

Betriebsanleitung  
wireSENSOR CANopen

Schnittstellenanleitung

MICRO-EPSILON MESSTECHNIK  
GmbH & Co. KG  
Königbacher Str. 15

94496 Ortenburg / Deutschland

+49 (0) 8542 / 168-0

+49 (0) 8542 / 168-90

e-mail: [info@micro-epsilon.de](mailto:info@micro-epsilon.de)

[www.micro-epsilon.com/contact/worldwide/](http://www.micro-epsilon.com/contact/worldwide/)

<https://www.micro-epsilon.de>

## Inhaltsverzeichnis

1	Funktionsübersicht.....	4
2	Positionierung von CANopen im Schichtenmodell.....	5
3	Gerätemodell.....	6
4	Kommunikationsspezifische Standard-Objekte.....	7
5	Objektverzeichnis.....	8
5.1	Zugriff auf Parameterdaten per Service-Daten-Objekt (SDO).....	8
5.2	Kommunikationsparameter nach CIA DS-301.....	9
5.3	Herstellerspezifische Kommunikationsparameter.....	11
5.4	Kommunikationsparameter nach CIA DS-406.....	11
5.5	LSS-Dienste.....	12
5.5.1	Bitrate ändern.....	12
5.5.2	Node-ID ändern.....	13
5.6	Parametrisierung des Sensors.....	13
5.6.1	Preset.....	13
5.6.2	Umkehr Zählrichtung Messwerte.....	14
6	Prozessdatenobjekte: PDO (TxPDO1 - TxPDO2).....	15
6.1	Synchronisiertes Senden.....	15
6.2	Zyklisches Senden.....	15
7	Fehlermeldungen - EMCY-Codes.....	16
	Index.....	17

# 1 Funktionsübersicht

Der Sensor besitzt ein standardisiertes CANopen-Interface gemäß CiA-301 und ein CiA-DS406 Geräteprofil. Alle generierten Messwerte sind über das Objektverzeichnis zugänglich. Vorgenommene Einstellungen können im permanenten Speicher des Sensors gesichert werden.

Verfügbare Funktionalitäten:

- zwei Prozessdatenobjekte (TxPDO1 – TxPDO2)
- dynamisch mappbare Prozessdaten
- ereignisgesteuerter Prozessdatenversand ausgelöst durch Messdatenänderung
- intervallzeitgesteuerter Prozessdatenversand
- Senden von Prozessdaten als Reaktion auf dem Empfang eines SYNC-Telegramms
- ein Service-Datenobjekt (Standard-SDO)
- Überwachungsmechanismus Heartbeat
- Speicher- und Wiederherstellungsfunktion für im Gerät speicherbare Parameter
- Fehlermeldungen per Emergency-Objekt (EMCY)
- allgemeines Fehlerregister (Error Register)
- Fehlerliste (Pre-defined Error Field)

Im Bereich der geräteprofil- bzw. herstellerspezifischen Konfigurationsmöglichkeiten existieren:

- Einstellung der Node-ID und Baudrate mittels LSS (CiA DSP-305) oder Objektverzeichnis
- Richtungsumkehr der Messdaten
- Nullsetzen bzw. Preset

## 2 Positionierung von CANopen im Schichtenmodell

CANopen wurde im Verband „CAN in Automation“ (CiA) standardisiert und stellt einen offenen Protokollstandard in der Automatisierungstechnik unter Nutzung des CAN-Busses als Übertragungsmedium dar. Ebenso wie fast alle Feldbusse baut auch CANopen auf dem ISO/OSI 7-Schichtenmodell auf. CANopen definiert die Elemente für das Netzwerkmanagement, die Verwendung der CAN-Identifizier (Nachrichtenadresse), das zeitliche Verhalten auf dem Bus, die Art der Datenübertragung, sowie anwendungsbezogene Profile. Dadurch ist es möglich, dass CANopen Geräte unterschiedlicher Hersteller in einem Netzwerk kombiniert verwendet werden können. Den Application-Layer beschreibt CANopen als Kommunikationsprofil, um eine einheitliche Art der Kommunikation zu gewährleisten; dies wurde im CiA DS-301 durch die CiA spezifiziert. Zudem wurden auch verschiedene Geräte- und Anwendungsprofile definiert. Diese sind in den Standards CiA DS-4xx zu finden.

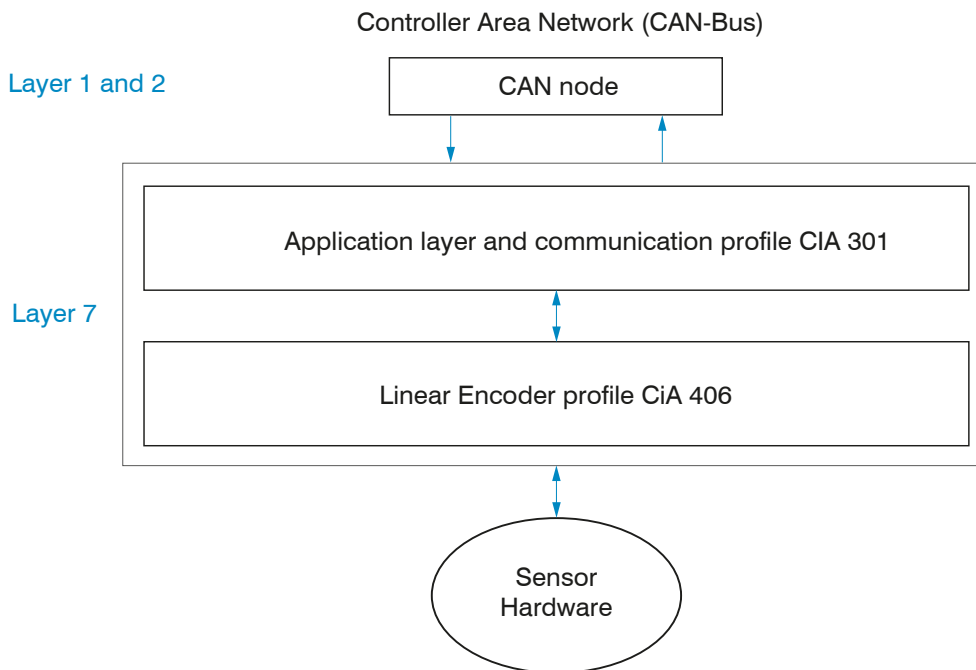


Abb. 2.1: Schichtenmodell der Schnittstelle CAN-Bus

### 3 Gerätemodell

Die Kommunikation mit einem CANopen-Gerät (CAN node), [siehe Abb. 4.1](#) erfolgt über sogenannte Datenobjekte.

Hierfür wurden von der CiA verschiedene Objekte für die unterschiedlichen Einsatzbereiche definiert.

Zum einen gibt es hoch priorisierte Prozessdatenobjekte (PDOs). Diese Telegramme dienen dem Austausch von Prozessdaten. Profilunabhängig gesehen, kann es sich bei PDOs um Telemetrie-Daten oder um Daten zur Parametrierung handeln. Üblicherweise erfolgt der Zugriff auf Parameter des Objektverzeichnisses eines Gerätes mittels Service-Datenobjekt (SDO).

Für das Netzwerkmanagement existieren die NMT-Objekte, mit dessen Hilfe der Zustandsautomat des CANopen-Gerätes gesteuert wird. Zusätzlich wird darüber die Zustandsüberwachung der Netzwerkknoten realisiert.

Es existieren noch weitere Objekte, welche für Synchronisation, Fehlermeldungen und Zeitstempeln definiert wurden. Jedes CANopen-Gerät besitzt ein eigenes Objektverzeichnis. In diesem sind die Parameter für alle CANopen-Objekte eingetragen.

Die elektronischen Datenblätter (EDS-Datei) finden Sie online unter <https://www.micro-epsilon.de/fileadmin/download/software/EDS-Device-Description-Datei-wireSENSOR.zip> Kopieren Sie die entpackte Datei in das benötigte Verzeichnis Ihrer Festplatte, bevor der Sensor konfiguriert werden kann. Löschen Sie eventuell vorhandene ältere Dateien. Konfigurationstools können EDS-Dateien einlesen und mit ihrer Hilfe mit dem jeweiligen Sensor kommunizieren und gegebenenfalls parametrisieren.

## 4 Kommunikationsspezifische Standard-Objekte

Die CAN-Identifizier der Kommunikationsobjekte (COB-IDs) werden entsprechend des Pre-Defined Connection Set, abhängig von der eingestellten Node-ID bestimmt. Diese ist bei den Sensoren werksseitig auf 01h eingestellt. Die Kommunikationsobjekte werden, wie folgt berechnet.

Kommunikationsobjekt (COB)	Standardwert	Berechnung der CAN-ID
NMT	0h	0h
SYNC	80h	80h
EMCY	81h	80h + Node-ID
TPDO1	181h	180h + Node-ID
TPDO2	281h	280h + Node-ID
Standard-SDO (CANopen-Master / Client CANopen-Slave / Server)	601h	600h + Node-ID
Standard-SDO (CANopen-Slave / Client CANopen-Master / Client)	581h	580h + Node-ID
Heartbeat	701h	700h + Node-ID

Abb. 4.1: CAN-ID Berechnung nach Predefined Connection Set

## 5 Objektverzeichnis

Hier sind alle existierenden Parameter enthalten, die durch andere Busteilnehmer zugänglich sein müssen, um den Sensor parametrieren zu können. Durch diese Einflussgrößen werden Statusmaschinen, Kommunikationsverhalten und die Applikation selbst beeinflusst. Die Untergliederung CANopen untergliedert ein Objektverzeichnis. Dabei sind die Bereiche 1000h - 1FFFh, 2000h - 5FFFh und 6000h - 9FFFh am relevantesten, da über diese die Kommunikation und die Eingruppierung in ein bestimmtes Geräteprofil realisiert wird. Zusätzlich können herstellerspezifische Besonderheiten darin implementiert werden, welche in keinem der beiden anderen Bereiche zulässig wären.

Index	Verwendung
0000	Nicht genutzt
0001-009F	Datentypen (Sonderfall)
00A0-0FFF	Reserviert
1000-1FFF	Kommunikationsprofil
2000-5FFF	Herstellerspezifischer Bereich
6000-9FFF	Bis zu 8 standardisierte Geräteprofile
A000-AFFF	Prozessabbilder von IEC61131-Geräten
B000-BFFF	Prozessabbilder von CANopen-Gateways nach CiA 302-7
C000-FFFF	Reserviert

Abb. 5.1: Untergliederung Objektverzeichnis

### 5.1 Zugriff auf Parameterdaten per Service-Daten-Objekt (SDO)

Ein CANopen-Gerät wird in der Regel per SDO parametrieren. Die entsprechenden COB-IDs werden im nachfolgenden Kapitel beschrieben, [siehe Kap. 5.4](#). Das Datenfeld dieser CAN-Nachricht ist wie in folgender Darstellung aufgebaut.

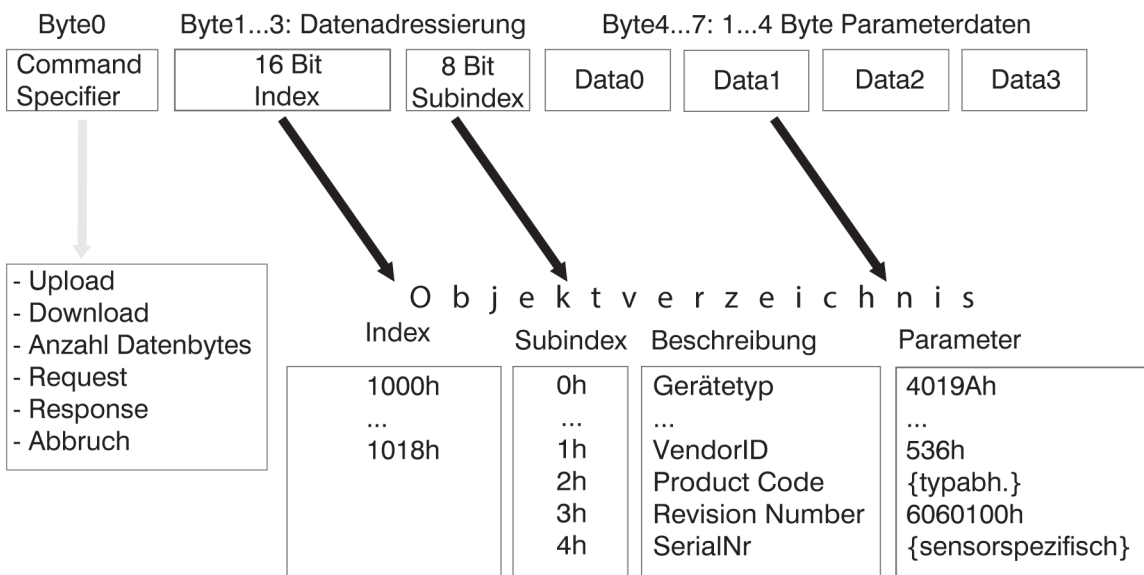


Abb. 5.1: Datenfeld einer CAN-Nachricht

Über die Service-Daten-Objekte ist ein Zugriff auf die in Kapitel Objektverzeichnis, [siehe Kap. 5.5](#), erwähnten Faktoren möglich. Wie der Grafik entnommen werden kann, ist ein entsprechender Parameter über eine Kombination von 16-bit Index und 8-bit Subindex adressierbar. Die Grafik stellt zusätzlich dar, dass insgesamt bis zu 8 Bytes im Datenteil der CAN-Nachricht untergebracht sind, welche den Command Specifier, die Adressierung und die Parameterdaten enthalten.



## 5.2 Kommunikationsparameter nach CIA DS-301

Index	Subindex	Parameter	Datentyp	Attribute	Standardvalue	Ability to store value	Mapping
0x1000	0	Gerätetyp	UNS32	ro	0x00080196 / 524694		
0x1001	0	Fehlerregister	UNS8	ro	0x00 / 0		x
0x1003		Pre-defined error field					
0x1003	0	Number of errors	UNS8	rw	0x00 / 0		
	1	Standard error field	UNS32	ro	0		
	2	Standard error field	UNS32	ro	0		
	3	Standard error field	UNS32	ro	0		
	4	Standard error field	UNS32	ro	0		
	5	Standard error field	UNS32	ro	0		
	6	Standard error field	UNS32	ro	0		
	7	Standard error field	UNS32	ro	0		
	8	Standard error field	UNS32	ro	0		
0x1005	0	COB-ID SYNC message	UNS32	rw	0x00000080 / 128		
0x1007	0	Synchronous window length	UNS32	rw	0x00000000 / 0		
0x1008	0	Manufacturer device name	STRING	const	wireSENSOR CO P		
0x1009	0	Manufacturer hardware version	STRING	const	1.0		
0x100a	0	Manufacturer software version	STRING	const	1.13		
0x1010		Store parameters					
	0	max sub-index	UNS8	ro	0x04 / 4		
	1	save all parameters	UNS32	rw	0x00000001 / 1		
	2	save comunication parameters	UNS32	rw	0x00000001 / 1		
	3	save application parameters	UNS32	rw	0x00000001 / 1		
0x1011		restore application parameters					
	0	max sub-index	UNS8	ro	0x04 / 4		
	1	restore all parameters	UNS32	rw	0x00000001 / 1		
	2	restore communication parameters	UNS32	rw	0x00000001 / 1		
	3	restore application parameters	UNS32	rw	0x00000001 / 0		
0x1014	0	COB-ID EMCY	UNS32	ro	\$NODEID+0x80		
0x1015	0	inhibit time EMCY	UNS16	rw	0x0064 / 100	x	
0x1017	0	Producer heartbeat time	UNS16	rw	0	x	
0x1018		Identity					
	0	max sub-index	UNS8	ro	0x04 / 4		
	1	Vendor-ID	UNS32 ro	0x00000536			
	2	Product code	UNS32	ro			
	3	Revision number	UNS32	ro			
	4	Serial number	UNS32	ro			
0x1029	Error behaviour						
	0	max sub-index	UNS8	ro	0x06 / 6		
	1	Communication	UNS8	rw	0x01 / 1		
	2	Communication other	UNS8	rw	0x01 / 1		
	3	Communication passive	UNS8	rw	0x01 / 1		
	4	Generic	UNS8	rw	0x00 / 0		

Index	Subindex	Parameter	Datentyp	Attribute	Standardvalue	Ability to store value	Mapping
	5	Device profile	UNS8	rw	0x00 / 0		
	6	Manufacturer specific	UNS8	rw	0x00 / 0		
0x1200		SDO server parameter					
	0	max sub-index	UNS8	ro	0x02 / 2		
	1	COB-ID client to server	UNS32	ro	\$NODEID+0x600		
	2	COB-ID server to client	UNS32	ro	\$NODEID+0x580		
0x1800		TPDO communication parameter					
	0	max sub-index	UNS8	ro	0x06 / 6		
	1	COB-ID used by TPDO	UNS32	rw	\$NODEID +0x400001	x	
	2	transmission type	UNS8	rw	0xfe / 254	x	
	3	inhibit time	UNS16	rw	0x01f4 / 500	x	
	4	compatibility entry	UNS8	rw			
	5	event timer	UNS16	rw	0x0000 / 0	x	
	6	SYNC start value	UNS8	rw	0x00 / 0	x	
0x1801		TPDO communication parameter					
	0	max sub-index	UNS8	ro	0x06 / 6		
	1	COB-ID used by TPDO	UNS32	rw	\$NODEID +0x400002	x	
	2	transmission type	UNS8	rw	0x01 / 1	x	
	3	inhibit time	UNS16	rw	0x0000 / 0	x	
	4	compatibility entry	UNS8	rw			
	5	event timer	UNS16	rw	0x0000 / 0	x	
	6	SYNC start value	UNS8	rw	0x00 / 0	x	
0x1A00		TPDO mapping parameter					
	0	Number of mapped objects	UNS8	rw	0x01 / 1		
	1	Mapped object 1	UNS32	rw	0x60040020 / 16108	x	
	2	Mapped object 2	UNS32	rw	0x00000000 / 0	x	
	3	Mapped object 3	UNS32	rw	0x00000000 / 0	x	
	4	Mapped object 4	UNS32	rw	0x00000000 / 0	x	
	5	Mapped object 5	UNS32	rw	0x00000000 / 0	x	
	6	Mapped object 6	UNS32	rw	0x00000000 / 0	x	
	7	Mapped object 7	UNS32	rw	0x00000000 / 0	x	
	8	Mapped object 8	UNS32	rw	0x00000000 / 0	x	
0x1A01		TPDO mapping parameter					
	0	Number of mapped objects	UNS8	rw	0x01 / 1		
	1	Mapped object 1	UNS32	rw	0x60040020 / 16108	x	
	2	Mapped object 2	UNS32	rw	0x00000000 / 0	x	
	3	Mapped object 3	UNS32	rw	0x00000000 / 0	x	
	4	Mapped object 4	UNS32	rw	0x00000000 / 0	x	
	5	Mapped object 5	UNS32	rw	0x00000000 / 0	x	
	6	Mapped object 6	UNS32	rw	0x00000000 / 0	x	
	7	Mapped object 7	UNS32	rw	0x00000000 / 0	x	
	8	Mapped object 8	UNS32	rw	0x00000000 / 0	x	

Index	Subindex	Parameter	Datentyp	Attribute	Standardvalue	Ability to store value	Mapping
0x1F80	0	NMT startup	UNS32	rw	0x00000000 / 0	x	

### 5.3 Herstellerspezifische Kommunikationsparameter

Index	Subindex	Parameter	Datentyp	Attribute	Standardvalue	Ability to store value	Mapping
0x2103		Sensors position					
	0	max sub-index	UNS8	ro	0x02 / 2		
	1	Sensor 1	UNS32	ro	dynamisch		x
	2	Sensor 2	UNS32	ro	dynamisch		x
	3	Sensor 1 RAW	UNS16	ro	dynamisch		x
	4	Sensor 2 RAW	UNS16	ro	dynamisch		x
0x2104		Sensor speed					
	0	max sub-index	UNS8	ro			
	1	Sensor 1	UNS16	ro	dynamisch		x
	2	Sensor 2	UNS16	ro	dynamisch		x
0x2200		Sensors preset value					
	0	max sub-index	UNS8	ro			
	1	Sensor 1	UNS32	rw	specific	x	
	2	Sensor 2	UNS32	rq	specific	x	
0x2201		Sensors direction					
	0	max sub-index	UNS8	ro			
	1	Sensor 1	UNS16	rw			
	2	Sensor 2	UNS16	rw	specific	x	
0x3000	0	Node Number	UNS8	rw	0x01 / 1	x	
0x3001	0	Baudrate	UNS8	rw	0x09 / 9	x	

### 5.4 Kommunikationsparameter nach CIA DS-406

Index	Subindex	Parameter	Datentyp	Attribute	Standardvalue	Ability to store value	Mapping
0x6000	0	Operation parameters	UNS16	rw	specific	x	
0x6002	0	Total measuring range in measuring units	UNS32	ro	specific		
0x6003	0	Preset value	UNS32	rw	specific	x	
0x6004	0	Position value	UNS32	ro	dynamisch		x
0x6005		Linear encoder measuring step settings					
	0	max sub-index	UNS8	ro	0x02 / 2		
	1	Position step setting	UNS32	ro	specific		
	2	Speed step setting	UNS32	ro	specific		
0x6030		Speed value					
	0	max sub-index	UNS8	ro	0x02 / 2		
	1	Speed value channel 1	UNS16	ro	dynamisch		x

Index	Subindex	Parameter	Datentyp	Attribute	Standardvalue	Ability to store value	Mapping
	2	Speed value channel 2	UNS16	ro	dynamisch		x
0x6200	0	Cyclic timer	UNS16	rw			
0x6500	0	Operating Status	UNS16	ro	specific		
0x6501	0	Measuring step	UNS32	ro	specific		

## 5.5 LSS-Dienste

Die CiA DSP 305 CANopen Layer Setting Service and Protocol (LSS) Dienste und Protokolle wurden implementiert, um das Lesen und Ändern der folgenden Parameter über das Netzwerk zu ermöglichen:

- die CANopen-Knoten-ID
- die CAN-Baudrate
- die LSS-Adresse

Ein LSS-Master ist verantwortlich für die Konfiguration dieser Parameter auf einem oder mehreren LSS-Slaves im CANopen-Netzwerk. Der Master verwendet die COB-ID 7E5h und der Slave die 7E4h. Der Zugriff auf die LSS-Dienste ist ausschließlich verfügbar, wenn sich der CANopen-Knoten im Zustand Stopped befindet.

Nachfolgend zwei Beispiele für Kommandosequenzen möglicher Änderungen. Nach den Änderungen empfiehlt es sich, den Slave neu zu starten. Bei der Änderung der Knoten-ID sollte anschließend eine Bootup-Nachricht erscheinen, welche die neue Knotennummer beinhaltet. Im Anschluss an die Änderung der Bitrate ist die Bootup-Nachricht nach dem Neustart erst sichtbar, wenn die Gegenstelle ebenso auf die geänderte Bitrate eingestellt ist.

### 5.5.1 Bitrate ändern

Die Bitrate kann direkt über das Objekt 0x3001 oder wie nachfolgend über LSS eingestellt werden.

CAN-Identifizier	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Kommentar
00h	02h	01h	-	-	-	-	-	-	Wechsel in NMT Zustand <<stopped>>
7E5h	04h	01h	00h	00h	00h	00h	00h	00h	LSS Konfiguration
7E5h	13h	00h	02h	00h	00h	00h	00h	00h	Bitrate 500 kBit wählen
7E5h	17h	00h	00h	00h	00h	00h	00h	00h	Speichern
00h	81h	00h	-	-	-	-	-	-	Reset Node

Abb. 5.2: LSS-Bitrate auf 500 kBit ändern

Index	Bitrate
0	1 MBit/s
1	800 kBit/s
2	500 kBit/s
3	250 kBit/s
4	125kBit/s
5	100 kBit/s
6	50 kBit/s
7	20 kBit/s
8	10 kBit/s

Index	Bitrate
9	Auto detection

Abb. 5.3: LSS Baudratentabelle mit den in CANopen verfügbaren Bitraten

### 5.5.2 Node-ID ändern

Die Bitrate kann direkt über das Objekt 0x3000 oder wie nachfolgend über LSS eingestellt werden.

CAN-Identifizier	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Kommentar
00h	02h	01h	-	-	-	-	-	-	Wechsel in NMT Zustand <<stopped>>, Beispiel NodeID 1
7E5h	40h	36h	05h	00h	00h	00h	00h	00h	Herstellerkennung (aus Index 1018h/01)
7E5h	41h	D1h	11h	28h	00h	00h	00h	00h	Produkt-Code (aus Index 1018h/01), Beispiel 2626001
7E5h	42h	01h	00h	00h	00h	00h	00h	00h	Revisionsnummer (aus Index 1018h/02), Beispiel 1
7E5h	43h	87h	D6h	12h	00h	00h	00h	00h	Seriennummer (aus Index 1018h/04), Beispiel 1234567
7E5h	11h	02h	00h	00h	00h	00h	00h	00h	neue NodeID (0Ch) zuweisen, Beispiel NodeID 2
7E5h	17h	00h	00h	00h	00h	00h	00h	00h	Speichern
00h	81h	01h	-	-	-	-	-	-	Reset Node, Beispiel NodeID 1

Abb. 5.4: LSS-NodeID ändern

### 5.6 Parametrisierung des Sensors

Standardmäßig gibt der Sensor einen Wertebereich von 0 (Seil eingezogen) bis zum maximalen Messbereich (Seil ausgezogen) über das Objekt 0x6004 (Position Value) / 0x2103:1 (Sensor Position - Sensor 1) aus. Die Einheit ist Mikrometer. Die maximale Unterteilung des Wertebereiches (Auflösung) beträgt 4096.

#### 5.6.1 Preset

Durch Schreiben des Objektes 0x6003 (Preset Value) oder 0x2200:1 (Sensors preset value – Sensor 1) wird der aktuelle Messwert auf den geschriebenen Wert gesetzt. Die Einheit ist Mikrometer.

Wird das Objekt 0x6003 (Preset Value) / 0x2200:1 (Sensors preset value – Sensor 1) anschließend gelesen, wird die Differenz zum Default Preset Wert ausgegeben. Der mögliche Wertebereich wurde durch Setzen des Preset um diese Differenz verschoben.

Beispiele:

1. Der Sensorwert steht bei 1'000'000µm, d.h. das Messseil ist 1m ausgezogen. Wird nun 0 als Preset gesetzt, wird an dieser Position auch 0 angezeigt. Am Messbereichsanfang zeigt der Sensor -1'000'000µm bzw. den entsprechenden Überlaufwert (4'293'967'296), am Messbereichsende 4'000'000µm. Liest man das Preset-Objekt, wird die Differenz -1'000'000 angezeigt.
2. Der Sensorwert steht bei 1'000'000µm, d.h. das Messseil ist 1m ausgezogen. Wird nun 2'000'000 als Preset gesetzt, wird an dieser Position auch 2'000'000µm angezeigt. Am Messbereichsanfang zeigt der Sensor 1'000'000µm, am Messbereichsende 6'000'000µm. Liest man das Preset-Objekt, wird die Differenz +1'000'000 angezeigt.

## 5.6.2 Umkehr Zählrichtung Messwerte

Durch Beschreiben des Objektes 0x6000 (Operating parameters) oder 0x2201:1 (Sensors direction – Sensor 1) mit dem Wert 0x0 bzw. 0x08 kann die Richtung der ansteigenden Werte geändert werden. Dabei wird nicht nur die Richtung der Werte geändert, sondern der gesamte Wertebereich gekippt.

Je nach Modell kann der Wert standardmäßig sowohl 0x00 als auch 0x08 sein. In Werkseinstellung steigen die Werte beim Seilauszug an.

Beispiele: Änderung des aktuellen Messwertes eines Sensors mit Messbereich 5m bei Umstellung von steigenden Messwerten auf fallende Messwerte, bei Auszug des Seiles.

Aktuell	Nach Umstellung
0µm	5'000'000µm
5'000'000µm	0µm
2'000'000µm	3'000'000µm
3'000'000µm	2'000'000µm
2'500'000µm	2'500'000µm

Abb. 5.5: Umkehr Zählrichtung Messwerte

## 6 Prozessdatenobjekte: PDO (TxPD01 - TxPD02)

### 6.1 Synchronisiertes Senden

Alle aktivierten TxPDOs können jederzeit (Zustand Operational vorausgesetzt), durch Senden einer SYNC-Nachricht an das Gerät abgefragt werden. Es können mehrere Sensoren gleichzeitig befragt werden. Dafür muss der Subindex 02h im Objekt 1800h einen Wert zwischen 01h und F0h enthalten. Dieser Wert gibt vor, nach welcher Anzahl der eingegangenen SYNC-Nachrichten die damit konfigurierten PDOs gesendet werden.

### 6.2 Zyklisches Senden

Neben dem synchronisierten Senden können auch andere zyklische Übertragungsmodi gewählt werden. Zum einen existiert im System die rein ereignisgesteuerte Übertragung bei Wertänderung. Dazu muss im Subindex 02h der Wert auf FEh gesetzt werden. Des Weiteren existiert die TimerEvent-gesteuerte Übertragung (Subindex 05h, Einheit ms). Diese wird zusätzlich durch applikationsgesteuerte Events beeinflusst. Hierbei wird mittels der Zeit im Subindex 05h, welche in Millisekunden angegeben ist, zyklisch gesendet. Zusätzlich wird bei einer Wertänderung gesendet. Bei Verwendung dieser Kommunikationsart muss der Subindex 02h den Wert FFh enthalten und Subindex 05h einen Wert größer 00h.

Über die Inhibit Time in Subindex 03h kann ein Zeitraum (Einheit 0,1ms), bestimmt werden, in welcher nach einer TPDO Übertragung keine weitere TPDO Übertragung gesendet wird.

## 7 Fehlermeldungen - EMCY-Codes

Mittels sogenannter Emergency-Nachrichten werden wichtige interne Fehler, wie auch CAN-Kommunikationsfehler, an die anderen Teilnehmer im Bus übermittelt. Signalisiert der Status einen aufgetretenen Fehler, so werden zusätzlich die Objekte 1001h (Fehlerregister) und 1003h (vordefiniertes Fehlerfeld) aktualisiert. Werden Fehler bereinigt, wird in der Regel eine Emergency-Nachricht mit dem Code 0000h gesendet.

0x1000U	Generic Error
0x6100U	Internal Software
0x8100U	Communication
0x8110U	CAN Overrun (Objects lost)
0x8120U	CAN in Error Passive Mode
0x8130U	Heartbeat Error
0x8140U	Recovered from bus off
0x8200U	Protocol Error
0x8210U	PDO not processed due to length error
0x8220U	PDO length exceeded
0x8240U	Unexpected SYNC data length







MICRO-EPSILON MESSTECHNIK GmbH & Co. KG  
Königbacher Str. 15 94496 Ortenburg / Deutschland  
+49 (0) 8542 / 168-0  
e-mail: [info@micro-epsilon.de](mailto:info@micro-epsilon.de)  
[www.micro-epsilon.com/contact/worldwide/](http://www.micro-epsilon.com/contact/worldwide/)

X9750489-A012074TSw  
© MICRO-EPSILON MESSTECHNIK