:1)	Geber befindet sich im Stopped-Mode

## 7. Inbetriebnahme

Vor der Inbetriebnahme des Gebers sind folgende Arbeiten durchzuführen:

- Einstellung der Node-ID (diese darf nur einmal im System vorkommen),
- Einstellung der für das System gültigen CAN-Baudrate,
- korrekter Anschluss der Versorgungs- und Bus-Leitungen

### 7.1. Einschalten der Versorgungsspannung

Nach dem Einschalten der Versorgungsspannung initialisiert sich der Geber, was durch zweimaliges Blinken der gelben LED (nur sichtbar nach Abnahme der Verschlusschraube an der Geberhaube) angezeigt wird.

Nach Abschluss der Initialisierungsprozedur sendet der Geber ein spezielles NMT-Kommando, die **Boot-Up-Message**, um dem System das Vorhandensein des Gebers mitzuteilen. Während der Initialisierung werden die Parameter des Objektverzeichnisses aus dem nichtflüchtigen Speicher in den Arbeitsspeicher des Controllers geladen.

Wurde der Geber noch nicht programmiert, sind alle Parameter auf ihre Default-Werte gesetzt, ansonsten arbeitet der Geber mit den zuletzt parametrisierten Daten.

Der Geber befindet sich nun im **Pre-Operational-Mode**. In diesem Zustand kann der Geber gemäß den Forderungen der Anwendung per SDO-Kommandos parametrisiert werden. Dies betrifft insbesondere die Art und Weise, wie der Geber seine Positionswerte dem System zur Verfügung stellt (asynchrone oder synchrone Datenübertragung).



**HINWEIS:** Wird als Positionswert der Zahlenwert 7FFFFFFh ausgelesen, so bedeutet dies, dass der Geberinterne CAN-Controller keine Verbindung mit der Gebergrundkarte herstellen konnte! Ursache hierfür kann sein, dass der DIP-Schalter 1 (siehe Abb2, Kap. 6.1) sich nicht in Stellung ON befindet oder dass Kontaktschwierigkeiten dieses Schalters eine Kommunikation des CAN-Controllers mit der Grundkarte verhindern.

### 7.2. Senden des Positionswertes

Bevor der Geber seinen Positionswert senden kann, muss mit dem NMT-Kommando **Node-Start** der Geber in den **Operational-Mode** geschaltet werden.

COB-ID	Kommando Byte	Knotennummer
0h	1h	0h .. 1Fh (0 .. 31)

Wird als Knotennummer die Node-ID des Gebers angegeben, so startet nur dieser Geber. Wird für die Knotennummer der Wert 0 übertragen, starten alle an dem Bus angeschlossenen Geräte.

Nun kann der Geber über die mit PDO1 bzw. PDO2 spezifizierten Art und Weise seinen Positionswert übertragen:

#### 7.2.1. Asynchrone (zyklische) Übertragung

Für diese Übertragungsart ist der PDO1 verantwortlich. Der Positionswert wird, entsprechend der in Objekt 1800h, Subindex 5 parametrisierten Zeit, zyklisch gesendet.

Eine weitere Möglichkeit besteht über das Objekt 2800h. Wird dieses Objekt mit dem Wert 1 parametrierd, wird der Positionswert bei jeder Änderung desselben 1mal gesendet. Der Wert 3 hat zur Folge, dass der Positionswert bei einer Änderung 3mal gesendet wird.

### 7.2.2. Synchroner Übertragung

Damit der Positionswert synchron gesendet werden kann, muss der PDO2 zuerst freigeschaltet werden (Bit31 des COB-ID PDO2 auf 0 setzen). Des Weiteren muss in Objekt 1801h, Subindex 2 ein Wert zwischen 1 und 240 (= F0h) eingeschrieben sein.

Wenn nun der Wert z.B. 3 beträgt, wird der PDO2 auf jedes dritte SYNC-Telegramm gesendet (beim Wert = 1 wird auf jedes SYNC-Telegramm gesendet), solange in Objekt 2800h der Wert 0 eingeschrieben ist.

Ist das Objekt 2800h z.B. mit dem Wert 4 parametrierd, wird der PDO2 nach wie vor auf jedes dritte SYNC-Telegramm gesendet, aber insgesamt nur 4mal. Dem entsprechend folgt auf das 12. SYNC-Telegramm der letzte PDO2.

Der Zähler für die Anzahl der zu übertragenden PDO's wird bei einer Positionswertänderung oder durch das NMT-Kommando **Reset** zurückgesetzt. D.h. die Position wird, falls sie sich nicht ändert, 4mal gesendet. Ändert sich die Position, wird sie wieder 4mal gesendet.

Eine weitere Möglichkeit, den PDO2 zu übertragen, besteht in der Antwort auf eine RTR-Anforderung. Hierzu muss Objekt 1801h, Subindex 2 mit dem Wert 253 (=FDh) beschrieben werden.

### 7.3. Beenden der Positionswertübertragung

Um die Datenübertragung vom Geber zu stoppen, kann dieser entweder in den **Stopped-Mode** oder in den **Pre-Operational-Mode** zurückgeschaltet werden:

Kommando Stop-Mode

COB-ID	Kommando Byte	Knotennummer
0h	2h	0h .. 1Fh (0 .. 31)

Kommando Pre-Operational-Mode

COB-ID	Kommando Byte	Knotennummer
0h	80h	0h .. 1Fh (0 .. 31)

Über die Knotenadresse 0 werden alle an den Bus angeschlossenen Geräte angesprochen.