



TSL- / TSR-CAN-03

Temperatursensor mit CANopen-Protokoll und LSS-Slave-Funktion (SW-Version 3.0)

Fluid.iO Sensor + Control GmbH & Co. KG
Stand: 01.05.2025

Folgende Dokumentation ist geistiges Eigentum der Fluid.iO Sensor + Control GmbH & Co. KG.
Vervielfältigung, Reproduktion, Speicherung in Informations-Systemen oder Weitergabe dieser

Dokumentation, entweder ganz oder auszugsweise, ohne die ausdrückliche schriftliche Genehmigung der Fluid.iO Sensor + Control ist untersagt.

1	NETWORK MANAGEMENT	4
2	LISTE DER UNTERSTÜTZTEN INDIZES	5
3	SDO-KOMMUNIKATION	7
3.1	TELEGRAMMSTRUKTUR SDO	7
3.2	SDO-STRUKTUR NACH CiA-DS-301	7
3.2.1	Geräte-Profil (Index 1000h)	7
3.2.2	Fehler-Register (Index 1001h)	8
3.2.3	Synchronisations-Botschaft (Index 1005h)	9
3.2.4	Geräte-Bezeichnung (Index 1008h)	9
3.2.5	Hardware-Version (Index 1009h)	9
3.2.6	Software-Version (Index 100Ah)	10
3.2.7	Node-ID (Index 100Bh)	10
3.2.8	Guard Time (Index 100Ch)	11
3.2.9	Lifetime-Factor (Index 100Dh)	11
3.2.10	Parameter speichern (Index 1010h)	11
3.2.11	Default-Parameter laden (Index 1011h)	12
3.2.12	COB-ID Emergency Object (Index 1014h)	12
3.2.13	Identity Object (Index 1018h)	13
3.3	GERÄTESPEZIFISCHE SDO-STRUKTUR NACH CiA-DSP-404	15
3.3.1	Sensor-Typ (Index 6110h)	15
3.3.2	Physikalische Größe (Maßeinheit) (Index 6131h)	15
3.3.3	Anzahl der Dezimalstellen (Index 6132h)	16
3.3.4	Alarm-Typ (Index 6508h)	16
3.3.5	Alarm-Aktion (Index 6509h)	17
3.3.6	Abfrage des Alarm-Status (Index 6600h)	18
3.3.7	Abfrage der AD-Werte (Index 7100h)	18
3.3.8	Offset-Verschiebung (Index 7124h)	19
3.3.9	Abfrage des Prozeßwertes (Index 7130h)	19
3.3.10	Abfrage des Alarm-Input-Wertes (Index 7500h)	20
3.3.11	Alarm-Schwelle (Index 750Ah)	20
3.3.12	Alarm-Hysteresis (Index 750Bh)	21
3.4	SDO-FEHLERMELDUNGEN	21
4	PDO-KOMMUNIKATION	22
4.1	EINSTELLUNG SENDE-PDO	22
4.2	PDO-MAPPING	23
4.3	PDO-SENDUNG	23
5	EMERGENCY MESSAGES	24
6	NODE GUARDING	26
7	LSS-SLAVE-FUNKTIONEN	27
7.1	SWITCH MODE GLOBAL	27
7.2	SWITCH MODE SELECTIVE	28
7.3	CONFIGURE NODE ID	28
7.4	CONFIGURE BIT TIMING PARAMETERS	29
7.5	ACTIVATE BIT TIMING PARAMETERS	30
7.6	STORE CONFIGURATION PROTOCOL	30

1 Network Management

Nach dem Anschluß der Versorgungsspannung an dem Fluid.iO CAN-Sensor, meldet sich dieser durch das Senden der CAN-Message „Boot-up Message“. Dies ist ein CAN-Telegramm mit 1 Datenbyte und mit dem COB-Identifizier 1792d + Modul-ID. Wenn der Modul-ID 16 ist (Defaultwert), so wird der Identifizier 1808d empfangen. Der Sensor befindet sich anschließend im Pre-Operational Mode.

Vom Fluid.iO CAN-Sensor werden folgende Objects unterstützt:

Object	COB-ID (dez.)	Modul-ID	Bemerkung
Network Management	0	0	Start, Stop, Reset
SYNC	128	0	Senden PDO
EMERGENCY	129 – 255	1-127	vom Sensor
PDO	385 – 511	1-127	vom Sensor
SDO	1409 – 1535	1-127	vom Sensor
SDO	1537 – 1663	1-127	zum Sensor
Node-Guarding	1793 – 1919	1-127	Bidirektional

Network Management (COB-ID = 0)

Start Node

ID	DLC	Byte 1	Byte 2
0	2	01h	Node

Node = Modul-ID, 0 = alle Module

Über den Befehl „Start Node“ wird der CAN-Sensor in den Operational Modus gesetzt, d.h. er kann dann auch PDOs senden (Senden von Meßwerten).

Stop Node

ID	DLC	Byte 1	Byte 2
0	2	02h	Node

Node = Modul-ID, 0 = alle Module

Der Befehl „Stop Node“ setzt den CAN-Sensor in den Stopped Mode. Hier können nur NMT-Kommandos empfangen werden und Node-Guarding durchgeführt werden.

Enter Pre-Operational Mode

ID	DLC	Byte 1	Byte 2
0	2	80h	Node

Node = Modul-ID, 0 = alle Module

Dieser Befehl setzt den CAN-Sensor in den Pre-Operational Mode. Der Sensor ist voll arbeitsfähig, lediglich PDOs können nicht gesendet werden.

Reset Node

ID	DLC	Byte 1	Byte 2
0	2	81h	Node

Node = Modul-ID, 0 = alle Module

Über den Befehl „Reset Node“ wird ein Reset des CAN-Sensors ausgeführt. Nach dem Reset befindet sich der Knoten im Pre-Operational Modus und sendet die „Boot-up Message“ (siehe oben).

Reset Communication

ID	DLC	Byte 1	Byte 2
0	2	82h	Node

Node = Modul-ID, 0 = alle Module

Über den Befehl „Reset Communication“ wird lediglich ein Reset des CAN-Controllers ausgeführt. Danach befindet sich der Knoten im Pre-Operational Modus und sendet die „Boot-up Message“ (siehe oben).

2 Liste der unterstützten Indizes

Vom Fluid.iO CAN-Sensor werden folgende Indizes unterstützt:

<i>Index</i>	<i>Name</i>
1000h	Geräte-Profil
1001h	Fehler-Register
1005h	COB-ID SYNC-Message
1008h	Gerätebezeichnung
1009h	Hardware Version
100Ah	Software Version
100Bh	Node-ID
100Ch	Guard-Time
100Dh	Lifetime-Factor
1010h	Parameter speichern
1011h	Default-Parameter laden
1014h	COB-ID Emergency Object
1018h	Identity Object
1800h	Setup für PDO-Sendung
1A00h	Abfrage PDO-Mapping
6110h	Sensor-Typ
6131h	Physikalische Größe (Maßeinheit)
6132h	Anzahl der Dezimalstellen
6508h	Alarm-Typ
6509h	Alarm-Aktion
6600h	Alarm-Status
7100h	Abfrage des AD-Wertes
7124h	Offset-Verschiebung
7130h	Abfrage des Prozeßwertes
7500h	Abfrage des Alarm-Input-Wertes
750Ah	Alarm-Schwelle
750Bh	Alarm-Hysteresis

3 SDO-Kommunikation

3.1 Telegrammstruktur SDO

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
	8	CMD	Index		Sub-Index	Datenbytes			

Das Command Byte (**CMD**) hat folgende Bedeutung:

Lesezugriff des Servers auf CAN-Sensor	40h
CAN-Sensor antwortet	42h
Schreibzugriff des Servers auf CAN-Sensor	22h
CAN-Sensor antwortet	60h

Bei **Index** und **Datenbytes** wird das LSB zuerst übertragen!

Der Bereich des Kommunikations-Profiles befindet sich in den Indizes 1000h - 1fffh und beinhaltet alle Parameter, welche das CAN-Netzwerk betreffen. Dieser Bereich ist in allen CANopen-Geräten definiert.

3.2 SDO-Struktur nach CiA-DS-301

<i>Index</i>	<i>Name</i>
1000h	Geräte-Profil
1001h	Fehler-Register
1005h	COB-ID SYNC-Message
1008h	Gerätebezeichnung
1009h	Hardware Version
100Ah	Software Version
100Bh	Node-ID
100Ch	Guard-Time
100Dh	Lifetime-Factor
1010h	Parameter speichern
1011h	Default-Parameter laden
1014h	COB-ID Emergency Object
1018h	Identity-Object

3.2.1 Geräte-Profil (Index 1000h)

Beispiel: Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	00	10h	00	00	00	00	00

Die Berechnung der ID entnehmen Sie bitte aus dem Punkt "Network Management"
Als Antwort sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	00	10h	00	94h	01h	22h	00h

Die Datenbytes sind gemäß CANopen Standard DSP 404 festgelegt:

Byte 5 + Byte 6 = 0194h = 404d

(Device Profile Number: Measurement Devices)

Byte 7 + Byte 8 = 0022h = 22h

(Additional Informations: Das Gerät enthält einen Sensoreingang und eine Alarmfunktion).

Der Index 1000h hat den Status „Nur-Lese-Zugriff“. Subindizes werden nicht unterstützt. Jeder fehlerhafter Zugriff wird durch Senden einer entsprechenden SDO-Fehlermeldung beantwortet. Siehe hierzu "SDO-Fehlermeldungen".

3.2.2 Fehler-Register (Index 1001h)

Über den Index 1001h kann das Fehler-Register des Gerätes ausgelesen werden.

Beispiel: Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	01h	10h	00	00	00	00	00

Als Antwort wird in Byte 5 das Fehlerregister übertragen. Es werden folgende momentan anliegende Fehler angezeigt:

Generic Error

Bit 0 im Byte 5 ist gesetzt.

Der Generic Error wird ausgelöst bei einem Sensorfehler, d.h. wenn das Output-Word des Digitalensors nicht im definierten Bereich liegt. In diesem Fall liegt ein Hardwaredefekt vor.

EEPROM-Error

Bit 1 im Byte 5 ist gesetzt.

Der Fehler wird ausgelöst, wenn ein Schreiben auf den EEPROM (Datenspeicher) des CAN-Sensors nicht möglich ist.

Communication Error

Bit 4 im Byte 5 ist gesetzt.

Der Fehler wird ausgelöst bei Störungen in der Kommunikation auf dem CAN-Bus, und zwar, wenn einer der Fehlerzähler des CAN-Controllers den Wert von 95 überschritten hat.

Außerdem wird dieses Bit gesetzt, wenn Node-Guarding eingeschaltet ist und der NMT-Master den CAN-Sensor nicht mehr innerhalb der Node Life Time pollt.

Alarm-Error

Bit 5 im Byte 5 ist gesetzt.

Der Fehler ist aktiv, wenn sich der CAN-Sensor im Alarmzustand befindet, d.h. der Prozeßwert die eingegebenen Alarmgrenzwerte überschritten hat.

Der Index 1001h hat den Status „Nur-Lese-Zugriff“, Subindizes werden nicht unterstützt. Jeder fehlerhafter Zugriff wird durch Senden einer entsprechenden SDO-Fehlermeldung beantwortet. Siehe hierzu "SDO-Fehlermeldungen".

3.2.3 Synchronisations-Botschaft (Index 1005h)

Konfigurierung und Abfrage der Konfigurierung der SYNC-Message. Über die SYNC-Message kann ein Senden von PDOs ausgelöst werden (vgl. "PDO-Kommunikation"). Der Index ist wie folgt aufgebaut:

<i>Index</i>	<i>Parameter</i>	<i>Access</i>
1005h	ID (32 bit)	lesen/schreiben

Der 32-bit Parameterbereich ist wie folgt aufgebaut:

<i>Bit 31</i>	<i>Bit 30-11</i>	<i>Bit 10-0</i>
1	0	ID 11 Bit

Der Default-ID ist 80h. Dies gewährleistet den SYNC-Botschaften eine hohe Priorität auf dem CAN-Bus. Über das Bit 31 wird angezeigt, ob das Gerät SYNC-Message verarbeitet oder selbst generiert. Wenn das Bit 31 gesetzt ist, verarbeitet das Gerät SYNC-Message. Der CAN-Sensor kann die SYNC-Message nur verarbeiten, deswegen ist Bit 31 immer gesetzt, unabhängig davon, was der Server sendet.

3.2.4 Geräte-Bezeichnung (Index 1008h)

Über den Index 1008h kann die Geräte-Bezeichnung des Herstellers abgefragt werden. Diese ist herstellerspezifisch und wird als ASCII-Text übertragen.

Beispiel: Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	08h	10h	00	00	00	00	00

Als Antwort sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	08h	10h	00	5Ah	54h	53h	00h

Byte 5 = 5Ah entspricht ASCII = Z (ZILA)

Byte 6 = 44h entspricht ASCII = T (Temperatur)

Byte 7 = 53h entspricht ASCII = S (Sensor)

Der Index 1008h hat den Status „Nur-Lese-Zugriff“, Subindizes werden nicht unterstützt. Jeder fehlerhafter Zugriff wird durch Senden einer entsprechenden SDO-Fehlermeldung beantwortet. Siehe hierzu "SDO-Fehlermeldungen".

3.2.5 Hardware-Version (Index 1009h)

Über den Index 1009h kann die Hardware-Version abgefragt werden.

Beispiel: Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
----	-----	--------	--------	--------	--------	--------	--------	--------	--------

1552	8	40h	09h	10h	00	00	00	00	00
------	---	-----	-----	-----	----	----	----	----	----

Als Antwort sendet der CAN-Sensor

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	09h	10h	00	33h	2Eh	30h	00h

Das entspricht also der Version 3.0

Der Index 1009h hat den Status „Nur-Lese-Zugriff“, Subindizes werden nicht unterstützt. Jeder fehlerhafter Zugriff wird durch Senden einer entsprechenden SDO-Fehlermeldung beantwortet. Siehe hierzu „SDO-Fehlermeldungen“.

3.2.6 Software-Version (Index 100Ah)

Über den Index 100Ah kann die Software-Version abgefragt werden.

Beispiel: Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	0Ah	10h	00	00	00	00	00

Als Antwort sendet der CAN-Sensor die Versionsnummer in ASCII-codierter Form (vergleiche „Hardware-Version“).

Der Index 100Ah hat den Status „Nur-Lese-Zugriff“, Subindizes werden nicht unterstützt. Jeder fehlerhafter Zugriff wird durch Senden einer entsprechenden SDO-Fehlermeldung beantwortet. Siehe hierzu „SDO-Fehlermeldungen“.

3.2.7 Node-ID (Index 100Bh)

Über den Index 100Bh kann der Modul-ID abgefragt werden.

Beispiel: Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	0Bh	10h	00	00	00	00	00

Als Antwort sendet der CAN-Sensor in Byte 5 den Modul-ID:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	0Bh	10h	00	10h	00	00	00

Der Index 100Bh hat den Status „Nur-Lese-Zugriff“, Subindizes werden nicht unterstützt. Jeder fehlerhafter Zugriff wird durch Senden einer entsprechenden SDO-Fehlermeldung beantwortet. Siehe hierzu „SDO-Fehlermeldungen“.

Der Modul-Identifizierer kann nur über den LMT-Service (Layer Management) editiert werden.

3.2.8 Guard Time (Index 100Ch)

Über den Index 100Ch kann die Guard-Time in Millisekunden (16Bit-Integerwert) abgefragt oder eingetragen werden (Schreib-Lese-Zugriff). Die Guard-Time ist ein Parameter des Node-Guarding (siehe dort). Wird als Guard-Time Null eingetragen (Defaultwert), so ist die Node-Guarding-Funktion ausgeschaltet.

Beispiel: Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	0Ch	10h	00	00	00	00	00

Als Antwort sendet der CAN-Sensor in Byte 5 und 6 die Guard-Time, im Beispiel 500 ms:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	0Ch	10h	00	F4h	01h	00	00

Der Status unterstützt keine Subindizes. Jeder fehlerhafter Zugriff wird durch Senden einer entsprechenden SDO-Fehlermeldung beantwortet. Siehe hierzu "SDO-Fehlermeldungen".

3.2.9 Lifetime-Factor (Index 100Dh)

Über den Index 100Dh kann der Lifetime-Factor (8Bit-Integerwert) abgefragt oder eingetragen werden (Schreib-Lese-Zugriff). Der Lifetime-Factor ist ein Parameter des Node-Guarding (siehe dort). Multipliziert mit der Guard-Time ergibt er die Zeitspanne, innerhalb der der NMT-Master mindestens ein Remote-Request-Telegramm an den CAN-Sensor senden muß. Anderenfalls wird ein Life Guarding Event ausgelöst (Senden einer Emergency-Botschaft). Wird als Lifetime-Factor Null eingetragen (Defaultwert), so ist die Node-Guarding-Funktion ausgeschaltet.

Beispiel: Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	0Dh	10h	00	00	00	00	00

Als Antwort sendet der CAN-Sensor in Byte 5 den Lifetime-Factor, im Beispiel 3:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	0Ch	10h	00	03h	00	00	00

Der Status unterstützt keine Subindizes. Jeder fehlerhafter Zugriff wird durch Senden einer entsprechenden SDO-Fehlermeldung beantwortet. Siehe hierzu "SDO-Fehlermeldungen".

3.2.10 Parameter speichern (Index 1010h)

Die vom CAN-Sensor empfangenen Parameter werden nur flüchtig im RAM gespeichert, d.h. sie gehen bei Ausfall der Betriebsspannung verloren. Aus diesem Grunde muß nach jeder Parameteränderung, bzw. nach der Änderung mehrerer Parameter mit dem Index 1010h eine Speicherung der Parameter in einem nichtflüchtigen Speicher (EEPROM) durchgeführt werden. Das Abspeichern wird ausgelöst, indem der Index 1010h mit der Botschaft „save“ (in ASCII) auf dem Subindex 1 gesendet wird. Die Botschaft hat somit folgenden Aufbau:

Beispiel: Schreibzugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	22h	10h	10h	01h	73h	61h	76h	65h

Als Antwort sendet der CAN-Sensor nach abgeschlossener Speicherung:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	60h	10h	10h	01h	73h	61h	76h	65h

3.2.11 Default-Parameter laden (Index 1011h)

Über den Index 1011h kann der Default-Parametersatz des Gerätes geladen werden.

Er umfaßt folgende Parameter:

- 1 Nachkommastelle, d.h. full scale ist 80,0 °C
- SYNC-ID ist 80h
- PDO-ID ist 180h + Modul-ID
- Periodische Sendung der PDOs im Sekundentakt
- Alarmtyp: Alarm ist ausgeschaltet
- Alarmaktion: Sendung PDO bei Eintritt und Verlassen des Alarmfensters, Senden der Emergency-Message
- Alarm-Level ist 50,0 °C
- Alarm-Hysterese ist 10,0 K

Das Laden der Ursprungsparameter wird ausgelöst, indem der Index 1011h mit der Botschaft „load“ (in ASCII) auf dem Subindex 1 gesendet wird. Die Botschaft hat somit folgenden Aufbau:

Beispiel: Schreibzugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	22h	11h	10h	01h	6Ch	6Fh	61h	64h

Als Antwort sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	60h	11h	10h	01h	6Ch	6Fh	61h	64h

Das Laden des Default-Parametersatzes erfolgt in den RAM des Gerätes. Sollen diese Parameter nichtflüchtig gespeichert werden, so ist anschließend ein SDO mit dem Index 1010h zu senden (siehe oben).

3.2.12 COB-ID Emergency Object (Index 1014h)

Abfrage des COB-ID des Emergency-Telegrammes. Nur-Lese-Status. Der COB-ID beträgt 128d plus Modul-ID. Der Index ist wie folgt aufgebaut:

Index	Parameter	Access
1014h	ID (32 bit)	Lesen

Der 32-bit Parameterbereich ist wie folgt aufgebaut:

Bit 31	Bit 30-11	Bit 10-0
0	0	ID 11 Bit

3.2.13. Identity Object (Index 1018h)

Das Identitätsobjekt enthält grundlegende Informationen über den CANopen-Temperatursensor. Mittels dieser Angaben kann ein einzelner Temperatursensor in einem Netz zweifelsfrei identifiziert werden. Es werden insgesamt 5 Subindizes unterstützt, welche allesamt Nur-lese-Status besitzen:

Index	Sub-Index	Parameter	Access
1018h	0	Anzahl der unterstützten Sub-Indizes	lesen
	1	Vendor-ID	lesen
	2	Produkt-Code	lesen
	3	Haupt- und Neben-Revisionsnummer	lesen
	4	Serien-Nummer	Lesen

Vendor-ID

Die Vendor-ID ist eine 4 Byte lange Herstellernummer, welche der CiA dem Hersteller von CANopen-Geräten zuordnet. Die Vendor-ID für Geräte der Fluid.iO Sensor + Control lautet: 00h 00h 00h 0C0h

Beispiel: Lesezugriff des Servers auf Subindex 1, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	18h	10h	01h	00	00	00	00

Als Antwort sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	18h	10h	01h	0C0h	00h	00h	00h

Produkt-Code

Der Produkt-Code (ebenfalls 4 Byte lang) wird vom Hersteller für die jeweilige Geräteklasse vergeben. Der Produktcode für Fluid.iO Temperatursensoren ist 00h 00h 00h 02h

Als Antwort auf einen Lesezugriff auf Subindex 2 sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	18h	10h	02h	02h	00h	00h	00h

Haupt- und Neben-Revisions-Nummer

Die Revisions-Nummern entsprechen im Wesentlichen der Software-Versionsnummer, wobei hier noch feiner untergliedert wird: Die Hauptrevisions-Nummer (Byte 7 und 8) ändert sich nur mit steigender Funktionalität, d.h. wenn neue Funktionen hinzukommen. Die Nebenrevisions-Nummer ändert sich, wenn bei gleicher Funktionalität z.B. ein Fehler in der Software beseitigt wurde.

Als Antwort auf einen Lesezugriff auf Subindex 3 sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	18h	10h	03h	02h	00h	01h	00h

Serien-Nummer

Die Seriennummer (ebenfalls 4 Byte lang) wird bei der Herstellung jedem Gerät zugeteilt.

Als Antwort auf einen Lesezugriff auf Subindex 4 sendet der CAN-Sensor mit der Seriennummer 1:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	18h	10h	04h	01h	00h	00h	00h

3.3 Gerätespezifische SDO-Struktur nach CiA-DSP-404

Die gerätespezifische SDO-Struktur umfaßt folgende Indizes:

Index	Name
6110h	Sensor-Typ
6131h	Physikalische Größe (Maßeinheit)
6132h	Anzahl der Dezimalstellen
6508h	Alarm-Typ
6509h	Alarm-Aktion
6600h	Alarm-Status
7100h	Abfrage des AD-Wertes
7124h	Offset-Verschiebung
7130h	Abfrage des Prozeßwertes
7500h	Abfrage des Alarm-Input-Wertes
750Ah	Alarm-Schwelle
750Bh	Alarm-Hysteresis

3.3.1 Sensor-Typ (Index 6110h)

Die Abfrage des Sensor-Typs erfolgt über den Index 6110h. Der Index ist wie folgt aufgebaut:

Index	Sub-Index	Parameter	Access
6110h	0	Anzahl der unterstützten Subindizes	lesen
	1	Sensortyp	lesen

Beispiel: Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	10h	61h	01	00	00	00	00

Als Antwort sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	10h	61h	01	64h	00	00	00

Sensor-Typ = Byte 5 = 64h = 100d (Temperatur –Sensor)

3.3.2 Physikalische Größe (Maßeinheit) (Index 6131h)

Der Index 6131h ist ein Index mit Nur-Lese-Zugriff. Er gibt die Maßeinheit der Meßgröße an. Der Index ist wie folgt aufgebaut:

Index	Sub-Index	Parameter	Access
6131h	0	Anzahl der unterstützten Subindizes	lesen

	1	Maßeinheit	lesen
--	---	------------	-------

Beispiel: Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	31h	61h	01	00	00	00	00

Als Antwort sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	31h	61h	01	2Dh	00	00	00

Die Maßeinheit wird nach CiA DRP 303-2 V1.0 verschlüsselt und in Byte 5 dargestellt:

2Dh = °C

Byte 6 enthält die Zehnerpotenz der Maßeinheit (z.B. milli, mega, u.s.w.)

00h = 10⁰

3.3.3 Anzahl der Dezimalstellen (Index 6132h)

Der Index 6132h gibt die Anzahl der Dezimalstellen des Meßwertes an. Der Index ist wie folgt aufgebaut:

Index	Sub-Index	Parameter	Access
6132h	0	Anzahl der unterstützten Subindizes	lesen
	1	Anzahl Dezimalstellen	Lesen

Lesezugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	40h	32h	61h	01	00	00	00	00

Als Antwort sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	32h	61h	01	01h	00	00	00

Die Anzahl der Dezimalstellen ist bei diesem Sensor fest auf 1 eingestellt.

3.3.4 Alarm-Typ (Index 6508h)

Der Index 6508h gibt den Alarm-Typ an. Der Index ist wie folgt aufgebaut:

Index	Sub-Index	Parameter	Access
-------	-----------	-----------	--------

6508h	0	Anzahl der unterstützten Subindizes	lesen
	1	Alarm-Typ	lesen/ schreiben

Alarm-Typ	Alarm, wenn Meßwert...
0	keine Alarmfunktion
2	größer oder gleich Level
3	unter dem Level
6	im Fenster (Level, Level+Hyster.)
7	außerhalb des Fensters

Beispiel: Es ist erwünscht, daß der Sensor alarmiert, wenn der Meßwert die Alarmschwelle unterschreitet (Alarm-Typ = 3).

Schreibzugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	22h	08h	65h	01	03h	00	00	00

Als Antwort sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	60h	08h	65h	01	03h	00	00	00

Sollte der gesendete Alarm-Typ ungültig sein, werden die Änderungen nicht übernommen und im Byte 5 kommt der vorherige Wert zurück.

3.3.5 Alarm-Aktion (Index 6509h)

Der Index 6509h legt fest, was im Alarmfall für eine Reaktion erfolgen soll. Der Index ist wie folgt aufgebaut:

Index	Sub-Index	Parameter	Access
6509h	0	Anzahl der unterstützten Subindizes	lesen
	1	Alarm Aktions-Nr.	lesen/ schreiben

Alarm-Aktions-Nr.	Bedeutung
Bit 0 = 1	Emergency Message bei Alarmbeginn und –ende senden
Bit 1 = 1	Reserved
Bit 2 = 1	PDO wird bei Alarmbeginn gesendet
Bit 3 = 1	PDO wird bei Alarmende gesendet

Beispiel: Es ist erwünscht, daß der Sensor mit „Emergency Message“ alarmiert und bei Alarmende ein PDO sendet (Alarm-Aktions-Nr. = Bit 0 + Bit 3 = 09h).

Schreibzugriff des Servers, Modul-ID = 16d

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1552	8	22h	09h	65h	01	09h	00	00	00

Als Antwort sendet der CAN-Sensor:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	60h	09h	65h	01	09h	00	00	00

Sollte die gesendete Alarm-Aktions-Nr. ungültig sein, werden die Änderungen nicht übernommen und im Byte 5 kommt der vorherige Wert zurück.

3.3.6 Abfrage des Alarm-Status (Index 6600h)

Der Index 6600h ist ein Index mit Nur-Lese-Zugriff. Er gibt den Alarm-Status an. Der Index ist wie folgt aufgebaut:

Index	Sub-Index	Parameter	Access
6600h	0	Anzahl der unterstützten Subindizes	lesen
	1	Alarm-Status	lesen

Der Alarm-Status (Byte 5) selbst hat dabei folgende Struktur:

Alarm-Status	Bedeutung
0	kein Alarm-Zustand
1	Alarm ist aktiv

Bei einem Lesezugriff auf den Index 6600h erhält man folgende Antwort:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	00h	66h	01	01	00	00	00

...wenn sich der CAN-Sensor im Alarmzustand befindet, was durch das SDO 6508h spezifiziert ist.

3.3.7 Abfrage der AD-Werte (Index 7100h)

Der Index 7100h ist ein Index mit Nur-Lese-Zugriff. Er gibt den reinen Digitwert aus, den der Digitalsensor liefert und welcher nicht skaliert ist. Der Index hat folgenden Aufbau:

Index	Sub-Index	Parameter	Access
7100h	0	Anzahl der unterstützten Subindizes	lesen
	1	AD-Wert	lesen

Bei einem Lesezugriff auf den Index 7100h erhält man folgende Antwort:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	00h	71h	01	ADLo	ADHigh	00	00

Byte 5 + 6 geben den Meßwert in unskaliertes Form aus. Ein Abfragen des AD-Wertes ist auch über PDOs möglich (siehe "PDO-Kommunikation").

3.3.8 Offset-Verschiebung (Index 7124h)

Der Index 7124h ist ein Index mit Lese-Schreib-Zugriff. Über dieses Objekt ist es möglich, eine Offsetverschiebung (Nullabgleich) der Meßgröße des CAN-Sensors zu realisieren. Der Index hat folgenden Aufbau:

Index	Sub-Index	Parameter	Access
7124h	0	Anzahl der unterstützten Subindizes	lesen
	1	Offsetwert Temperatur	lesen/ schreiben

Bei einem Lesezugriff auf den Index 7124h, Index 1 erhält man folgende Antwort:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	24h	71h	01	Byte1h	Byte2h	00	00

Byte 5 und 6 geben den eingetragenen Offsetwert als vorzeichenbehafteten 16Bit-Integerwert (Zweierkomplement) aus, wobei Byte1h LSB und Byte2h MSB ist. Zu beachten ist, daß der Offsetwert mit 10 multipliziert einzugeben ist (eine Nachkommastelle).

Beispiel: Bei einer positiven Verschiebung des Temperaturwertes um 0,6 K ergäbe sich ein Offset von +6 (Byte1h = 06h, Byte2h = 00h).

3.3.9 Abfrage des Prozeßwertes (Index 7130h)

Der Index 7130h ist ein Index mit Nur-Lese-Zugriff. Er gibt den physischen Prozeßwert, also den Temperaturwert aus. Der Index hat folgenden Aufbau:

Index	Sub-Index	Parameter	Access
7130h	0	Anzahl der unterstützten Subindizes	lesen
	1	Prozeßwert	lesen

Bei einem Lesezugriff auf den Index 7130h erhält man folgende Antwort:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	30h	71h	01	Byte1h	Byte2h	00	00

Byte 5 + 6 geben den gemessenen Temperaturwert als 16Bit-Integerwert aus, wobei Byte1h LSB und Byte2h MSB ist. Die Anzahl der Dezimalstellen ist durch das Objekt 6132h definiert und beträgt 1. Beispielsweise wird bei einem Temperaturwert von 35,0 °C ein Wert von 350d gelesen: Byte1h = 5Eh, Byte2h = 01h. Byte3h und Byte4h sind immer null.

Negative Temperaturwerte werden als 32bit-Zweierkomplement dargestellt.

3.3.10 Abfrage des Alarm-Input-Wertes (Index 7500h)

Der Index 7500h ist ein Index mit Nur-Lese-Zugriff. Über dieses Objekt wird der Prozeßwert, also der Temperaturwert gelesen, der als Eingangsparameter für die Alarm-Routine gilt. Der Index hat folgenden Aufbau:

Index	Sub-Index	Parameter	Access
7500h	0	Anzahl der unterstützten Subindizes	lesen
	1	Prozeßwert	lesen

Bei einem Lesezugriff auf den Index 7500h erhält man folgende Antwort:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	00h	75h	01	Byte1h	Byte2h	00	00

Das Ergebnis ist identisch mit der Antwort auf den Index 7130h.

3.3.11 Alarm-Schwelle (Index 750Ah)

Der Index 750Ah ist ein Index mit Lese-Schreib-Zugriff. Über dieses Objekt wird der Alarm-Wert (als Temperatur-Wert) eingetragen. Der Index hat folgenden Aufbau:

Index	Sub-Index	Parameter	Access
750Ah	0	Anzahl der unterstützten Subindizes	lesen
	1	Alarm-Schwelle	lesen/ schreiben

Bei einem Lesezugriff auf den Index 750Ah erhält man folgende Antwort:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	0Ah	75h	01	Byte1h	Byte2h	00	00

Byte 5 + 6 geben den eingetragenen Temperaturwert als 16Bit-Integerwert aus, wobei Byte1h LSB und Byte2h MSB ist. Die Anzahl der Dezimalstellen ist durch das Objekt 6132h definiert und beträgt 1. Beispielsweise wird bei einem Temperaturwert von -10°C ein Wert von -100d gelesen: Byte1h = 9Ch, Byte2h = FFh (Negative Temperaturwerte werden als 16Bit-Zweierkomplement dargestellt).

3.3.12 Alarm-Hysteresis (Index 750Bh)

Der Index 750Bh ist ein Index mit Lese-Schreib-Zugriff. Über dieses Objekt wird der Hysteresis-Wert (als Temperatur-Wert) eingetragen, d.h. der Abstand der oberen Alarmschwelle von der unteren Alarmschwelle. Dieser Wert hat natürlich nur dann Bedeutung, wenn mit einem Alarmfenster gearbeitet wird (siehe „Alarm-Typ“). Der Index hat folgenden Aufbau:

Index	Sub-Index	Parameter	Access
750Bh	0	Anzahl der unterstützten Subindizes	Lesen
	1	Hysteresis-Wert	lesen/ schreiben

Bei einem Lesezugriff auf den Index 750Bh erhält man folgende Antwort:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	0Bh	75h	01	Byte1h	Byte2h	00	00

Byte 5 + 6 geben den eingetragenen Temperaturwert als 16Bit-Integerwert aus, wobei Byte1h LSB und Byte2h MSB ist. Die Anzahl der Dezimalstellen ist durch das Objekt 6132h definiert und beträgt 1. Beispielsweise wird bei einer Hysterese von 20 K ein Wert von 200d gelesen: Byte1h = C8h, Byte2h = 00h.

3.4 SDO-Fehlermeldungen

Bei fehlerhaften Zugriffen auf SD-Objekte erhält man eine Fehlermeldung als Antwort. Eine Fehlermeldung hat immer folgenden Aufbau:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
	8	80h	Index		Sub-Index	Additional Code		Error Code	Error Class

Der ID der Botschaft, sowie der Index und Sub-Index beziehen sich auf den ID, auf welchen der fehlerhafte Zugriff stattgefunden hat.

Die Fehlermeldungen können folgende Inhalte aufweisen:

Additional Code	Error Code	Error Class	Bedeutung
01h 00h	04h	05h	Server-Kommando ungültig oder unbekannt (weder Schreiben noch Lesen)
00h 00h	01h	06h	Lesezugriff auf ein Nur-Schreib-Objekt
01h 00h	01h	06h	Schreibzugriff auf ein Nur-Lese-Objekt
11h 00h	09h	06h	Sub-Index existiert nicht
00h 00h	02h	06h	Objekt existiert nicht
00h 00h	00h	08h	Anderer Fehler (Codewort falsch)

4 PDO-Kommunikation

Über die Kommunikation mittels PDOs (Process Data Objects) ist es möglich, bestimmte Werte des CAN-Sensors auf einfache und schnelle Weise abzufragen. Im CAN-Sensor wird ein festes PDO-Mapping realisiert, d.h. der Aufbau der PDOs liegt fest.

Die Anforderung der PDOs erfolgt über eine Synchronisations-Message. Es kann jedoch auch ein automatisches Senden in einem frei wählbarem Zeitraster eingestellt werden.

Das Senden der PDOs ist dem CAN-Sensor nur im Operational-Modus möglich.

4.1 Einstellung Sende-PDO

Index 1800h

Über den Index 1800h werden die Einstellungen vorgenommen, um mit den Sende-PDOs arbeiten zu können.

Bei dem CAN-Sensor wird ein festes PDO-Mapping realisiert, d.h. ein PDO enthält den skalierten Meßwert (Temperaturwert), den AD-Wert und den Alarm-Status.

Der Index hat folgenden Aufbau:

Index	Sub-Index	Parameter	Access
1800h	0	Anzahl der unterstützten Subindizes	lesen
	1	COB-ID-PDO-Message	lesen
	2	Art der Sendung	lesen/ schreiben
	3	reserviert	
	4	reserviert	
	5	Timer (16 Bit)	lesen/ schreiben

Subindex 1 kann nur gelesen werden und enthält den Identifier der PDO-Message. Dieser ist unveränderlich und beträgt 180h + Modul-ID.

Über den Subindex 2 kann die Art der Sendung eingestellt werden. Der CAN-Sensor unterstützt folgende Sendungs-Typen:

Sendungstyp	Beschreibung
1 – 240d	zyklisch synchron. Der CAN-Sensor reagiert auf jede n-te SYNC-Botschaft (mit n = 1 .. 240)
254 d	Der CAN-Sensor sendet selbständig nach x ms eine PDO. Der Wert für x wird unter Subindex 5 eingetragen.

Wird der Sendungstyp 254 eingestellt und ist unter Subindex 05 ein Wert >4ms eingetragen, so sendet der CAN-Sensor selbständig das PDO, sobald er in den Operational Modus gesetzt wurde. Über den Subindex 05 wird ein Timer in ms geladen, über welchen das PDO automatisch gesendet wird. Es werden vom CAN-Sensor nur Zeiten größer oder gleich 5ms unterstützt.

4.2 PDO-Mapping

Index 1A00h

Über den Index 1A00h können die PDO-Mapping-Parameter ausgelesen werden. Unter diesem Index sind die SDO-Index, Subindex und der Variablen-Typ für alle Daten gespeichert, die in einer PDO-Sendung eingetragen sind.

Bei dem CAN-Sensor wird ein festes PDO-Mapping realisiert, so daß der Index 1A00h Nur-Lese-Zugriff hat. Der Index hat folgenden Aufbau:

Index	Sub-Index	Parameter	Access
1A00h	0	Anzahl der unterstützten Sub-Indizes	lesen
	1	„Map“-Angaben für die skalierten Meßwerte	lesen
	2	„Map“-Angaben für die AD-Werte	lesen
	3	„Map“-Angaben für den „Alarm-Status“	lesen
	4	„Map“-Angaben für „Fehler-Register“	lesen

Beispiel:

Bei einem Lesezugriff auf den Index 1A00h und den Subindex 1 erhält man folgende Antwort:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	42h	00h	1Ah	01	10h	01h	30h	71h

Byte 5 gibt die Länge der Daten, die das SDO-Objekt liefert. (10h = 16d = 16 Bit)

Byte 6 zeigt den Sub-Index des SDO-Objektes, welches die gleichen Daten wie das PDO liefert. Byte 7 und 8 geben den Index des SDO-Objektes an, welches die gleichen Daten wie das PDO liefert.

4.3 PDO-Sendung

Die PDO-Sendung des CAN-Temperatursensor (Modul-ID = 10h) hat folgendes Aussehen:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
01A0h	8	F2h	00h	00h	00h	F8h	02h	00h	00h

Byte 1 (LSB) + Byte 2 (MSB) bilden den Meßwert (Temperaturwert): 00F2h=242d. Dabei muß berücksichtigt werden, daß in SDO-Objekt 6132h (Anzahl der Nachkommastellen) der Wert „1“ fest eingetragen ist. Das ergibt schließlich den Temperaturwert von 24,2 °C.

Byte 3 und Byte 4 sind immer null.

Byte 5 + Byte 6 bilden den reinen AD-Wert (02F8h = 760d), aus welchem der Temperaturwert berechnet wird.

Byte 7 zeigt den „Alarm-Status“ an (SDO-Objekt 6600h).

Byte 8 beinhaltet das „Fehler-Register“ (SDO-Objekt 1001h).

4.4 Start im Operational Mode

Normalerweise startet der Sensor mit einer Boot-Up Message und geht dann in den Pre-Operational Mode. Um PDO's zu versenden, muss der Sensor mit einem "Start Node" Befehl in den Operational Mode versetzt werden. Um sofort in den Operational Mode zu starten, gibt es untenstehenden Befehl. Diese Einstellungen werden bei Ausführung sofort in den EEPROM geschrieben. Den aktuellen Zustand ändert dieses Kommando aber nicht.

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1424	8	22h	00h	18h	03h	01h	00h	6Fh	6Bh
1424	8	22h	00h	18h	03h	00h	00h	6Fh	6Bh

5 Emergency Messages

Emergency Messages werden im Fehlerfall vom CAN-Sensor selbständig gesendet. Es ist hierbei auf den Unterschied zwischen SDO-Fehlermeldungen bei einem fehlerhaften Zugriff auf ein SDO-Objekt und den „echten“ Fehlermeldungen als Emergency Message zu achten.

Emergency Messages werden automatisch **einmalig** beim Auftreten der unten genannten Fehler gesendet. Die Sendung wird nicht wiederholt, auch wenn der Fehler längere Zeit anliegt. Die Sendung erfolgt zwangsweise, lediglich die Sendung der Emergency Message bei Alarm läßt sich mittels SDO 6509h unterbinden (Bit 0 = 0 setzen).

Verschwundet ein Fehler, so wird ebenfalls eine Message mit Fehlercode 00h 00h gesendet. Das gleiche geschieht, wenn alle Fehler beseitigt sind.

Der Identifier der Emergency Message berechnet sich aus dem Wert 128d + Modul-ID.

Eine Emergency-Message hat folgenden Aufbau:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
144	8	Error Code		Fehler-Register	Manufacturer Specific Error Field				

Es werden folgende Error Codes unterstützt:

Error Code	Bedeutung	Bit im Fehlerregister
00h 00h	Fehler beseitigt	
11h 00h	Generic Error (ADC)	Bit 0 gesetzt
50h 00h	EEPROM-Error	Bit 1 gesetzt
81h 00h	Communication Error	Bit 4 gesetzt
81h 30h	Life Guarding Event	Bit 4 gesetzt
FFh 00h	Alarm Error	Bit 5 gesetzt

Die genaue Beschreibung der Fehler findet man unter SDO 1001h (Fehler-Register).

Im Error-Code werden nur die aktuell hinzugekommenen Fehler angezeigt, bzw. das Verschwinden eines Fehlers (Error-Code 00h 00h). Die noch anliegenden Fehler können dem Fehler-Register (Byte 3) entnommen werden (siehe hierzu Index 1001h Fehler-Register).

Das Manufacturer Specific Error Field ist zur Zeit noch ungenutzt, d.h. Byte 4 bis Byte 8 sind Null.

6 Node Guarding

Unter Node Guarding versteht man die zyklische Knotenüberwachung durch einen NMT-Master. Wichtig ist dies bei Knoten, welche nicht periodisch PDOs senden. Sollten diese ausfallen, so wird dies ohne Node Guarding nicht oder zu spät erkannt, was die gesamte Systemsicherheit gefährden kann.

Beim Node Guarding pollt der NMT-Master jeden Knoten in einem bestimmten zeitlichen Abstand mit einem knotenspezifischen Remote-Transmission-Request-Telegramm (ID ist 1792d plus Modul-ID). Der Knoten beantwortet diese Anfrage durch Senden seines Kommunikationsstatus (Pre-Operational, Operational, Stopped Mode). Erfolgt diese Antwort nicht innerhalb einer bestimmten Zeit oder stimmt der Kommunikationsstatus nicht mit dem beim NMT-Master gespeicherten überein, so führt der Master ein Node-Guarding-Event aus.

Umgekehrt „wartet“ der Knoten auf die Telegramme vom Master, nachdem er erstmalig eines empfangen hat. Dabei muß der Abstand der Remote-Transmission-Request-Telegramme kleiner als die Node Life Time sein. Die Node Life Time errechnet sich durch Multiplikation der Guard Time (SDO 100Ch) mit dem Life-Time-Factor (SDO 100Dh). Wird diese Node Life Time überschritten, wird im Knoten ein Life Guarding Event ausgelöst. Konkret bedeutet das beim Fluid.iO CAN-Sensor, daß eine Emergency-Message mit dem Fehlercode 81h 30h gesendet wird und das Bit 4 im Fehlerregister gesetzt wird, bis wieder ein Telegramm vom NMT-Master eingegangen ist. Node Guarding läßt sich deaktivieren, indem entweder SDO 100Ch oder SDO 100Dh mit einer Null beschrieben wird.

7 LSS-Slave-Funktionen

Der Layer Setting Service (LSS) dient dazu, bei einem CANopen-Gerät über das CAN-Netz unter anderem den Modul-ID und die Bit-Timing-Parameter (Baudrate) zu ändern. Hierzu wird eine Master-Slave-Struktur angewandt: Es existiert im Netz ein LSS-Master und in den CANopen-Geräten die LSS-Slave-Software. Es werden bestimmte Identifier reserviert: Der LSS-Master nutzt die COB-ID 2021d, die LSS-Slaves antworten mit COB-ID 2020d.

Es existieren folgende Dienste:

Switch mode

LSS-Master gibt Umschaltbefehl, durch den der LSS-Slave vom Operation Mode (normale Betriebsart) in den Configuration Mode schaltet

Switch mode global:

Wechsel aller CANopen-Geräte des Netzes in den Configuration Mode ohne weitere Prüfung.

Switch mode selective:

Schaltet nur einen Knoten im Netzwerk in den Configuration Mode. Dazu benötigt der Master die LSS-Adresse des Slaves, welche mit dem Identify-Object 1018h korrespondiert.

Configure Node-ID

Flüchtige Änderung der Knotennummer des CANopen-Gerätes. Die Knotennummer muß zwischen 1 und 127 liegen und nur ein Knoten darf sich im Configuration Mode befinden.

Configure Bit Timing Parameters

Flüchtige Änderung der Baudrate, welche in einer Baudratentabelle bereitgestellt ist. Auch hier sollte sich nur ein Knoten im Configuration Mode befinden. Die Änderung wird hierbei nur vorbereitet, jedoch noch nicht ausgeführt.

Activate Bit Timing Parameters

Mit diesem Kommando werden die Knoten, welche sich im Configuration Mode befinden, aufgefordert, die geänderte Baudrate zu aktivieren. Dies muß für alle Knoten eines Netzwerkes gleichzeitig geschehen. Es wird vom Master eine Delay-Time übergeben, welche zweimal wirksam wird: Slave empfängt Kommando, wartet Delay-Time ab, aktiviert die neue Baudrate und wartet wiederum die Delay-Time ab, ehe er senden darf. Die Delay-Time muß so gewählt werden, daß der langsamste Knoten die Baudraten-Umschaltung in der gegebenen Zeit vornehmen kann. Damit soll verhindert werden, daß zu einem bestimmten Zeitpunkt Knoten mit unterschiedlichen Baudraten im Netz sind

Store Configuration

Nichtflüchtige Speicherung der Node-ID und Baudrate im EEPROM des Knotens, damit diese auch nach einem Neustart wirksam werden können.

7.1 Switch Mode Global

Schalten aller Slaves in Configuration Mode oder zurück in Operation Mode

LSS-Master sendet:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2021	8	04h	X	00	00	00	00	00	00

X = 0: Umschaltung in Operation Mode, 1 = 1: Umschaltung in Configuration Mode

Byte 3...8 sind reserviert

Die LSS-Slaves senden keine Bestätigung

7.2 Switch Mode Selective

Schalten eines Slaves in Configuration Mode

LSS-Master sendet:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2021	8	40h	Vendor-ID	Vendor-ID	Vendor-ID	Vendor-ID	00	00	00

Byte 6...8 sind reserviert

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2021	8	41h	Product Code	Product Code	Product Code	Product Code	00	00	00

Byte 6...8 sind reserviert

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2021	8	42h	Revis. Nr.	Revis. Nr.	Revis. Nr.	Revis. Nr.	00	00	00

Byte 6...8 sind reserviert

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2021	8	43h	Serial Nr.	Serial Nr.	Serial Nr.	Serial Nr.	00	00	00

Byte 6...8 sind reserviert

LSS-Slave antwortet:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2020	8	44h	X	00	00	00	00	00	00

X = 0: Knoten noch in Operation Mode, x = 1: Umschaltung in Configuration Mode ist erfolgt

Byte 3...8 sind reserviert

7.3 Configure Node ID

Ändern der Knotennummer eines Slaves

LSS-Master sendet:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
----	-----	--------	--------	--------	--------	--------	--------	--------	--------

2021	8	11h	x	00	00	00	00	00	00
------	---	-----	---	----	----	----	----	----	----

X = neue Node-ID

Byte 3...8 sind reserviert

LSS-Slave antwortet:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2020	8	11h	Error-Code	Spec-Error	00	00	00	00	00

Byte 4...8 sind reserviert

Error-Code

0: o.k.

1: Node-ID out of range (1-127)

255: spezieller Fehler, der in Byte 3 behandelt wird (nicht implementiert)

7.4 Configure Bit Timing Parameters

Ändern der Baudrate eines Slaves

LSS-Master sendet:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2021	8	13h	Table Selector	Table Index	00	00	00	00	00

Byte 4...8 sind reserviert

Table Selector: immer 0 für Standard CiA-Tabelle

Table Index	Baudrate	unterstützt
0	1000 kBaud	nein
1	800 kBaud	nein
2	500 kBaud	ja
3	250 kBaud	ja
4	125 kBaud	ja
5	100 kBaud	ja
6	50 kBaud	ja
7	20 kBaud	ja
8	10 kBaud	ja

LSS-Slave antwortet:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2020	8	13h	Error-Code	Spec-Error	00	00	00	00	00

Byte 4...8 sind reserviert

Error-Code

0: o.k.

1: Bit Timing wird nicht unterstützt

255: spezieller Fehler, der in Byte 3 behandelt wird (nicht implementiert)

7.5 Activate Bit Timing Parameters**Aktivierung der neuen Baudrate bei allen Slaves in Configuration Mode**

LSS-Master sendet:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2021	8	15h	Switch Delay Low	Switch delay High	00	00	00	00	00

Byte 4...8 sind reserviert

Switch delay: Verzögerungszeit in ms

Die LSS-Slaves senden keine Bestätigung

7.6 Store Configuration Protocol**Nichtflüchtige Speicherung der geänderten Daten**

LSS-Master sendet:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2021	8	17h	00	00	00	00	00	00	00

Byte 2...8 sind reserviert

LSS-Slave antwortet:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
2020	8	17h	Error- Code	Spec- Error	00	00	00	00	00

Byte 4...8 sind reserviert

Error-Code

0: o.k.

2: Speicherfehler

255: spezieller Fehler, der in Byte 3 behandelt wird (nicht implementiert)