

# **HYDAC** **ELECTRONIC**

**Functional Safety**  
**PL d**  
**SIL 2**

**CAN**open  
*safety easy to use*

## **Linear Wegmessumformer**

### **HLT 1100-R2 CAN**

**CANopen Safety**  
**Modifikation: 000**

**Protokoll-  
beschreibung**  
(Originalanleitung)



## **Inhaltsverzeichnis**

<b>1</b>	<b>Einleitung.....</b>	<b>6</b>
<b>2</b>	<b>Allgemeine Hinweise .....</b>	<b>6</b>
2.1	Geltungsbereich.....	7
2.2	Haftungsausschluss.....	7
2.3	Symbole.....	8
<b>3</b>	<b>Funktionen des HLT CANopen Safety.....</b>	<b>8</b>
<b>4</b>	<b>Übertragungsgeschwindigkeit.....</b>	<b>9</b>
<b>5</b>	<b>CAN-Frames .....</b>	<b>9</b>
<b>6</b>	<b>Node-ID .....</b>	<b>9</b>
<b>7</b>	<b>Übertragungsdienste.....</b>	<b>10</b>
7.1	Service Data Object (SDO).....	10
7.2	Process Data Object (PDO).....	11
7.3	Synchronisation Object (SYNC).....	11
7.4	Safety-relevant Data Object (SRDO) .....	12
7.5	Emergency Object (EMCY).....	13
7.6	Heartbeat.....	13
7.7	Network Management Services (NMT) .....	14
7.8	Boot Up Protocol.....	14
<b>8</b>	<b>Datenfluss im HLT CANopen Safety.....</b>	<b>15</b>
8.1	Sensor Unit.....	15
8.2	Calibration & Scaling .....	15
<b>9</b>	<b>Das Object Dictionary.....</b>	<b>15</b>
9.1	Aufbau des Object Dictionary .....	15
9.2	Struktur des gerätespezifischen Teils nach DS406 .....	16
<b>10</b>	<b>Einträge im Object Dictionary.....</b>	<b>16</b>
10.1	Communication Profile Specific Entries (DS301) .....	16
10.1.1	Index 1000h: DeviceType (read only).....	16
10.1.2	Index 1001h: Error register (read only) .....	16
10.1.3	Index 1002h: Manufacturer status register (read only) .....	17
10.1.4	Index 1003h: Pre-defined error field (read only) .....	17
10.1.5	Index 1005h: COB-ID SYNC (read write) .....	17
10.1.6	Index 1008h: Manufacturer device name (const).....	18
10.1.7	Index 1009h: Manufacturer hardware version (const) .....	18
10.1.8	Index 100Ah: Manufacturer software version (const) .....	18

10.1.9	Index 1010h: Store parameters .....	18
10.1.10	Index 1011h: Restore default parameters.....	19
10.1.11	Index 1014h: COB-ID emergency message (read write) .....	19
10.1.12	Index 1015h: Inhibit time EMCY (read write) .....	19
10.1.13	Index 1017h: Producer heartbeat time (read write) .....	20
10.1.14	Index 1018h: Identity object (record) .....	20
10.1.15	Index 1029h: Error behaviour .....	20
10.1.16	Index 1301h: SRDO communication parameter 1 .....	21
10.1.17	Index 1381h: SRDO mapping parameter 1.....	21
10.1.18	Index 13FEh: Configuration valid (read write).....	22
10.1.19	Index 13FFh: Safety configuration checksum.....	22
10.1.20	Index 1800h & 1801h: TPDO communication parameter 1 & 2 .....	22
10.1.21	Index 1A00h & 1A01h: TPDO mapping parameter 1 & 2.....	23
10.1.22	Index 1F80h: NMT startup (read write).....	24
10.2	<i>Device Profile Specific Entries (DS406) .....</i>	<i>24</i>
10.2.1	Index 6000h: Operating parameter (read write).....	24
10.2.2	Index 6002h: Total measuring range in measuring units (read only) ..	24
10.2.3	Index 6003h: Preset value (read write).....	24
10.2.4	Index 6004h: Position value (read only).....	24
10.2.5	Index 6005h: Linear encoder measuring step settings (read only) .....	24
10.2.6	Index 600Ch: Position raw value (read only) .....	25
10.2.7	Index 6030h: Speed value (read only).....	25
10.2.8	Index 6031h: Speed parameters .....	25
10.2.9	Index 6100h: Safety position configuration parameters .....	26
10.2.10	Index 6101h: Safety speed configuration parameters .....	26
10.2.11	Index 6120h: Safety normal resolution position sequence.....	27
10.2.12	Index 6121h: Safety inverted normal resolution position sequence....	28
10.2.13	Index 6124h: Safety speed sequence.....	28
10.2.14	Index 6125h: Safety inverted speed sequence .....	28
10.2.15	Index 61FEh: Safety application configuration valid (read write) .....	28
10.2.16	Index 61FFh: Safety application configuration signature .....	29
10.2.17	Index 6200h: Cyclic timer (read write) .....	29
10.2.18	Index 6500h: Operating status (read only).....	29
10.2.19	Index 6501h: Measuring step (const).....	29
10.2.20	Index 6503h: Alarms (read only).....	29
10.2.21	Index 6504h: Supported alarms (const).....	29
10.2.22	Index 6505h: Warnings (read only).....	29
10.2.23	Index 6506h: Supported warnings (const) .....	30
10.2.24	Index 6507h: Profile and software version (const).....	30
10.2.25	Index 6508h: Operating time (read only) .....	30
10.2.26	Index 6509h: Offset value (read only).....	30
10.2.27	Index 650Ah: Module identification (read only).....	30
10.2.28	Index 650Bh: Serial number (read only) .....	30
10.2.29	Index 650Eh: Device capability (read only).....	31
10.3	<i>Manufacturer Specific Entries.....</i>	<i>31</i>
10.3.1	Index 2001h: Node-ID.....	31
10.3.2	Index 2002h: Baudrate .....	31

10.3.3	Index 2010h: Checksum (read only).....	32
10.3.4	Weitere Indizes im Bereich 2000h bis 5FFFh (reserved).....	32
<b>11</b>	<b>Layer setting services (LSS) und Protokolle .....</b>	<b>32</b>
11.1	Finite state automaton, FSA.....	33
11.2	Übertragung von LSS-Diensten.....	35
11.2.1	LSS-Nachrichtenformat .....	35
11.3	Switch mode Protokolle.....	35
11.3.1	Switch mode global Protokoll.....	35
11.3.2	Switch mode selective Protokoll .....	35
11.4	Configuration Protokolle .....	36
11.4.1	Configure Node-ID Protokoll.....	36
11.4.2	Configure bit timing parameters Protokoll.....	37
11.4.3	Activate bit timing parameters Protokoll.....	38
11.4.4	Store configuration Protokoll.....	38
11.5	Inquire LSS-Address Protokolle .....	39
11.5.1	Inquire Identity Vendor-ID Protokoll .....	39
11.5.2	Inquire Identity Product-Code Protokoll .....	39
11.5.3	Inquire Identity Revision-Number Protokoll.....	40
11.5.4	Inquire Identity Serial-Number Protokoll .....	40
11.5.5	Inquire Node-ID Protokoll .....	41
11.6	Identification Protokolle .....	41
11.6.1	LSS identify remote slave Protokoll .....	41
11.6.2	LSS identify slave Protokoll .....	42
11.6.3	LSS identify non-configured remote slave Protokoll .....	42
11.6.4	LSS identify non-configured slave Protokoll .....	43
<b>12</b>	<b>Anschluss.....</b>	<b>43</b>
12.1	Einschalten der Versorgungsspannung.....	43
12.2	Einstellen der Node-ID mittels LSS-Dienste .....	43
12.2.1	Konfiguration der Node-ID, Ablauf .....	43
12.2.2	Konfiguration der Baudrate, Ablauf.....	44
<b>13</b>	<b>Inbetriebnahme .....</b>	<b>45</b>
13.1	CAN – Schnittstelle .....	45
13.2	EDS-Datei .....	45

## Vorwort

Diese Dokumentation beschreibt die bestimmungsgemäße Einbindung des Produktes in ein übergeordnetes Steuerungssystem. Sie dient dazu, die bereitgestellte Kommunikationsschnittstelle kennen zu lernen und ihre Einsatzmöglichkeiten optimal zu nutzen.

Die Angaben in dieser Dokumentation entsprechen dem Zeitpunkt der Literaturerstellung. Abweichungen bei technischen Angaben, Abbildungen und Maßen sind deshalb möglich.

Entdecken Sie beim Lesen dieser Dokumentation Fehler oder haben weitere Anregungen und Hinweise, so wenden Sie sich bitte an:

HYDAC ELECTRONIC GMBH  
Technische Dokumentation  
Hauptstraße 27  
66128 Saarbrücken  
-Deutschland-  
Tel: +49(0)6897 / 509-01  
Fax: +49(0)6897 / 509-1726  
Email: [electronic@hydac.com](mailto:electronic@hydac.com)

Die Redaktion freut sich über Ihre Mitarbeit.

**„Aus der Praxis für die Praxis“**

## 1 Einleitung

Die Wegmessumformer der Familie HLT CANopen Safety entsprechen dem CANopen Standard gemäß folgenden Profilen und Standards:

- [1] CiA DS301, Version: 4.2.0 (21 February 2011)  
**CANopen application layer and communication profile**
- [2] EN 50325-5 mit kundenspezifischen Modifikationen  
**Framework for safety-relevant communication**
- [3] CiA DS305, Version 3.0.0 (08 May 2013)  
**Layer setting services (LSS) and protocols**
- [4] CiA DS406, Version 4.1.0 (01 June 2019)  
**Device profile for encoders**
- [5] CiA DS302-2, Version 4.1.0 (02.02.2009)  
**Additional application layer functions – Network management**

Dieses Handbuch beschreibt die vom HLT CANopen Safety unterstützten Funktionen. Dabei werden Grundkenntnisse von CAN, CANopen und CANopen Safety vorausgesetzt. Die genaue Funktionsweise ist in den *oben genannten Dokumenten* beschrieben. Da diese in Englisch abgefasst sind, werden die in diesem Handbuch beschriebenen Dinge, zur besseren Zuordnung, mit dem in der Spezifikation genannten englischen Begriff bezeichnet und *kursiv* dargestellt.



In der Software des HLT 1100 sind zwei CANopen-Knoten realisiert, die sich lediglich hinsichtlich der Konfiguration unterscheiden. Daher wird im vorliegenden Dokument lediglich ein CANopen-Knoten beschrieben..

## 2 Allgemeine Hinweise

Diese Protokollbeschreibung, einschließlich der darin enthaltenen Abbildungen, ist urheberrechtlich geschützt. Drittanwendungen dieses Dokuments, welche von den urheberrechtlichen Bestimmungen abweichen, sind verboten. Die Reproduktion, Übersetzung sowie die elektronische und fotografische Archivierung und Veränderung bedarf der schriftlichen Genehmigung durch den Hersteller. Ein Verstoß kann rechtliche Schritte gegen den Zuwiderhandelnden nach sich ziehen.

Lesen Sie vor der Inbetriebnahme des Produkts die zugehörige Bedienungsanleitung sowie diese Protokollbeschreibung und stellen Sie sicher, dass das hier beschriebene Produkt, nachfolgend auch als Mess-System bezeichnet, für Ihre Anwendung geeignet ist.



Vor jeder Inbetriebnahme, Montage oder einem Austausch ist der Zustand des Mess-Systems sowie des mitgelieferten Zubehörs auf Beschädigung durch Sichtkontrolle zu prüfen.



Falsche Handhabung bzw. die Nichteinhaltung von Gebrauchshinweisen oder technischen Angaben kann zu Sach- und / oder Personenschäden führen.

## 2.1 Geltungsbereich

Diese Protokollbeschreibung gilt ausschließlich für folgende Mess-System-Baureihen zur Erfassung von Linearbewegungen. Die durch diese Beschreibung abgedeckten Produkte sind anhand des folgenden Typenschlüssel-Aufbaus erkennbar:

- CANopen: **HLT 1100-R2-xxx-F13-xxxx-S2PD-000**
  - Nur die mit „x“ gekennzeichneten Stellen des Typenschlüssels können mit den im Datenblatt aufgeführten Eigenschafts-Kennzeichen variable belegt sein.

Die Produkte sind durch aufgeklebte Typenschilder gekennzeichnet und sind Bestandteil einer Anlage bzw. Maschine.

Folgende Dokumentationen gelten somit immer zusammen:

- Anlagen- / maschinenspezifische Betriebsanleitungen des Betreibers
- Die zugehörige Bedienungsanleitung
- Diese Protokollbeschreibung für CANopen Safety

## 2.2 Haftungsausschluss

Diese Protokollbeschreibung haben wir nach bestem Wissen und Gewissen erstellt. Es ist dennoch nicht auszuschließen, dass sich trotz größter Sorgfalt Fehler eingeschlichen haben könnten. Bitte haben Sie deshalb Verständnis dafür, dass wir, soweit sich nachstehend nichts Anderes ergibt, unsere Gewährleistung und Haftung - gleich aus welchen Rechtsgründen - für die Angaben in dieser Bedienungsanleitung ausschließen.







Im Falle der Übersetzung ist der Text der deutschen Protokollbeschreibung der allein gültige.

Insbesondere haften wir nicht für entgangenen Gewinn oder sonstige Vermögensschäden. Dieser Haftungsausschluss gilt nicht bei Vorsatz und grober Fahrlässigkeit. Er gilt ferner nicht für Mängel, die arglistig verschwiegen wurden oder deren Abwesenheit garantiert wurde, sowie bei schuldhafter Verletzung von Leben, Körper und Gesundheit. Sofern wir fahrlässig eine vertragswesentliche Pflicht verletzen, ist unsere Haftung auf den vorhersehbaren Schaden begrenzt. Ansprüche aus Produkthaftung bleiben unberührt.

## 2.3 Symbole

Nachfolgend sind alle in diesem Dokument verwendete Symbole mit ihrer Bedeutung aufgelistet.

	Das Symbol bedeutet, dass der an dieser Stelle beschriebene Sachverhalt verboten ist ( <i>Allgemeines Verbotssymbol gemäß DIN EN ISO 7010</i> ).
	Das Symbol bedeutet, dass Tod, schwere Körperverletzung oder erheblicher Sachschaden eintreten können, wenn die an dieser Stelle beschriebene Maßnahme nicht beachtet oder angewendet wurde ( <i>Allgemeines Warnzeichen gemäß ISO 7010</i> ).
	Das Symbol kennzeichnet wichtige Informationen bzw. Merkmale und Anwendungstipps zum verwendeten Produkt.
	Das Symbol bedeutet, dass entsprechende ESD-Schutzmaßnahmen nach DIN EN 100 015-1 zu beachten sind. (Herbeiführen eines Potentialausgleichs zwischen Körper und Geräte-masse sowie Gehäusemasse über einen hochohmigen Widerstand (ca. 1 MOhm) z.B. mit einem handelsüblichen ESD-Armband).

## 3 Funktionen des HLT CANopen Safety

Die Mess-Systeme der Familie HLT 1100 wird zur Erfassung von Linearbewegungen, sowie der Aufbereitung der Messdaten für eine nachgeschaltete Steuerung bei mobilen Prozess- und Steuerungs-Abläufen verwendet.

- **Erfassung von Linearbewegungen und Bestimmung der Position des Messmagneten auf dem Sensorrohr (absolut)**
- Übertragung der aktuellen Position und/oder der Geschwindigkeit als **PDO oder SRDO** bei folgenden Ereignissen:
  - **Synchron** über empfangene SYNC-Objekte (nur PDO)
  - **Asynchron** nach Ablauf einer einstellbaren Zykluszeit



## 4 Übertragungsgeschwindigkeit

Die Übertragungsgeschwindigkeit von CAN kann in bestimmten Bereichen gewählt werden. Sie wird in bit/s angegeben und auch als **Baudrate** bezeichnet. Die Baudrate eines Gerätes kann über einen Object Dictionary Parameter (siehe *Object Dictionary, Index 2002h*) oder über das Layer setting services (LSS) Protokoll geändert werden (LSS siehe [1]).

Eine Besonderheit von CAN ist, dass die Baudrate die maximale Länge der Verdrahtung stark beeinflusst. Sowohl die maximale Buslänge wie auch die Länge von Stichleitungen sind von der Übertragungsgeschwindigkeit abhängig. Nachfolgende Tabelle beschreibt diese Abhängigkeit:

Bitrate [kbit/s]	Buslänge [m]	Stichlänge [m]	Bitlänge [µs]
1000	25	0,3	1
800	50	0,5	1,25
500	100	0,8	2
250	250	1,5	4
125	500	3	8
50	1000	5	20
20	2500	7	50
10	5000	10	100

Das Timing entspricht der DS301, *Bit rates and timing*.

Die verwendete Übertragungsrate ist in einem nicht-flüchtigen Speicher hinterlegt. Sie ist im Auslieferungszustand auf **250 kbit/s** eingestellt.

## 5 CAN-Frames

Der HLT CANopen Safety unterstützt die in der Spezifikation geforderten 11-bit Standard-Frames mit 11-bit Identifier sowie 29-bit Frames mit 29-bit Identifier.

## 6 Node-ID

Zum Betrieb des HLT CANopen Safety in einem CANopen-Netzwerk ist es notwendig, dass eine innerhalb des Netzes einmalige *Node-ID* eingestellt wird.

Die eingestellte *Node-ID* ist in einem nicht-flüchtigen Speicher hinterlegt und kann über den CAN-Bus (siehe *Object Dictionary Index 2001h*) oder LSS [3] eingestellt werden. Im Auslieferungszustand ist die Adresse **1** eingestellt.

## 7 Übertragungsdienste

### 7.1 Service Data Object (SDO)

Bei CANopen werden alle Daten eines Gerätes (Einstellparameter und Messdaten) in einem *Object Dictionary* unter einem definierten *Index* abgelegt. Verschiedene Einträge des *Object Dictionary* werden mit einem *Subindex* noch weiter untergliedert. Mit den *SDOs* können nun andere Netzteilnehmer das *Object Dictionary* des HLT CANopen Safety auslesen oder beschreiben.

Der HLT CANopen Safety übernimmt dabei die Rolle eines *Servers*, das Gerät, das die Daten auslesen oder beschreiben will, die eines *Clients*.

Zum Transfer von Daten muss der HLT CANopen Safety ein *Receive-SDO* besitzen, mit dem er Daten empfängt und ein *Transmit-SDO*, mit dem er die Daten sendet. Ablauf des Datentransfers:

#### Auslesen des Object Dictionary:

1. Ein Gerät (*Client*) sendet das *Receive-SDO* des HLT CANopen Safety (*Server*). In diesem *SDO* befindet eine Kennung, dass das *Object Dictionary* gelesen werden soll, sowie der gewünschte *Index* und *Subindex*.
2. Der HLT CANopen Safety (*Server*) sendet sein *Transmit-SDO*. In diesem befinden sich ebenfalls der *Index* und der *Subindex*, sowie die gelesenen Daten.

#### Beschreiben des Object Dictionary:

1. Ein Gerät (*Client*) sendet das *Receive-SDO* des HLT CANopen Safety (*Server*). In diesem *SDO* befinden eine Kennung, dass das *Object Dictionary* beschrieben werden soll, sowie der gewünschte *Index*, *Subindex* und die einzuschreibenden Daten.
2. Der HLT CANopen Safety (*Server*) sendet sein *Transmit-SDO*. In diesem befinden sich ebenfalls der *Index* und der *Subindex*, sowie eine Kennung, dass das *Object Dictionary* beschrieben wurde.

Sollte dabei ein Fehler auftreten, z.B. dass der angegebene *Index* nicht existiert, oder dass versucht wurde einen *read only* Eintrag zu beschreiben, oder dass die Daten nicht innerhalb des Gültigkeitsbereiches lagen, so enthält das *Transmit-SDO* eine entsprechende *Abort SDO Transfer* - Kennung und einen entsprechenden *Abort Code* (siehe [1])

Die jeweilige *COB-ID* des *SDO* entspricht dem in der DS301 festgelegten *Pre defined Connection Set* und ist nicht änderbar.

COB-IDs für Service Data Objects

<b>SDO</b>	<b>COB-ID</b>
<i>Receive – SDO</i>	600h+Node-ID
<i>Transmit – SDO</i>	580h+Node-ID

**7.2 Process Data Object (PDO)**

Die Datenübertragung mittels *SDOs* ist zwar sehr flexibel, hat aber für die Übertragung von Messwerten oder Stellgrößen einige Nachteile: Es kann nur ein Datum gelesen werden, die Daten müssen erst mit einem *SDO* angefordert werden und dadurch, dass der jeweilige *Index* und *Subindex* mit übertragen wird, steigt der so genannte Overhead weiter.

Aus diesem Grund definiert CANopen Safety so genannte *Process Data Objects*. Diese enthalten nur die notwendigen Nutzdaten. Es gibt zwei Arten von *PDOs*:

1. *Transmit-PDOs*  
Hiermit kann ein Messgerät seine Messwerte senden.
2. *Receive-PDOs*  
Hiermit können einem Stellglied oder einer Regelungseinheit die Stellgrößen übertragen werden.

Welche Daten sich nun in einem *PDO* befinden, wird durch das so genannte *PDO-Mapping* festgelegt. Dieses *PDO-Mapping* ist im *Object Dictionary* hinterlegt (siehe *Object Dictionary*, *Index 1A00h*).

Mit welcher ID und bei welchem Ereignis ein *PDO* übertragen wird, ist in dem *PDO-Transmission Type* festgelegt. Diese Einstellungen befinden sich ebenfalls im *Object Dictionary* hinterlegt (siehe *Object Dictionary*, *Index 1800h*).

Ereignisse, die zum Senden eines *PDOs* führen:

1. Empfang eines *SYNC* Objektes (Synchroner Transfer).
2. Ablauf einer einstellbaren Zykluszeit im Bereich von 1 Millisekunden bis >1 Minute (Zyklischer Transfer).

Der HLT CANopen Safety implementiert zwei *Transmit-PDOs*, die die aktuellen Sensor- bzw. Prozesswerte beinhalten.

**7.3 Synchronisation Object (SYNC)**

*SYNC* Objekte dienen zur Realisierung eines synchronen Datentransfers. Ein *SYNC* Objekt ist im Prinzip eine CAN Nachricht mit einem definierten Identifier, ohne Daten. CANopen unterscheidet zwischen *SYNC Producer* und *SYNC Consumers*. *SYNC Producer* sind Geräte am Bus, die in einstellbaren Zeitabständen ein *SYNC* senden. *SYNC Consumers* sind Geräte, die auf den Empfang eines *SYNC* reagieren. In einem CANopen Netzwerk können mehrere *SYNC* Objekte existieren. Unterschieden werden die einzelnen *SYNC* Objekte anhand der *SYNC-ID*, welche dem verwendeten CAN Identifier entspricht. Die verwendete *SYNC-ID* ist im *Object Dictionary* hinterlegt.

Der HLT CANopen Safety bietet die Funktionalität eines *SYNC Consumers*. Bei entsprechender Einstellung des *SRDO Transmission Type* wird beim Empfang eines *SYNC* ein *SRDO* gesendet. Die *SYNC-ID* ist auf 80h voreingestellt und kann im *Object Dictionary* geändert werden (siehe *Object Dictionary, Index 1005h*). Unter *SRDO Transmission Type* kann die Anzahl der empfangenen *SYNC* Objekte, die zum Senden eines *SRDO*, führt eingestellt werden.

#### 7.4 Safety-relevant Data Object (SRDO)

Die Datenübertragung mittels so genannter *Safety-relevant Data Objects* sorgt für einen sicheren Transfer von Messwerten. *SRDOs* können zum Senden oder Empfangen von Daten genutzt oder deaktiviert werden.

Ein *SRDO* besteht aus zwei eigenständigen CAN-Frames. Das erste Frame enthält die Daten in normaler Darstellung, das zweite CAN-Frame enthält die Daten in invertierter Darstellung. Dies erlaubt dem Empfänger die Daten zu verifizieren.

Der Versand des *SRDOs* erfolgt zyklisch nach Ablauf einer eingestellten Zeitspanne.

Welche Daten sich nun in einem *SRDO* befinden wird durch das so genannt *SRDO-Mapping* festgelegt. Dieses Mapping ist im *Object Dictionary* hinterlegt.

Die Firmware unterstützt ein *Transmit-SRDO* und keine *Receive-SRDOs*.

Folgende Punkte sind bei der Konfiguration zu beachten:

- Das Ändern der Kommunikationsparameter ist nur im Zustand *Pre-Operational* möglich.
- Um die COB-IDs oder den *Transmission type* zu ändern, ist das zugehörige *SRDO* vorher abzuschalten (*Information Direction* ist 0).
- Die Prüfsumme ist nach Änderung für jedes betroffene *SRDO* neu zu berechnen und im *Object Dictionary* (siehe *Index 13FFh*) abzulegen.
- Nach Änderung der Konfiguration ist diese zurückzulesen und zu verifizieren.
- Nach erfolgter Verifizierung der Daten ist die Konfiguration als gültig zu erklären (siehe *Object Dictionary, Index 13FE, Configuration Valid*).

Folgende Punkte sind bei der Plausibilitätsprüfung der Konfiguration zu beachten:

- Die COB-ID der Nachricht mit den normalen Daten muss stets ungerade sein.
- Die COB-ID der Nachricht mit den invertierten Daten muss stets gerade sein.

Für die Sinnfälligkeit der eingestellten Parameter ist der Konstrukteur der Gesamtanlage verantwortlich (siehe COB-ID Bereich DS304, siehe [2]).

Die Plausibilitätsprüfung erfolgt beim Schreiben des Wertes A5h auf *Configuration Valid*. Das Schreiben ist nur erfolgreich, wenn die Prüfung erfolgreich abgeschlossen wurde. Die Plausibilitätsprüfung erfolgt für alle *SRDOs*, die aktiviert sind (*Information Direction* ungleich 0).

**Achtung:**

**SRDOs werden nur gesendet, wenn die Konfiguration der CANopen Safety Parameter gültig ist (siehe [4]).**

## 7.5 Emergency Object (EMCY)

*EMCY* Objekte werden beim Auftreten eines Fehlers gesendet. *EMCY* Objekte enthalten einen *Emergency Error Code*, den Inhalt des *Error register* sowie ein *Manufacturer specific Error Field*. Ist ein gemeldeter Fehler beseitigt oder verschwunden, so wird dies ebenfalls durch ein spezielles *EMCY* Objekt gemeldet.

Eine Emergency-Nachricht wird gesendet, wenn ein Fehler auftritt oder dieser Fehler wieder verschwindet. (siehe *Index 1002h: Manufacturer status register (read only)*)

Das *EMCY* Objekt hat die voreingestellte ID  $80h + \text{Node-ID}$  und kann im *Object Dictionary* geändert werden (siehe *Object Dictionary, Index 1014h*).

## 7.6 Heartbeat

Mit dem *Heartbeat Protocol* kann eine Überwachung der einzelnen Teilnehmer durchgeführt werden. CANopen unterscheidet zwischen folgenden Funktionen:

1. *Heartbeat Producer*  
sendet in zyklischen Abständen ein *Heartbeat* Objekt.
2. *Heartbeat Consumer*  
überwacht das Senden von bestimmten *Heartbeat* Objekten.

Die Zykluszeit ist im *Object Dictionary* in Millisekunden einstellbar. Eine Zeitangabe von 0 bedeutet „*Heartbeat* nicht aktiv“.

Mit dem *Heartbeat* Objekt wird immer der Status des *Heartbeat Producers* als Byte mit übertragen.

### Bedeutung des Heartbeat- Objekthinhaltes

Wert	Status	Anmerkung
0	<i>BOOTUP</i>	Das Gerät hat gebootet.
4	<i>STOPPED</i>	Das Gerät ist gestoppt.
5	<i>OPERATIONAL</i>	Das Gerät arbeitet normal.
127	<i>PRE-OPERATIONAL</i>	Das Gerät sendet keine <i>PDOs</i> , kann aber <i>SDOs</i> bearbeiten.

Der HLT CANopen Safety kann als *Heartbeat Producer* arbeiten. Die ID des *Heartbeat* ist  $700h + \text{Node-ID}$ . Die Zeit ist mit 0 (nicht aktiv) voreingestellt und kann geändert werden (siehe *Object Dictionary, Index 1017h*).

## 7.7 Network Management Services (NMT)

*NMT* Objekte dienen dazu Geräte zu starten, zu stoppen oder zurückzusetzen. CANopen unterscheidet zwischen folgenden Funktionalitäten:

1. *NMT Master*  
steuert andere Knoten.
2. *NMT Slave*  
wird von einem *Master* gesteuert.

In einem CANopen Netzwerk existiert nur ein *NMT* Objekt mit dem Identifier 0. Es werden immer 2 Bytes übertragen. Das erste Byte enthält den *Command Specifier*, der den Befehl repräsentiert, das zweite Byte enthält die *Node-ID* des Knotens, der diesen Befehl ausführen soll. Ein Wert von 0 bedeutet, dass dieser Befehl für alle Knoten gilt. Folgende Befehle sind möglich:

### NMT Befehle

1. **Start Remote Node**  
Der Knoten wechselt in den Zustand *Operational*.
2. **Stop Remote Node**  
Der Knoten wechselt in den Zustand *Stopped*.
3. **Enter Pre-Operational**  
Der Knoten wechselt in den Zustand *Pre-Operational*.
4. **Reset Node**  
Der "Device Profile Specific"-OD-Bereich wird zurückgesetzt, die Baudrate wird gegebenenfalls neu initialisiert und wechselt dann in den Zustand Reset Communication.
5. **Reset Communication**  
Die Kommunikationseinheit des Knotens wird zurückgesetzt und danach wechselt der Knoten in den Zustand *Pre-Operational*.

Der HLT CANopen Safety arbeitet als *NMT Slave* und unterstützt alle *NMT* Dienste.

## 7.8 Boot Up Protocol

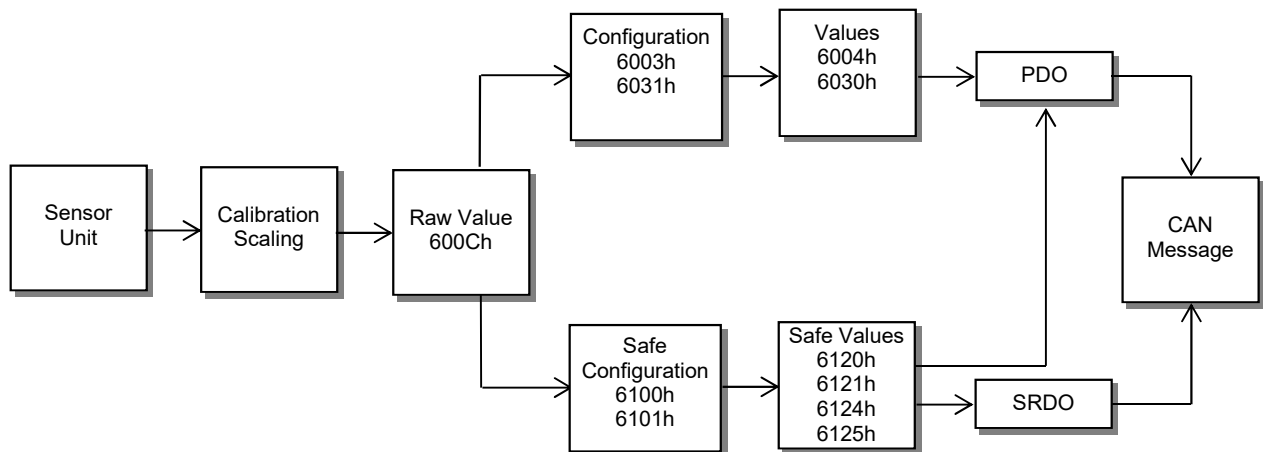
Nach der Initialisierung sendet ein NMT-Slave ein *Boot Up* Objekt. Dies ist im Prinzip nichts anders als ein *Heartbeat* Objekt mit dem Status 0.

Beim HLT CANopen Safety ist diese Funktion realisiert.

## 8 Datenfluss im HLT CANopen Safety

Nachfolgende Bilder zeigen den Datenfluss innerhalb des HLT CANopen Safety, sowie die jeweiligen Indizes des *Object Dictionary*.

### Datenfluss im HLT CANopen Safe



### 8.1 Sensor Unit

Die Sensoreinheit erfasst die Position des Messmagneten auf dem Sensorrohr. Der Typ der Sensoreinheit (*Absolute Linear Encoder*) ist in *Device Type* hinterlegt.

### 8.2 Calibration & Scaling

Die Sensorwerte werden in Calibration & Scaling aufbereitet und berechnet.

## 9 Das Object Dictionary

### 9.1 Aufbau des Object Dictionary

Im *Object Dictionary* sind alle Daten hinterlegt. In den folgenden Kapiteln sind die vom HLT 1100 CANopen Safety unterstützten Einträge aufgeführt. Die Angabe des Index erfolgt spezifikationsgemäß immer in hexadezimaler Notation, ohne dass die hexadezimale Darstellung extra angezeigt wird. Bei jedem Eintrag ist auch die Zugriffsart angegeben. CANopen unterscheidet dabei folgende Zugriffsarten:

#### Zugriffsarten (Access Type) für das Object Dictionary

1. **const**  
Kann nur gelesen werden und liefert immer den gleichen Wert.
2. **read only**  
Kann nur gelesen werden, der Wert kann sich aber während des Betriebes ändern.
3. **write only**  
Der Eintrag kann nur geschrieben werden.

#### 4. read write

Der Eintrag kann geschrieben und gelesen werden.

CANopen unterscheidet innerhalb des Data Dictionary folgende Bereiche:

##### Bereiche des Object Dictionary:

1. Index 0000h .. 1FFFh: Communication profile specific entries  
Einstellungen, die für alle CANopen-Geräte gelten. Diese Einträge sind in der DS301 und DS302 festgelegt.
2. Index 6000h .. 9FFFh: Device profile specific entries  
Gerätespezifische Daten, die in einer Draft Standard festgelegt sind.  
Der HLT CANopen hat das Geräteprofil DS406 realisiert.
3. Index 2000h .. 5FFFh: Manufacturer specific entries  
Herstellerspezifische zusätzliche Daten, die in keiner Spezifikation festgelegt sind.

## 9.2 Struktur des gerätespezifischen Teils nach DS406

Der HLT CANopen Safety implementiert die DS406. Diese beschreibt das Verhalten und die Funktionalität von Winkelgebern oder linearen Encodern (Encodern [4]).

## 10 Einträge im Object Dictionary

Im Folgenden sind die vom HLT CANopen Safety realisierten Funktionalitäten aufgezeigt. Eine detaillierte Beschreibung der Einträge kann in [1] und [4] nachgelesen werden.

### 10.1 Communication Profile Specific Entries (DS301)

#### 10.1.1 Index 1000h: DeviceType (read only)

Enthält die Nummer des verwendeten Geräteprofils (3F080196h):

**406 (0196h)** - *Device profile number*

Sowie die profilspezifische Erweiterung:

**8 (0008h)** – *Absolute linear encoder*

Die Geräteart ist produktspezifisch und für das jeweilige Projekt spezifiziert.

#### 10.1.2 Index 1001h: Error register (read only)

Enthält den aktuellen Fehlerzustand (siehe *EMCY*, sowie [1]).

Der Eintrag kann in TPDOs gemappt werden.

Bit 0	Generic error
Bit 1-3	reserviert
Bit 4	Communications error
Bit 5-6	reserviert
Bit 7	Hersteller spezifisch

Sobald ein Kommunikationsfehler oder herstellerspezifischer Fehler auftritt, ist der generische Fehler gesetzt.



### 10.1.3 Index 1002h: Manufacturer status register (read only)

#### Gerätestatus

Der Eintrag kann in TPDOs gemappt werden.

Bit 0 0001 Fehler beim Laden der Hardwareeinstellungen

Bit 1 0002 Fehler beim Laden der Produktionseinstellungen

Bit 2 0004 Fehler beim Laden der Werkseinstellungen

Bit 3 0008 Fehler beim Laden der Anwendereinstellungen

(Dieser Fehler kann nur durch Speichern / Wiederherstellen und anschließendem Neustart des Gerätes (Power off / Power on) behoben werden.)

Bit 4-5 reserviert

Bit 6 0040 Fehler bei der Positionswerterfassung

Bit 7-8 reserviert

Bit 9 0200 Überlauf der Empfangswarteschlange des CAN-Controllers

(Dieser Fehler kann nur durch Neustart des Gerätes (Power off / Power on) behoben werden.)

Bit 10-15 reserviert

### 10.1.4 Index 1003h: Pre-defined error field (read only)

Hier werden die Fehler bereitgestellt, die im CANopen Gerät aufgetreten sind, und mittels *emergency object* signalisiert wurden (siehe CiA 301)

Beim Versenden eines *EMCY* fügt das Gerät den zugehörigen Fehler der Fehlerhistorie hinzu. Die maximale Anzahl Fehler sind bei Mess-Systemen der Baureihe HLT 1100 auf maximal 10 Einträge definiert.

Der Inhalt eines Eintrags besteht beim HLT 1100 aus dem „*emergency error code*“ (EMCY-EC) (16 Bit) und der Fehlernummer gemäß Eintrag in Index 1002:

1003.x UNSIGNED32

Bit:	31	-	16		15	-	0
	<b>Fehlernummer</b>				<i>EMCY-EC</i>		
	(UNSIGNED16)				(UNSIGNED16)		

Bsp.:                      0008 8140h                      → Gerät hat in den angegebenen Fehlerzustand gewechselt

### 10.1.5 Index 1005h: COB-ID SYNC (read write)

Hier kann die *COB-ID* für das *SYNC*-Objekt eingestellt werden, d.h. die Nachrichten ID für die Identifikation der Synchron-Nachricht bei synchroner Prozessdatenübertragung. Die Priorität dieser Nachricht sollte hoch gewählt werden, damit die Latenz durch andere Nachrichten gering bleibt.

**Standardeinstellung:** 80h (128d)

### 10.1.6 Index 1008h: Manufacturer device name (const)

Liefert den Gerätenamen als Zeichenkette, i.d.R. der Typenschlüssel, z.B. HLT1100-R2-L08-F13-0500-S2PD-000.

Zum Lesen dieses Objektes ist ein „segmented“ Zugriff erforderlich.

### 10.1.7 Index 1009h: Manufacturer hardware version (const)

Liefert die Hardwareversion als Zeichenkette (z.B. „D“).

### 10.1.8 Index 100Ah: Manufacturer software version (const)

Liefert die Gerätesoftware mit Versionsnummer als Zeichenkette (z.B. „Hltco4 V01.02“). Zum Lesen dieses Objektes ist ein „segmented“ Zugriff erforderlich.

### 10.1.9 Index 1010h: Store parameters

Durch Einschreiben der Zeichenkette „save“ (65766173h) werden die aktuellen Einstellungen in den nicht flüchtigen Speicher übertragen.

Der HLT CANopen Safety speichert Einstellungen nicht automatisch, wenn sie geändert werden, sondern nur auf Anforderung.



**Achtung:**

Geänderte Einstellungen müssen mit *StoreParameters* explizit gesichert werden, sonst gehen sie beim Abschalten des Gerätes oder bei den *NMT-Befehlen Reset Node* und *Reset Communication* verloren.

CANopen bietet die Möglichkeit mit Hilfe verschiedener *Subindexes* verschiedene Parameterbereiche zu sichern.

Unterstützt wird hier Subindex 1, 2, 3 und 4.

Nähere Informationen sind [1] zu entnehmen.



**Achtung:**

Die Einstellungen der Node-ID und Baudrate bleiben bei *StoreAllParameters* erhalten.

**Verwendete Subindexes:**

**0: Highest sub-index supported (const)**

**1: Save all parameters (read write)**

**2: Save communication parameters (read write)** (Index von 1000h to 1FFFh)

**3: Save application parameters (read write)** (Index von 6000h to 9FFFh)

**4: Save LSS parameters (read write)** (Index von 2000h to 20FFFh)

Subindex 4 ist Manufacturer specific, folglich nicht im Standard [1] beschrieben

### 10.1.10 Index 1011h: Restore default parameters

Durch Einschreiben der Zeichenkette „load“ (64616F6Ch) werden die werksseitigen Voreinstellungen in den nicht flüchtigen Speicher übertragen.

Der HLT CANopen Safety arbeitet allerdings bis zum Abschalten oder bis zur Ausführung der Befehle *Reset Node* und *Reset Communication* noch mit den aktuellen Einstellungen weiter.

CANopen bietet die Möglichkeit mit Hilfe verschiedener *Subindexes* verschiedene Parameterbereiche zu restaurieren.

Unterstützt wird hier Subindex 1, 2, 3 und 4.

Nähere Informationen sind [1] zu entnehmen.



**Achtung:**

**Die Einstellungen der Node-ID und Baudrate bleiben bei *RestoreAllParameters* erhalten.**

**Verwendete Subindexes:**

**0: Highest sub-index supported (const)**

**1: Restore all parameters (read write)**

**2: Restore communication parameters (read write)** (Index von 1000h to 1FFFh)

**3: Restore application parameters (read write)** (Index von 6000h to 9FFFh)

**4: Restore LSS parameters (read write)** (Index von 2000h to 20FFFh)

Subindex 4 ist Manufacturer specific, folglich nicht im Standard [1] beschrieben

### 10.1.11 Index 1014h: COB-ID emergency message (read write)

Hier kann die *COB-ID* für das *EMCY*-Objekt eingestellt werden (siehe *EMCY*).

Wird die *COB-ID* über ein *SDO-Kommando* auf eine explizite CAN-ID gesetzt, so ist der Mechanismus für die automatische Erweiterung der *COB-ID* um aktive Node-ID deaktiviert. Die vorgegebene CAN-ID wird dann unabhängig von der Node-ID immer für die Übertragung einer *EMCY* genutzt. Wird die *COB-ID* = 0 gesetzt, so wird die Standardeinstellung wieder wirksam.

**Standardeinstellung:** \$NODEID+80h.

### 10.1.12 Index 1015h: Inhibit time EMCY (read write)

Hier kann die *Inhibit Time* für das *EMCY*-Objekt, d.h. Verzögerung zwischen zwei *EMCY* Nachrichten eingestellt werden, z.B. „100 µs“; (siehe *Emergency Object (EMCY)* ).

**0** Es wird keine Inhibit Time gesendet.

**>0** Zeitintervall in [100 µs] für Inhibit time.

### 10.1.13 Index 1017h: Producer heartbeat time (read write)

Heartbeat „Producing“ aktivieren/deaktivieren.

Das Gerät kann Heartbeat-Nachrichten zyklisch versenden, (siehe *Heartbeat*).

- 0** Es werden keine Heartbeat-Nachrichten gesendet.
- >0** Zeitintervall in [ms] für zyklische Heartbeat-Nachrichten.

### 10.1.14 Index 1018h: Identity object (record)

Das Identity Objekt identifiziert den HLT CANopen Safety. Die Identifikation besteht aus vier 32bit-Zahlen. Die Kombination dieser 4 Zahlen ergibt eine weltweit eindeutige Identifikation eines Gerätes.

#### Verwendete Subindizes:

- 0: Highest sub-index supported (const)**
- 1: Vendor-ID (read only)**  
Eindeutiger Herstellercode (DAh für HYDAC ELECTRONIC GmbH)
- 2: Product code (read only)**  
Produktcode der Hydac Electronic (z.B: 929265)
- 3: Revision number (read only)**  
Revisionsnummer des Gerätes.
- 4: Serial number (read only)**  
Seriennummer des Gerätes

### 10.1.15 Index 1029h: Error behaviour

Das Objekt beschreibt das Fehlerverhalten

#### Verwendete Subindizes:

- 0: Highest sub-index supported (const)**
- 1: Communication error (read write)**  
Geräteverhalten bei Auftreten eines Kommunikationsfehlers
  - 0 : Change to NMT state Pre-operational
  - 1 : No change of the NMT state
  - 2 : Change to NMT state Stopped
- 0: Device error (read write)**  
Fehlerverhalten im Falle eines internen Gerätefehlers
  - 0 : Change to NMT state Pre-operational
  - 1 : No change of the NMT state
  - 2 : Change to NMT state Stopped

### 10.1.16 Index 1301h: SRDO communication parameter 1

Enthält die Kommunikationseinstellungen des ersten sicherheitsrelevanten Datenobjektes:

#### Verwendete Subindizes:

- 0: Highest sub-index supported (read only)
- 1: Information direction (read write)
- 2: Refresh-time or SCT (read write)
- 3: SRVT (read write)
- 4: Transmission type (read only)
- 5: COB-ID 1 (read write)
- 6: COB-ID 2 (read write)

### 10.1.17 Index 1381h: SRDO mapping parameter 1

Enthält die Mappingeinstellungen des ersten sicherheitsrelevanten Datenobjektes. Dieser Bereich legt die fest, welche konkreten Signalparameter-Objekte in einem der verfügbaren SRDO übertragen werden.

#### Verwendete Subindizes:

- 0: Highest sub-index supported (read write)
- 1: SR application data object 1 (plain data) (read write)
- 2: SR application data object 1 (bitwise inverted data) (read write)
- 3: SR application data object 2 (plain data) (read write)
- 4: SR application data object 2 (bitwise inverted data) (read write)
- 5: SR application data object 3 (plain data) (read write)
- 6: SR application data object 3 (bitwise inverted data) (read write)
- 7: SR application data object 4 (plain data) (read write)
- 8: SR application data object 4 (bitwise inverted data) (read write)
- 9: SR application data object 5 (plain data) (read write)
- 10: SR application data object 5 (bitwise inverted data) (read write)
- 11: SR application data object 6 (plain data) (read write)
- 12: SR application data object 6 (bitwise inverted data) (read write)
- 13: SR application data object 7 (plain data) (read write)
- 14: SR application data object 7 (bitwise inverted data) (read write)
- 15: SR application data object 8 (plain data) (read write)
- 16: SR application data object 8 (bitwise inverted data) (read write)

Im **Auslieferungszustand** hat der HLT CANopen Safety folgende Einträge:

Index 1381h		
Subindex	Inhalt	Bedeutung
0	12	12 Werte werden im <i>SRDO</i> übertragen.
1	61200108h	Der erste Wert im <i>SRDO</i> ist der Wert von <i>Index 6120h</i> , <i>Subindex 1</i> mit einer Breite von 08h (=8 bit)

2	61210108h	Der zweite Wert im SRDO ist der Wert von <i>Index 6121h, Subindex 1</i> mit einer Breite von 08h (=8 bit)
3	61200208h	Der erste Wert im SRDO ist der Wert von <i>Index 6120h, Subindex 2</i> mit einer Breite von 08h (=8 bit)
4	61210208h	Der zweite Wert im SRDO ist der Wert von <i>Index 6121h, Subindex 2</i> mit einer Breite von 08h (=8 bit)
5	61200308h	Der erste Wert im SRDO ist der Wert von <i>Index 6120h, Subindex 3</i> mit einer Breite von 08h (=8 bit)
6	61210308h	Der zweite Wert im SRDO ist der Wert von <i>Index 6121h, Subindex 3</i> mit einer Breite von 08h (=8 bit)
7	61200708h	Der erste Wert im SRDO ist der Wert von <i>Index 6120h, Subindex 4</i> mit einer Breite von 08h (=8 bit)
8	61210408h	Der zweite Wert im SRDO ist der Wert von <i>Index 6121h, Subindex 4</i> mit einer Breite von 08h (=8 bit)
9	61240108h	Der erste Wert im SRDO ist der Wert von <i>Index 6124h, Subindex 1</i> mit einer Breite von 08h (=8 bit)
10	61250108h	Der zweite Wert im SRDO ist der Wert von <i>Index 6125h, Subindex 1</i> mit einer Breite von 08h (=8 bit)
11	61240208h	Der erste Wert im SRDO ist der Wert von <i>Index 6124h, Subindex 2</i> mit einer Breite von 08h (=8 bit)
12	61250208h	Der zweite Wert im SRDO ist der Wert von <i>Index 6125h, Subindex 2</i> mit einer Breite von 08h (=8 bit)

#### 10.1.18 Index 13FEh: Configuration valid (read write)

Dieser Eintrag dient zur Validierung der SRDO Konfiguration. Durch Schreiben des Wertes A5h wird die Plausibilitätsprüfung angestoßen. Verläuft die Prüfung erfolgreich, wird der Wert akzeptiert. Andernfalls wird der Wert auf 0 gesetzt.

#### 10.1.19 Index 13FFh: Safety configuration checksum

Enthält die Prüfsummen der einzelnen SRDO Konfigurationen.

##### Verwendete Subindizes:

- 0: Highest sub-index supported (read only)
- 1: SRDO1 signature (read write)

#### 10.1.20 Index 1800h & 1801h: TPDO communication parameter 1 & 2

Dieser Bereich legt fest wie ein TPDO, also vom Gerät gesendete Prozessdaten, übertragen werden. Er beschreibt die Struktur zur Definition seiner Übertragungsart

##### Verwendete Subindizes:

- 0: Highest sub-index supported (const)
- 1: COB-ID (read write)
  - COB-ID zur Berechnung der CAN-ID im Betrieb unter welcher das TPDO gesendet wird.

**2: Transmission type (read write)**

Dieser Parameter legt die Übertragungsart fest.

- 0       Azyklisch synchron  
Interne Signalverarbeitung nicht synchron zu SYNC; Übertragung der Nachricht synchron zu SYNC.
- 1       Interne Signalverarbeitung synchron zu SYNC; Übertragung der Nachricht synchron mit jedem SYNC.
- 2       Übertragung der Nachricht synchron mit jedem zweiten SYNC.
- n – 240 Übertragung der Nachricht synchron mit jedem n .ten SYNC.
- 254     (FEh) ereignisgesteuert herstellerspezifische Ereignismöglichkeiten.
- 255     (FFh) ereignisgesteuert geräteprofilspezifische Ereignismöglichkeiten.

**3: Inhibit time (read write)**

Bei aktivem *Transmission type* 254 oder 255, definiert dieser Parameter Mindestwartezeit bevor ein TPDO nach Auftreten eines Ereignisses gesendet wird. Somit kann bei einem häufig auftretenden Ereignis die Anzahl der gesendeten TPDO reduziert werden.

- 0   Der Wert 0 deaktiviert die Mindestwartezeit.
- >0 Die Zeit wird als Vielfaches von 100 µs definiert.

**5: Event timer (read write)**

Bei aktivem *Transmission type* 254 oder 255, definiert dieser Parameter das zeitliche Intervall für das Auslösen eines „Timer-Ereignisses“ welches zum Senden des TPDO führt.

Verfügt das Gerät über gerätespezifische Ereignisse, so wird das TPDO beim Ausbleiben anderer Ereignisse spätestens nach Ablauf dieser Zeitperiode gesendet

- 0   Senden des TPDO ist deaktiviert.
- >0 Das Ereignisintervall als Vielfaches von 1 ms.

**10.1.21 Index 1A00h & 1A01h: TPDO mapping parameter 1 & 2**

Dieser Bereich legt die fest, welche konkreten Signalparameter-Objekte in einem der verfügbaren TPDO übertragen werden. Er beschreibt die Struktur zur Definition der über dieses PDO übertragenen Signalparameter-Objekte.

**Verwendete Subindizes:****0: Number of mapped objects in PDO (read write)****1: 1<sup>st</sup> object to be mapped (read write)**

Erstes Referenz-Objekt zur Festlegung des Signalparameter-Objektes welches über das TPDO übertragen wird

**2-8: 2<sup>nd</sup> – 8<sup>th</sup> object to be mapped (read write)**

Zweites bis achttes Referenz-Objekt zur Festlegung des Signalparameter-Objektes welches über das TPDO übertragen wird

Im **Auslieferungszustand** hat der HLT CANopen Safety folgende Einträge:

<b>Index 1A00h &amp; 1A01h</b>		
<b>Subindex</b>	<b>Inhalt</b>	<b>Bedeutung</b>
0	2	Zwei Werte werden im <i>PDO</i> übertragen.
1	60040020h	Der erste Wert im <i>PDO</i> ist der Wert von <i>Index 6004h</i> , <i>Subindex 0</i> mit einer Breite von 20h (=32 bit)
2	60300110h	Der zweite Wert im <i>SRDO</i> ist der Wert von <i>Index 6030h</i> , <i>Subindex 1</i> mit einer Breite von 10h (=16 bit)

### 10.1.22 Index 1F80h: NMT startup (read write)

Definiert das Startverhalten des Gerätes (siehe *Network Management*)

Bit 3 (Start node) muss immer gesetzt sein.

Ist Bit 2 (NMT master start) gesetzt, wechselt das Gerät nach dem Booten automatisch vom Zustand Pre operational in den Zustand Operational.

Alle anderen Bits müssen 0 sein.

Damit sind nur die Werte 0x08 und 0x0C erlaubt, wobei 0x0c für den automatischen Wechsel in Operational steht. Das Schreiben anderer Werte wird mit dem Abort-Code OutOfRange beantwortet.

## 10.2 Device Profile Specific Entries (DS406)

### 10.2.1 Index 6000h: Operating parameter (read write)

Dieser Eintrag enthält die Parameter für die Ausgabe des *Position Value* ([4]).

Es wird nur die Messrichtung der Betriebsparameter unterstützt.

Bit 3 Messrichtung Lineargeber: Wert: 0 = steigend

Wert: 1 = fallend

### 10.2.2 Index 6002h: Total measuring range in measuring units (read only)

Dieser Eintrag gibt an, wie groß der Messbereich in Messeinheiten ist ([4]).

### 10.2.3 Index 6003h: Preset value (read write)

Setzeingang für den Anfangspunkt des Permanentmagneten, entspricht einer Nullpunktverschiebung.

### 10.2.4 Index 6004h: Position value (read only)

Dieser Eintrag enthält den gemessenen Positionswert (*Position Value*, [4]).

Der Eintrag kann in TPDOs gemappt werden.

### 10.2.5 Index 6005h: Linear encoder measuring step settings (read only)

Dieser Eintrag enthält die Schrittweite der Messwerte von Position und Geschwindigkeit [4]).

#### Verwendete Subindizes:

**0: Highest sub-index supported (const)**

**1: Position step setting (read only)**

Schrittweite des Positionswertes.



Die Positionsschrittweite ist abhängig von den Werkseinstellungen des Gerätes, sie beträgt ein Vielfaches von 0,001 µm.

Bsp. 100000 = 0,1 mm

## **2: Speed step setting (read only)**

Schrittweite des Geschwindigkeitswertes.

Ob der Eintrag existiert, ist abhängig von den Werkseinstellungen.

Die Geschwindigkeitsschrittweite ist abhängig von den Werkseinstellungen, sie beträgt ein Vielfaches von 0,01 mm/s.

Bsp. 100 = 1 mm/s

### **10.2.6 Index 600Ch: Position raw value (read only)**

Dieser Eintrag enthält den Positionsrohwert (*Position Raw Value*, [4]).

Der Eintrag kann in TPDOs gemappt werden.

### **10.2.7 Index 6030h: Speed value (read only)**

Dieser Eintrag enthält den Geschwindigkeitswert [4])

Ob der Eintrag existiert, ist abhängig von den Werkseinstellungen.

Der Geschwindigkeitswert wird als Vielfaches der verwendeten Schrittweite angegeben.

Der Eintrag kann in TPDOs gemappt werden.

#### **Verwendete Subindizes:**

**0: Highest sub-index supported (const)**

**1: Speed value channel 1**

### **10.2.8 Index 6031h: Speed parameters**

Dieser Eintrag enthält die Geschwindigkeitseinstellungen, [4])

Ob der Eintrag existiert, ist abhängig von den Werkseinstellungen. Der

Geschwindigkeitswert wird als Vielfaches der verwendeten Schrittweite angegeben.

#### **Verwendete Subindizes:**

**0: HighestSubindexSupported (const)**

**1: Speed source selector (read only)**

**2: Speed integration time (read write)**

**3: Multiplier value (read only)**

**4: Divider value (read only)**

#### **Zu Subindex 0x01h: Speed source selector**

Auswahl der Geschwindigkeits Quelle

Der Wert beträgt immer 2: Bezugswert ist der Positionswert auf Index 600Ch)

#### **Zu Subindex 0x02h: Speed integration time**

Zeitfenster, über das die Geschwindigkeit gerechnet wird

Anwender spezifische Einstellung: Min. = 20 ms, Max. = 1000 ms

**Zu Subindex 0x03h: Multiplier value**

Multiplikationswert

Der Multiplikationswert berechnet sich aus:

$1000 \times \text{Auflösung des Positionswertes} / \text{Auflösung des Geschwindigkeitswertes}$  (in den Werkseinstellungen hinterlegt)

**Zu Subindex 0x04h: Divider value**

Divisionswert

Der Divisionswert beträgt immer 1000.

**10.2.9 Index 6100h: Safety position configuration parameters**

Dieser Eintrag enthält die sicheren Positionseinstellungen (*Safety position configuration parameters*, [4])

**Verwendete Subindizes:**

**0: Highest sub-index supported (const)**

**1: Safety code sequence (read write)**

**2: Safety preset value for encoder with normal resolution (read write)**

**Zu Subindex 0x01 Safety code sequence**

Bit 3 Messrichtung Lineargeber:      Wert: 0 = steigend  
   Wert: 1 = fallend

**Zu Subindex 0x02 Safety preset value for encoder with normal resolution**

Setzeingang für den Anfangspunkt des Permanentmagneten, entspricht einer Nullpunktverschiebung.

Der Presetwert wird als Vielfaches der verwendeten Schrittweite angegeben.

Beim Empfangen wird der Offsetwert (Siehe Offsetwert senden ) so modifiziert, dass der aktuelle Positionswert (Siehe Positionswert zum Lesen bereitstellen) dem empfangenen Presetwert entspricht.

Wird die Messrichtung geändert, (siehe Betriebsparameter) so wird der Presetwert gelöscht.

**10.2.10 Index 6101h: Safety speed configuration parameters**

Dieser Eintrag enthält die sicheren Geschwindigkeitseinstellungen (*Safety speed configuration parameters*, [4])

Über diesen Eintrag kann die Berechnung der Geschwindigkeit angepasst werden. Ob der Eintrag existiert, ist abhängig von den Werkseinstellungen.

**Verwendete Subindizes:**

- 0: HighestSubindexSupported (const)**
- 1: Safety code sequence (read write)**
- 2: Safety preset value for encoder with normal resolution (read write)**
- 4: Safety speed source selector (read only)**
- 5: Safety speed integration time (read write)**
- 6: Multiplier value (read only)**
- 7: Divider value (read only)**

**Zu Subindex 0x01 Safety code sequence / Safety measuring direction**

(identisch mit den Daten zu Index 6100.01h)

Bit 3 Messrichtung Lineargeber: Wert: 0 = steigend

Wert: 1 = fallend

**Zu Subindex 0x02 Safety preset value for encoder with normal resolution**

(identisch mit den Daten zu Index 6100.02h)

Setzeingang für den Anfangspunkt des Permanentmagneten, entspricht einer Nullpunktverschiebung.

Der Presetwert wird als Vielfaches der verwendeten Schrittweite angegeben.

**Zu Subindex 0x04h: Safety speed source selector**

Auswahl der Geschwindigkeits Quelle

Der Wert beträgt immer 2: Bezugswert ist der Positionswert auf Index 600Ch)

**Zu Subindex 0x05h: Safety speed integration time**

Zeitfenster, über das die Geschwindigkeit gerechnet wird

Anwender spezifische Einstellung: Min. = 20 ms, Max. = 1000 ms

**Zu Subindex 0x06h: Multiplier value**

Multiplikationswert

Der Multiplikationswert berechnet sich aus:

$1000 \times \text{Auflösung des Positionswertes} / \text{Auflösung des Geschwindigkeitswertes}$  (in den Werkseinstellungen hinterlegt)

**Zu Subindex 0x07h: Divider value**

Divisionswert

Der Divisionswert beträgt immer 1000.

**10.2.11 Index 6120h: Safety normal resolution position sequence**

Dieser Eintrag enthält den sicheren Positionswert (*Safety normal resolution position sequence*, [4])

Der Eintrag kann in TPDOs und SRDOs gemappt werden.

**Verwendete Subindizes:**

- 0: Highest sub-index supported (const)**
- 1: 1. Byte Safety normal resolution position value (read only)**
- 2: 2. Byte Safety normal resolution position value (read only)**
- 3: 3. Byte Safety normal resolution position value (read only)**
- 4: 4. Byte Safety normal resolution position value (read only)**

**10.2.12 Index 6121h: Safety inverted normal resolution position sequence**

Dieser Eintrag enthält den invertierten sicheren Positionswert (*Safety inverted normal resolution position sequence*, [4])

Der Eintrag kann in SRDOs gemappt werden.

**Verwendete Subindizes:**

- 0: Highest sub-index supported (const)**
- 1: 1. Byte Safety normal inverted resolution position value (read only)**
- 2: 2. Byte Safety normal inverted resolution position value (read only)**
- 3: 3. Byte Safety normal inverted resolution position value (read only)**
- 4: 4. Byte Safety normal inverted resolution position value (read only)**

**10.2.13 Index 6124h: Safety speed sequence**

Dieser Eintrag enthält den sicheren Geschwindigkeitswert (*Safety speed sequence*, [4])

Ob der Eintrag existiert, ist abhängig von den Werkseinstellungen.

Der Eintrag kann in TPDOs und SRDOs gemappt werden.

**Verwendete Subindizes:**

- 0: Highest sub-index supported (const)**
- 1: 1. Byte Safety speed value (read only)**
- 2: 2. Byte Safety speed value (read only)**

**10.2.14 Index 6125h: Safety inverted speed sequence**

Dieser Eintrag enthält den invertierten sicheren Geschwindigkeitswert (*Safety speed sequence*, [4])

Ob der Eintrag existiert, ist abhängig von den Werkseinstellungen.

Der Eintrag kann in SRDOs gemappt werden.

**Verwendete Subindizes:**

- 0: Highest sub-index supported (const)**
- 1: 1. Byte Safety inverted speed value (read only)**
- 2: 2. Byte Safety inverted speed value (read only)**

**10.2.15 Index 61FEh: Safety application configuration valid (read write)**

Dieser Eintrag enthält das Configuration-Valid-Flag, der Eintrag setzt den Marker für die gültige Konfiguration der Sicherheitsanwendung (*Safety application configuration valid*, [4])

### 10.2.16 Index 61FFh: Safety application configuration signature

Dieser Eintrag enthält die Prüfsummen für die sicherheitsrelevanten Einstellungen (*Safety configuration signature*, [4])

#### Verwendete Subindizes:

**0: Highest sub-index supported (const)**

**1: Safety position configuration signature (read write)**

Prüfsumme für sichere Positionseinstellungen

**2: Safety speed configuration signature (read write)**

Prüfsumme für sichere Geschwindigkeitseinstellungen.

Ob der Eintrag existiert, ist abhängig von den Werkseinstellungen.

### 10.2.17 Index 6200h: Cyclic timer (read write)

Eventtimer

Der Eventtimer entspricht dem Eventtimer des TPDO1, der über Index 1800.05 bereitgestellt wird.

### 10.2.18 Index 6500h: Operating status (read only)

Betriebsstatus [4]

Der Betriebsstatus entspricht den Betriebsparametern (Operating parameters, Index 6000.00).

### 10.2.19 Index 6501h: Measuring step (const)

Messschritt des Positionswertes [4].

Der Eintrag gibt die physikalische Auflösung als Vielfaches von 1nm an.

### 10.2.20 Index 6503h: Alarms (read only)

Alarme [4]

Der Eintrag kann in TPDOs gemappt werden.

Das Gerät unterstützt nur das Alarmsignal Positionsfehler.

Bit 0 Position error

0 kein Alarm

1 Alarm

### 10.2.21 Index 6504h: Supported alarms (const)

Unterstützte Alarme [4]

Das Gerät unterstützt nur das Alarmsignal Positionsfehler, der Wert ist immer 1.

Bit 0 1 Position error

### 10.2.22 Index 6505h: Warnings (read only)

Warnungen [4].

Das Gerät unterstützt keine Warnungen, der Wert ist immer 0.

**10.2.23 Index 6506h: Supported warnings (const)**

Unterstützte Warnungen [4].

Das Gerät unterstützt keine Warnungen, der Wert ist immer 0.

**10.2.24 Index 6507h: Profile and software version (const)**

Liefert die Gerätesoftware und die Profilversion zurück [4].

Bit 0...15     Profilversion

Bit 16...23   Softwareversion

Bit 24...31:   Softwarerelease

**10.2.25 Index 6508h: Operating time (read only)**

Betriebszeit des Gerätes

Das Gerät unterstützt keine Betriebszeit, der Wert ist immer 0xFFFFFFFF

**10.2.26 Index 6509h: Offset value (read only)**

Offset Wert [4]

Der Offset Wert ergibt sich durch das Setzen des Preset Wertes, und wird als Vielfaches der verwendeten Schrittweite angegeben.

Der Offset wird bei der Ausgabe des Positionswertes von dem gemessenen Wert subtrahiert.

**10.2.27 Index 650Ah: Module identification (read only)**

Modul Identifikation

Unter diesem Index sind die Herstellerangaben zum Offset und der minimalen und maximalen Position hinterlegt [4].

**Verwendete Subindizes:****0: Highest sub-index supported (const)****1: Manufacturer offset value (read only)**

Hersteller Offset Wert, der Wert beträgt immer 0.

**2: Manufacturer min position value (read only)**

Hersteller Min. Positionswert: Nullposition des Magneten. Der Wert beträgt immer 0.

**3: Manufacturer max position value (read only)**

Hersteller Max. Positionswert: Messlänge (Länge des Stabes). Die Länge des Stabes wird in der Positionsschrittweite gesendet.

**10.2.28 Index 650Bh: Serial number (read only)**

Der Eintrag entspricht dem Eintrag des Index 1018.04h (Geräteidentifikation, Seriennummer) [4].

### 10.2.29 Index 650Eh: Device capability (read only)

Gerätefähigkeit

Die Encoderklasse ist immer 2 (Class 2), die Auflösung 0 (Normal). Dieser Eintrag gibt die Gerätefähigkeit an, liefert 22h zurück (*Device capability*, [4]).

## 10.3 Manufacturer Specific Entries

Herstellerspezifische Objekte sind meist auch gerätespezifisch. Unter diesem Kapitel sind diejenigen Objekte beschrieben welche von den Geräten im Allgemeinen immer unterstützt werden.

### 10.3.1 Index 2001h: Node-ID

Unter diesem Index ist die *Node-ID* hinterlegt, das Objekt dient der Verwaltung der Geräteadresse.

#### Verwendete Subindizes:

- 0: Highest sub-index supported (const)**
- 1: Active node-ID (read only)**
- 2: Pending node-ID (read write)**  
Gewünschte Änderung der Geräteadresse

Damit die neue Node-ID wirksam wird, muss erst der Befehl *StoreLssParameters* übertragen werden und danach der Knoten neu gestartet werden.

### 10.3.2 Index 2002h: Baudrate

Das Objekt dient der Verwaltung der Baurate.

0	1000 kbit/s	
1	800 kbit/s	
2	500 kbit/s	
3	<b>250 kbit/s</b>	<b>Standard-Voreinstellung</b>
4	125 kbit/s	
5	100 kbit/s	[4]: reserviert (wird nicht von jedem Gerät unterstützt)
6	50 kbit/s	
7	20 kbit/s	
8	10 kbit/s	

#### Verwendete Subindizes:

- 0: Highest sub-index supported (const)**
- 1: Active Baudrate (read only)**
- 2: Pending Baudrate (read write)**  
Gewünschte Änderung der Baudrate

Damit die neue Baudrate wirksam wird, muss erst der Befehl *StoreLssParameters* übertragen werden und danach der Knoten neu gestartet werden.

### 10.3.3 Index 2010h: Checksum (read only)

Dieser Eintrag enthält die Checksumme der aktuellen Geräte-Software.

### 10.3.4 Weitere Indizes im Bereich 2000h bis 5FFFh (reserved)

**Achtung:**

Die Indizes in diesem Bereich enthalten **wichtige Werkseinstellungen und dürfen kundenseitig nicht verändert werden!**  
Jeder **Eingriff** kann zur **Fehlfunktion** des Winkelgebers führen!

## 11 Layer setting services (LSS) und Protokolle

Die LSS-Dienste und Protokolle, dokumentiert in CiA DS305 V2.2, siehe [4], unterstützen das Abfragen und Konfigurieren verschiedener Parameter des Data Link Layers und des Application Layers eines LSS-Slaves durch ein LSS-Master über das CAN Netzwerk.

Unterstützt werden folgende Parameter:

- Node-ID
- Baudrate
- LSS-Adresse, gemäß dem Identity Objekt 1018h

Der Zugriff auf den LSS-Slave erfolgt dabei über seine LSS-Adresse, bestehend aus:

- Vendor-ID
- Produkt-Code
- Revisions-Nummer und
- Serien-Nummer

Das Mess-System unterstützt folgende Dienste:

#### Switch mode services

- Switch mode selective
  - einen bestimmten LSS-Slave ansprechen
- Switch mode global
  - alle LSS-Slaves ansprechen

#### Configuration services

- Configure Node-ID
  - Node-ID konfigurieren
- Configure bit timing parameters
  - Baudrate konfigurieren
- Activate bit timing parameters
  - Baudrate aktivieren
- Store configured parameters
  - konfigurierte Parameter speichern



## Inquiry services

- Inquire LSS-address
  - LSS-Adresse anfragen
- Inquire Node-ID
  - Node-ID anfragen

## Identification services

- LSS identify remote slave
  - Identifizierung von LSS-Slaves innerhalb eines bestimmten Bereichs
- LSS identify slave
  - Rückmeldung der LSS-Slaves auf das vorherige Kommando
- LSS identify non-configured remote slave
  - Identifizierung von nicht-konfigurierten LSS-Slaves, Node-ID = FFh
- LSS identify non-configured slave
  - Rückmeldung der LSS-Slaves auf das vorherige Kommando

**11.1 Finite state automaton, FSA**

Der FSA entspricht einer Zustandsmaschine und definiert das Verhalten eines LSS-Slaves. Gesteuert wird die Zustandsmaschine durch LSS COBs erzeugt durch einen LSS-Master, oder NMT COBs erzeugt durch einen NMT-Master, oder lokale NMT-Zustandsübergänge.

**Die LSS-Modes unterstützen folgende Zustände:**

- (0) Initial: Pseudo-Zustand, zeigt die Aktivierung des FSAs an**
- (1) LSS waiting: Unterstützung aller Dienste wie unten angegeben**
- (2) LSS configuration: Unterstützung aller Dienste wie unten angegeben**
- (3) Final: Pseudo-Zustand, zeigt die Deaktivierung des FSAs an**

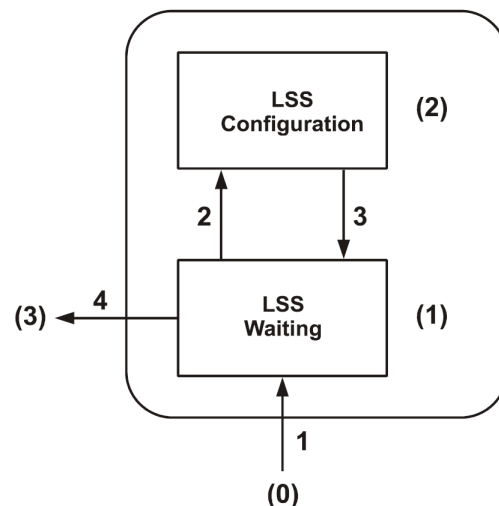


Abbildung 1: LSS-Modes

## Zustandsverhalten der unterstützten Dienste

Dienste	Waiting	Configuration
Switch mode global	Ja	Ja
Switch mode selective	Ja	Nein
Activate bit timing parameters	Nein	Ja
Configure bit timing parameters	Nein	Ja
Configure Node-ID	Nein	Ja
Store configuration	Nein	Ja
Inquire LSS-address	Nein	Ja
LSS identify remote slave	Ja	Ja
LSS identify slave	Ja	Ja
LSS identify non-configured remote slave	Ja	Ja
LSS identify non-configured slave	Ja	Ja

## LSS FSA Zustandsübergänge

Übergang	Ereignisse	Aktionen
1	Automatischer Übergang nach der Initialisierung beim Eintritt entweder in den NMT PRE OPERATIONAL Zustand oder NMT STOPPED Zustand, oder NMT RESET COMMUNICATION Zustand mit Node-ID = FFh.	keine
2	LSS 'switch state global' Kommando mit Parameter 'configuration_switch' oder 'switch state selective' Kommando	keine
3	LSS 'switch state global' Kommando mit Parameter 'waiting_switch'	keine
4	Automatischer Übergang, wenn eine ungültige Node-ID geändert wurde und die neue Node-ID erfolgreich im nichtflüchtigen Speicher abgelegt werden konnte UND der Zustand LSS waiting angefordert wurde.	keine

Sobald das LSS FSA weitere Zustandsübergänge im NMT FSA von NMT PRE OPERATIONAL auf NMT STOPPED und umgekehrt erfährt, führt dies nicht zum Wiedereintritt in den LSS FSA.

## 11.2 Übertragung von LSS-Diensten

Über die LSS-Dienste fordert der LSS-Master die einzelnen Dienste an, welche dann durch den LSS-Slave ausgeführt werden. Die Kommunikation zwischen LSS-Master und LSS-Slave wird über die implementierten LSS-Protokolle vorgenommen.

Ähnlich wie bei der SDO-Übertragung, werden auch hier zwei COB-IDs für das Senden und Empfangen benutzt:

COB-ID	Bedeutung
7E4h	LSS-Slave → LSS-Master
7E5h	LSS-Master → LSS-Slave

Tabelle 1: COB-IDs für LSS Services

### 11.2.1 LSS-Nachrichtenformat

Der maximal 8 Byte lange Datenbereich einer CAN-Nachricht wird von einem LSS-Dienst wie folgt belegt:

CS	Daten						
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7

Tabelle 2: LSS-Nachricht

Byte 0 enthält die **Command-Specifier** (CS), danach folgen 7 Byte für die Daten.

## 11.3 Switch mode Protokolle

### 11.3.1 Switch mode global Protokoll

Das angegebene Protokoll hat den *Switch mode global service* implementiert und steuert das Zustandsverhalten des LSS-Slaves. Über den LSS-Master können alle LSS-Slaves im Netzwerk in den *Waiting Mode* oder *Configuration Mode* gebracht werden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	CS	Mode	Reserved by CiA					
7E5h	04	0 = Waiting Mode 1 = Configuration Mode						

### 11.3.2 Switch mode selective Protokoll

Das angegebene Protokoll hat den *Switch mode selective service* implementiert und steuert das Zustandsverhalten des LSS-Slaves. Über den LSS-Master kann nur der LSS-Slave im Netzwerk in den *Configuration Mode* gebracht werden, dessen LSS-Adressattribute der LSS-Adresse entsprechen.

LSS-Master --&gt; LSS-Slave

	0	1	2	3	4	5	6	7
<b>COB-ID</b>	<b>CS</b>	<b>Vendor-ID</b>				Reserved by CiA		
7E5h	64	LSB				MSB		

	0	1	2	3	4	5	6	7
<b>COB-ID</b>	<b>CS</b>	<b>Product-Code</b>				Reserved by CiA		
7E5h	65	LSB				MSB		

	0	1	2	3	4	5	6	7
<b>COB-ID</b>	<b>CS</b>	<b>Revision-Number</b>				Reserved by CiA		
7E5h	66	LSB				MSB		

	0	1	2	3	4	5	6	7
<b>COB-ID</b>	<b>CS</b>	<b>Serial-Number</b>				Reserved by CiA		
7E5h	67	LSB				MSB		

LSS-Slave --&gt; LSS-Master

	0	1	2	3	4	5	6	7
<b>COB-ID</b>	<b>CS</b>	Reserved by CiA						
7E4h	68							

## 11.4 Configuration Protokolle

### 11.4.1 Configure Node-ID Protokoll

Das angegebene Protokoll hat den *Configure NMT-Address service* implementiert. Über den LSS-Master kann die Node-ID eines einzelnen LSS-Slaves im Netzwerk konfiguriert werden. Hierbei darf sich nur ein LSS-Slave im *Configuration Mode* befinden. Zur Speicherung der neuen Node-ID muss das *Store configuration protocol* an den LSS-Slave übertragen werden. Um die neue Node-ID zu aktivieren, muss der NMT-Dienst *Reset Communication* (82h) aufgerufen werden.

LSS-Master --&gt; LSS-Slave

	0	1	2	3	4	5	6	7
<b>COB-ID</b>	<b>CS</b>	<b>Node-ID</b>	Reserved by CiA					
7E5h	17	1...127 und 255						

LSS-Slave --&gt; LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	CS	Error Code	Spec. Error	Reserved by CiA				
7E4h	17							

Error Code

0: Ausführung erfolgreich  
 1...254: Reserved  
 255: applikationsspezifischer Fehler aufgetreten

Specific Error

Wenn Error Code = 255 --> applikationsspezifischer Fehler aufgetreten,  
 sonst reserviert durch die CiA

### 11.4.2 Configure bit timing parameters Protokoll

Das angegebene Protokoll hat den *Configure bit timing parameters service* implementiert. Über den LSS-Master kann die Baudrate eines einzelnen LSS-Slaves im Netzwerk konfiguriert werden. Zur Speicherung der neuen Baudrate muss das *Store configuration protocol* an den LSS-Slave übertragen werden.

LSS-Master --&gt; LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	CS	Table Selector	Table Index	Reserved by CiA				
7E5h	19	0	0..8					

LSS-Slave --&gt; LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	CS	Error Code	Spec. Error	Reserved by CiA				
7E4h	19							

Table Selector

0: Standard CiA Baudraten-Tabelle

Table Index

0: 1 Mbit/s  
 1: 800 kbit/s  
 2: 500 kbit/s  
 3: 250 kbit/s  
 4: 125 kbit/s  
 5: **reserved!**  
 6: 50 kbit/s  
 7: 20 kbit/s  
 8: 10 kbit/s

**Error Code**

- 0: Ausführung erfolgreich
- 1: selektierte Baudrate nicht unterstützt
- 2...254: reserved
- 255: applikationsspezifischer Fehler aufgetreten

**Specific Error**

Wenn Error Code = 255 --> applikationsspezifischer Fehler aufgetreten,  
sonst reserviert durch die CiA

**11.4.3 Activate bit timing parameters Protokoll**

Das angegebene Protokoll hat den *Activate bit timing parameters service* implementiert und aktiviert die über *Configure bit timing parameters protocol* festgelegte Baudrate bei allen LSS-Slaves im Netzwerk, die sich im *Configuration Mode* befinden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	CS	Switch Delay [ms]		Reserved by CiA				
7E5h	21	LSB	MSB					

**Switch Delay**

Der Parameter *Switch Delay* definiert die Länge zweier Verzögerungsperioden (D1, D2) mit gleicher Länge. Damit wird das Betreiben des Busses mit unterschiedlichen Baudratenparametern verhindert. Nach Ablauf der Zeit D1 und einer individuellen Verarbeitungsdauer wird die Umschaltung intern im LSS-Slave vorgenommen. Nach Ablauf der Zeit D2 meldet sich der LSS-Slave wieder mit CAN-Nachrichten und der neu eingestellten Baudrate.

Es gilt:

Switch Delay > längste vorkommende Verarbeitungsdauer eines LSS-Slaves

**11.4.4 Store configuration Protokoll**

Das angegebene Protokoll hat den *Store configured parameters service* implementiert. Über den LSS-Master können die konfigurierten Parameter eines einzelnen LSS-Slaves im Netzwerk in den nichtflüchtigen Speicher abgelegt werden. Hierbei darf sich nur ein LSS-Slave im *Configuration Mode* befinden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	CS	Reserved by CiA						
7E5h	23							

LSS-Slave --&gt; LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	CS	Error Code	Spec. Error	Reserved by CiA				
7E4h	23							

Error Code

- 0: Ausführung erfolgreich  
 1: *Store configuration* nicht unterstützt  
 2...254: Reserved  
 255: applikationsspezifischer Fehler aufgetreten

Specific Error

Wenn Error Code = 255 --> applikationsspezifischer Fehler aufgetreten,  
 sonst reserviert durch die CiA

## 11.5 Inquire LSS-Address Protokolle

### 11.5.1 Inquire Identity Vendor-ID Protokoll

Das angegebene Protokoll hat den *Inquire LSS-Address service* implementiert. Über den LSS-Master kann die Vendor-ID eines einzelnen LSS-Slaves im Netzwerk ausgelesen werden. Hierbei darf sich nur ein LSS-Slave im *Configuration Mode* befinden.

LSS-Master --&gt; LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	CS	Reserved by CiA						
7E5h	90							

LSS-Slave --&gt; LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	CS	Vendor-ID (= Index 1018h:01)				Reserved by CiA		
7E4h	90	LSB		MSB				

### 11.5.2 Inquire Identity Product-Code Protokoll

Das angegebene Protokoll hat den *Inquire LSS-Address service* implementiert. Über den LSS-Master kann der Hersteller-Gerätenamen eines einzelnen LSS-Slaves im Netzwerk ausgelesen werden. Hierbei darf sich nur ein LSS-Slave im *Configuration Mode* befinden.

LSS-Master --&gt; LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	CS	Reserved by CiA						
7E5h	91							

LSS-Slave --&gt; LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	CS	Product-Code (= Index 1018h:02)				Reserved by CiA		
7E4h	91	LSB				MSB		

### 11.5.3 Inquire Identity Revision-Number Protokoll

Das angegebene Protokoll hat den *Inquire LSS-Address service* implementiert. Über den LSS-Master kann die Revisionsnummer eines einzelnen LSS-Slaves im Netzwerk ausgelesen werden. Hierbei darf sich nur ein LSS-Slave im *Configuration Mode* befinden.

LSS-Master --&gt; LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	CS	Reserved by CiA						
7E5h	92							

LSS-Slave --&gt; LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	CS	Revision-Number (= Index 1018h:03)				Reserved by CiA		
7E4h	92	LSB				MSB		

### 11.5.4 Inquire Identity Serial-Number Protokoll

Das angegebene Protokoll hat den *Inquire LSS-Address service* implementiert. Über den LSS-Master kann die Seriennummer eines einzelnen LSS-Slaves im Netzwerk ausgelesen werden. Hierbei darf sich nur ein LSS-Slave im *Configuration Mode* befinden.

LSS-Master --&gt; LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	CS	Reserved by CiA						
7E5h	93							

LSS-Slave --&gt; LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	CS	Serial-Number (= Index 1018h:04)				Reserved by CiA		
7E5h	93	LSB				MSB		



### 11.5.5 Inquire Node-ID Protokoll

Das angegebene Protokoll hat den *Inquire Node-ID service* implementiert. Über den LSS-Master kann die Node-ID eines einzelnen LSS-Slaves im Netzwerk ausgelesen werden. Hierbei darf sich nur ein LSS-Slave im *Configuration Mode* befinden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	CS	Reserved by CiA						
7E5h	94							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	CS	Node-ID	Reserved by CiA					
7E4h	94	1...127 und 255						

#### Node-ID

Entspricht der Node-ID des selektierten Gerätes. Wenn die Node-ID eben gerade erst über den *Configure Node-ID service* geändert wurde, wird die ursprüngliche Node-ID zurückgemeldet. Erst nach Ausführung des NMT-Dienstes *Reset Communication* (82h) wird die aktuelle Node-ID zurückgemeldet.

## 11.6 Identification Protokolle

### 11.6.1 LSS identify remote slave Protokoll

Das angegebene Protokoll hat den *LSS identify remote slaves service* implementiert. Über den LSS-Master können LSS-Slaves im Netzwerk in einem bestimmten Bereich identifiziert werden. Alle LSS-Slaves, die dem angegebenen Vendor-ID, Product-Code, Revision-No. – Bereich und Serial-No. – Bereich entsprechen, antworten mit dem *LSS identify slave protocol*.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	CS	Vendor-ID				Reserved by CiA		
7E5h	70	LSB		MSB				

	0	1	2	3	4	5	6	7
COB-ID	CS	Product-Code				Reserved by CiA		
7E5h	71	LSB		MSB				

	0	1	2	3	4	5	6	7
COB-ID	CS	Revision-Number-Low				Reserved by CiA		
7E5h	72	LSB				MSB		

	0	1	2	3	4	5	6	7
COB-ID	CS	Revision-Number-High				Reserved by CiA		
7E5h	73	LSB				MSB		

	0	1	2	3	4	5	6	7
COB-ID	CS	Serial-Number-Low				Reserved by CiA		
7E5h	74	LSB				MSB		

	0	1	2	3	4	5	6	7
COB-ID	CS	Serial-Number-High				Reserved by CiA		
7E5h	75	LSB				MSB		

### 11.6.2 LSS identify slave Protokoll

Das angegebene Protokoll hat den *LSS identify slave service* implementiert. Alle LSS-Slaves, die den im *LSS identify remote slaves protocol* angegebenen LSS-Adress-Attributen entsprechen, antworten mit diesem Protokoll.

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	CS	Reserved by CiA						
7E4h	79							

### 11.6.3 LSS identify non-configured remote slave Protokoll

Das angegebene Protokoll hat den *LSS identify non-configured remote slave service* implementiert. Über den LSS-Master werden alle nichtkonfigurierten LSS-Slaves (Node-ID = FFh) im Netzwerk identifiziert. Die betreffenden LSS-Slaves antworten mit dem *LSS identify non-configured remote slave protocol*.

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	CS	Reserved by CiA						
7E4h	76							

### 11.6.4 LSS identify non-configured slave Protokoll

Das angegebene Protokoll hat den LSS *identify non-configured slave service* implementiert. Alle LSS-Slaves, die eine ungültige Node-ID (FFh) besitzen, antworten nach Ausführung *des LSS identify non-configured slave protocol* mit diesem Protokoll.

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
<b>COB-ID</b>	<b>CS</b>	Reserved by CiA						
7E4h	80							

## 12 Anschluss

Der Anschluss kann mit Hilfe der beigelegten gerätespezifischen Steckerbelegung durchgeführt werden, siehe Bedienanleitung HLT CANopen Safety (Die Bedienanleitung ist Teil des Lieferumfangs HLT CANopen Safety).

### 12.1 Einschalten der Versorgungsspannung

Nachdem der Anschluss und alle Einstellungen vorgenommen worden sind, kann die Versorgungsspannung eingeschaltet werden.

Nach dem Einschalten der Versorgungsspannung und Beendigung der Initialisierung geht das Mess-System in den Vor-Betriebszustand (PRE-OPERATIONAL). Dieser Zustand wird durch die Boot-Up-Meldung „**COB-ID 700h+Node-ID**“ bestätigt. Falls das Mess-System einen internen Fehler erkennt, wird eine Emergency-Meldung mit dem Fehlercode übertragen (siehe 7.5).

Im PRE-OPERATIONAL-Zustand ist zunächst nur eine Parametrierung über Service-Daten-Objekte möglich. Es ist aber möglich, PDOs unter Nutzung von SDOs zu konfigurieren. Ist das Mess-System in den Zustand OPERATIONAL überführt worden, ist auch eine Übertragung von PDOs möglich (siehe 7.7).

### 12.2 Einstellen der Node-ID mittels LSS-Dienste

#### 12.2.1 Konfiguration der Node-ID, Ablauf

Annahme:

- LSS-Adresse unbekannt
- der LSS-Slave ist der einzige Teilnehmer in Netzwerk
- es soll die Node-ID 12 dez. eingestellt werden

Vorgehensweise:

- LSS-Slave mit dem Command Specifier 04 *Switch mode global protocol*, Mode = 1 in den *Configuration Mode* bringen. (siehe 11.3)
- Command Specifier 17 *Configure NMT-Address protocol*, Node-ID = 12 ausführen. (siehe 11.4.1)  
--> Rückmeldung abwarten und erfolgreiche Ausführung überprüfen,  
--> Error Code = 0
- Command Specifier 23 *Store configuration protocol* ausführen.  
--> Rückmeldung abwarten und erfolgreiche Ausführung überprüfen,  
--> Error Code = 0  
(siehe 11.4.4)
- Versorgungsspannung des LSS-Slaves aus-, danach wieder einschalten. Die neue Konfiguration ist jetzt aktiv.

### 12.2.2 Konfiguration der Baudrate, Ablauf

Annahme:

- LSS-Adresse unbekannt
- der LSS-Slave ist der einzige Teilnehmer in Netzwerk
- es soll die Baudrate 125 kbit/s eingestellt werden

Vorgehensweise:

- NMT- Command Specifier *Stop Remote Node* (02h) aufrufen, um den LSS-Slave in den *Stopped state* zu bringen. Der LSS-Slave sollte keine CAN-Nachrichten mehr senden --> Heartbeat abgeschaltet (siehe 7.7).
- LSS-Slave mit dem Command Specifier *Switch mode global protocol*, Mode = 1 in den *Configuration Mode* bringen. (siehe 11.3.1).
- Command Specifier 19 *Configure bit timing parameters protocol* ausführen, Table Selector = 0, Table Index = 4  
--> Rückmeldung abwarten und erfolgreiche Ausführung überprüfen,  
--> Error Code = 0  
(siehe 11.4.2).
- Command Specifier 21 *Activate bit timing parameters protocol* aufrufen, damit die neue Baudrate aktiv wird. (siehe 11.4.3).

- Command Specifier 23 *Store configuration protocol* ausführen.  
--> Rückmeldung abwarten und erfolgreiche Ausführung überprüfen,  
--> Error Code = 0  
(siehe 11.4.4).
- Versorgungsspannung des LSS-Slaves aus-, danach wieder einschalten. Die neue Konfiguration ist jetzt aktiv.

## 13 Inbetriebnahme

### 13.1 CAN – Schnittstelle

Die CAN-Bus-Schnittstelle ist durch die internationale Norm ISO/DIS 11898 definiert und spezifiziert die zwei untersten Schichten des CAN Referenz-Models.

Die Konvertierung der Mess-System-Information in das CAN-Protokoll (CAN 2.0A) geschieht über einen CAN-Kontroller. Die Funktion des CAN-Kontrollers wird durch einen Watchdog überwacht.

Das CANopen Kommunikationsprofil (CiA Standard DS301, siehe [1]) basiert auf dem CAN Application Layer (CAL) und beschreibt, wie die Dienste von Geräten benutzt werden. Das CANopen Profil erlaubt die Definition von Geräteprofilen für eine dezentralisierte E/A.

Das Mess-System mit CANopen Protokoll unterstützt das Geräteprofil für Encoder (CiA DS406, Version 4.1.0 siehe [4]).

Die Kommunikations-Funktionalität und Objekte, welche im Messgeräteprofil benutzt werden, werden in einer EDS-Datei (Electronic Data Sheet) beschrieben. Wird ein CANopen Konfigurations-Hilfsprogramm benutzt (z.B. CANSETTER), kann der Benutzer die Objekte (SDO's) des Mess-Systems auslesen und die Funktionalität programmieren.

### 13.2 EDS-Datei

Die EDS-Datei (elektronisches Datenblatt) enthält alle Informationen über die Mess-System-spezifischen Parameter sowie Betriebsarten des Mess-Systems. Die EDS-Datei wird durch das CANopen-Netzwerkkonfigurationswerkzeug eingebunden, um das Mess-System ordnungsgemäß zu konfigurieren bzw. in Betrieb nehmen zu können.

Die EDS-Datei die zum Gerät passt, kann am Gerätenamen und der dem Gerät entsprechenden Major-Revisionsnummer des Gerätes im Dateinamen erkannt werden

Die Datei inklusive der vorliegenden CANopen Safety-Schnittstellenbeschreibung befindet sich zum Download auf unserer Homepage unter

<http://www.hydac.com/de-de/produkte/sensorik/show/Material/index.html>

**HYDAC ELECTRONIC GMBH**

Hauptstr. 27  
D-66128 Saarbrücken  
Germany

Web: [www.hydac.com](http://www.hydac.com)  
E-Mail: [electronic@hydac.com](mailto:electronic@hydac.com)  
Tel.: +49 (0)6897 509-01  
Fax.: +49 (0)6897 509-1726

**HYDAC Service**

Für Fragen zu Reparaturen steht Ihnen der HYDAC Service zur Verfügung.

**HYDAC SERVICE GMBH**

Hauptstr. 27  
D-66128 Saarbrücken  
Germany

Tel.: +49 (0)6897 509-1936  
Fax.: +49 (0)6897 509-1933

**Anmerkung**

Die Angaben in dieser Bedienungsanleitung beziehen sich auf die beschriebenen Betriebsbedingungen und Einsatzfälle. Bei abweichenden Einsatzfällen und/oder Betriebsbedingungen wenden Sie sich bitte an die entsprechende Fachabteilung.

Bei technischen Fragen, Hinweisen oder Störungen nehmen Sie bitte Kontakt mit Ihrer HYDAC-Vertretung auf.