



KS 800 Multi-Temperaturregler

KS 8000

CANopen

KS

Schnittstellenprotokoll

CANopen

9499 040 49518

gültig ab: 8389

©PMA Prozeß- und Maschinen-Automation GmbH. Printed in Germany
Alle Rechte vorbehalten. Ohne vorherige schriftliche
Genehmigung ist der Nachdruck oder die auszugsweise
fotomechanische oder anderweitige Wiedergabe dieses Dokumentes
nicht gestattet.

Dies ist eine Publikation der PMA Prozeß- und Maschinen-Automation GmbH

Bei Änderungen erfolgt keine gesonderte Mitteilung.

PMA Prozeß- und Maschinen-Automation GmbH
P.O. Box 31 02 29
D 34058 Kassel
Germany

Einschränkung der Gewährleistung:

Es wird keine Gewähr für die vollständige Richtigkeit des Inhalts übernommen, da sich trotz aller
Sorgfalt Fehler nie ganz vermeiden lassen. Für Hinweise sind wir jederzeit dankbar.

Inhaltsverzeichnis

1	Einleitung	5
2	Service-Daten-Objekte	7
3	Prozeß-Daten-Objekte	9
	3.1 Information Record (Sende-PDO vom KS 800):	10
	3.2 Control Record (Empfangs-PDO des KS 800):	11
	3.3 PDO Übertragungsverhalten	12
	3.3.1 Asynchrones Sende PDO	12
	3.3.2 Synchrones Sende PDO	13
	3.3.3 Asynchrones Empfangs PDO	13
	3.3.4 Synchrones Empfangs PDO	14
4	OP-Mode / Pre-OP-Mode	15
5	SYNC	16
6	Nodeguarding / Lifeguarding	17
7	LMT-Dienste	18
	7.1 Switch Mode Services	18
	7.1.1 Switch Mode Global	18
	7.1.2 Switch Mode Selective	18
	7.2 Configuration Services	19
	7.2.1 Configure Module ID	19
	7.2.2 Configure Bit Timing	19
	7.3 Inquire LMT Address	20
	7.3.1 Inquire Manufacturer Name	20
	7.3.2 Inquire Product Name	20
	7.3.3 Inquire Serial Number	21
	7.3.4 Serial Number Format	21
	7.4 Identification Services	21
	7.4.1 LMT Identify Remote Slaves	21
	7.4.2 LMT Identify Slave	22
	7.5 Herstellerspezifische LMT-Dienste	22
	7.6 Inquire Node-ID	22
	7.6.1 Configure Module-ID	23
	7.6.2 Configure Bit Timing	23
	7.6.3 Inquire Bit Timing	24
8	Objektverzeichnis	25
	8.1 Übersicht Objektverzeichnis herstellerspezifischer Teil	25
9	Konfigurierung und Parametrisierung über den CAN-Bus	36
	9.1 KS 800 über CAN_Bus Schnittstelle konfigurieren.	36
	9.1.1 Abspeichern des nichtflüchtigen Sollwertes	36
	9.2 Plausibilitätsprüfung der Daten bei der Übertragung.	37
	9.3 Zuordnungstabelle Firmwareversion, Bedienversion, Serien-Nummer und E-Tool - Version	37
	9.4 Reihenfolge der Daten zur Konfigurierung und Parametrisierung	38

10	Applikationsbeispiele	40
10.1	Definition der Beispiele	40
10.2	Beispiel 1 (Heizen/Kühlen-Regler)	40
10.3	Beispiel 2 (H/K-Regler einstellen und Selbstoptimierung aktivieren)	41
11	CAN Physical Layer	43
11.1	ISO 11898-2 Knoten	43
11.2	Baudraten und Buslängen	43
11.3	Praktische Buslängen	44
11.4	Kabel-Parameter	45

1 Einleitung

Allgemeines:

Der KS 800 besitzt eine CAN-Schnittstelle, über die dem Regler alle zum Betrieb notwendigen Daten (Parameter- und Konfigurationsdaten) gesendet werden können. Ebenso kann ein entsprechender Master (PC oder Steuerung) alle Prozeßdaten abrufen.

Gemäß "CANopen" sind die Zugriffe in SDO (Service Data Objects) und PDO (Process Data Objects) gegliedert. Dabei sind SDOs zur Konfiguration und Parametrisierung von Busteilnehmern vorgesehen und PDOs für die üblichen Betriebswerte.

Betrieb des KS 800 am CAN Bus:

Nach Aufstart des KS 800 befindet sich das Gerät im "Pre Operational Mode". Man kann über die CAN-Schnittstelle mittels SDOs kommunizieren. Nach der Bus- und Gerätekonfiguration muß der Master den KS 800 in den "Operational Mode" setzen (NMT Start) um auch PDOs nutzen zu können.

Wie PDOs genutzt werden, muß vorab konfiguriert werden. Man unterscheidet "asynchrone PDOs" und "synchrone PDOs". Asynchrone PDOs werden bei Statusänderungen wie Toleranzbandverletzungen, Fühlerbruch oder ähnlichem gesendet. Synchrone PDOs werden auf Anforderung durch den Master gesendet. Dazu sendet der Master einen SYNC Frame. In der Default Einstellung des KS 800 sind asynchrone PDOs freigegeben. (Details siehe: Prozeß-Daten-Objekte)

Die im KS 800 implementierten Kommunikationsdienste basieren auf dem "CANopen Communication Profile for Industrial Systems" (CIA Draft Standard 301). Die Funktionalität entspricht dem "Minimum Capability Device" mit "Predefined Master/Slave Connection Set", ergänzt um "LMT-Dienste" jedoch ohne "Time-Stamp-" und "Store-Parameter_" Objekte.

Typ	Adresse (dez.)	Adresse (hex.)
NMT_Commands	0	0x0
SYNC	128	0x80
LMT-RX	2021	0x7E5
LMT-TX	2020	0x7E4
EMERGENCY	128 + NodeID ^{*)}	0x80 + NodeID ^{*)}
PDO1-TX	384 + NodeID ^{*)}	0x180 + NodeID ^{*)}
PDO1-RX	512 + NodeID ^{*)}	0x200 + NodeID ^{*)}
PDO2-TX	640 + NodeID ^{*)}	0x280 + NodeID ^{*)}
PDO2_RX	768 + NodeID ^{*)}	0x300 + NodeID ^{*)}
SDO-TX	1408 + NodeID ^{*)}	0x580 + NodeID ^{*)}
SDO-RX	1536 + NodeID ^{*)}	0x600 + NodeID ^{*)}
NODE_GUARD	1769 + NodeID ^{*)}	0x6E0 + NodeID ^{*)} ^{**)}

^{*)} NodeID-Bereich von 1 ... 127

^{**)} KS 800 entspricht nicht 301-Standard

Netzwerkmanagement:

Zur Überwachung der Kommunikation bietet der KS 800 die Möglichkeit des "Nodeguardings". Damit ist es dem Master möglich zu erkennen, ob ein KS 800 am Bus angeschlossen ist und ob die Kommunikation zum Gerät sichergestellt ist.

Die zur Identifikation eines CAN Teilnehmers nötige Node-ID (Knoten-Adresse) sowie die CAN-Baudrate werden mit Hilfe des KS 800 Engineering Tools über die Diagnoseschnittstelle eingestellt.

Eine Einstellung über den CAN-Bus ist ebenfalls möglich (siehe Kap. 7 Objektverzeichnis, Index 200E/200F)

2 Service-Daten-Objekte

Ein SDO ist logisch festgelegt auf genau zwei Partner - 1 Master und 1 Slave. Es ist nicht möglich, von zwei unterschiedlichen Mastern gleichzeitig auf den KS 800 zuzugreifen, da für "tx" als auch für "rx" nur ein SDO-Kanal zur Verfügung steht.

Mittels SDOs können maximal 4 Byte Nutzdaten übertragen werden. Die restlichen 4 Byte werden wie folgt genutzt:

- 1 Byte für den Command: Art der Übertragung
- 2 Byte für den Index: Objektkennung
(z.B. 0x2213 für Wvol - siehe Objektverzeichn.)
- 1 Byte für den Subindex: Kanalnummer

Im folgenden wird jeweils ein Beispiel eines SDO-Datenframes für Lesen und für Schreiben gegeben.

Beispiel 1. Ein Master sendet einen Sollwert an einen KS 800 mit Knotenadresse 4:

Der Aufbau der Nachricht ist wie folgt:

- ⇒ COB_ID = SDO an Knoten 4 = $0x600 + 4$ = 0x604
- ⇒ CMD = Schreibzugriff = 0x2B = 0x2B
- ⇒ INDEX = Sollwert = 0x2213 = 0x2213
- ⇒ SUBINDEX = z.B. Regler 1 = 01 = 0x01
- ⇒ WERT = 30,0°C *) = 300 *) = 0x12C

*) Fixedpoint1- Darstellung, d.h 1 feste Nachkommastelle.

	COB-ID	LEN	8 data bytes			
			Cmd	Index	Sub.	Data
<u>Anforderung:</u> (Master)	0x604	0x08	0x2B	0x13 0x22	0x01	0x2C 0x01 0x00 0x00
<u>Antwort:</u> (KS 800)	0x584	0x08	0x60	0x13 0x22	0x01	0x00 0x00 0x00 0x00

Beispiel 2. Ein Master liest einen Istwert von einem KS 800 mit Knotenadresse 2:

Der Aufbau der Nachricht ist wie folgt:

- ⇒ COB_ID = SDO an Knoten 2 = 0x600 + 2 = 0x602
- ⇒ CMD = Lesezugriff = 0x40 = 0x40
- ⇒ INDEX = Istwert = 0x2202 = 0x2202
- ⇒ SUBINDEX = z.B. Regler 3 = 03 = 0x03

	COB-ID	LEN	8 data bytes						
			Cmd	Index	Sub.	Data			
<u>Anforderung:</u> <u>(Master)</u>	0x602	0x08	0x40	0x02 0x22	0x03	0x00	0x00	0x00	0x00
<u>Antwort:</u> <u>(KS 800)</u>	0x582	0x08	0x4B	0x02 0x22	0x03	0xFA	0x00	0x00	0x00

- ⇒ WERT = 0x00FA = 250 = 25,0°C

Mit Hilfe der SDOs können Konfiguration und Parametrisierung des Reglers über Einzelzugriffe auf alle Parameter (siehe Objektverzeichnis) eingestellt werden. Aber auch alle Bediendaten, die üblicherweise über PDOs übertragen werden, können über SDOs angesprochen werden.

3 Prozeß-Daten-Objekte

Mittels SDOs können maximal 4 Byte Nutzdaten zwischen zwei Partnern übertragen werden. Die restlichen 4 Byte werden für den Command, den Index und den SubIndex genutzt. Im Gegensatz dazu können mit einem PDO maximal 8 Byte Nutzdaten übertragen werden. Vom KS 800 werden zwei Sende- und zwei Empfangs-PDOs unterstützt, die über die Konfiguration entweder als "Synchrone PDOs" oder als "Asynchrone PDOs" definiert werden können. Der Datentransfer über PDOs ist zum Beispiel dann sinnvoll, wenn Istwerte und Statuswerte des Reglers abhängig von z.B. Fehlermeldungen (Statusänderungen) zum Master gesendet werden müssen. In diesem Fall wird man asynchrone PDOs verwenden, die bei Eintreten des Events auf dem Regler automatisch zum Master gesendet werden. Ein anderes Beispiel wäre die zyklische Anforderung von Istwerten durch eine Bedieneinheit. In diesem Fall kann der KS 800 PDOs synchronisiert zu einem Messtakt (SYNC-EVENT) senden.

PDOs sind, im Gegensatz zu SDOs, grundsätzlich **nicht** bestätigte Dienste. Sie können von mehreren Teilnehmern empfangen werden. Dies gilt, wie oben schon erwähnt, nicht für SDOs.

In der Default Konfiguration des KS 800 sind je ein asynchrones Sende- und Empfangs-PDO mit folgenden Identifiern freigeschaltet.

- ⇒ COB-ID des asynchronen Empfangs-PDOs: 0x200 + Node-ID
- ⇒ COB-ID des asynchronen Sende-PDOs: 0x180 + Node-ID

Eine Umkonfiguration der PDOs kann mittels SDO Zugriffen auf folgende Indizes geschehen:

		default	default COB-ID	
⇒	Index 1400:	erstes Empfangs-PDO	sync./disabled	300
⇒	Index 1401:	zweites Empfangs-PDO	async./enabled	200
⇒	Index 1800:	erstes Sende-PDO	sync./disabled	280
⇒	Index 1801:	zweites Sende-PDO	async./disabled	180

Unter Subindex 1 wird der Message-Identifizier (COB-ID) konfiguriert und das PDO freigegeben oder gesperrt.

- ⇒ COB-ID (low byte) data[1]
- ⇒ COB-ID (high byte) data[2]
- ⇒ Freigabe data[4] (0x00 = frei, 0x80 = gesperrt)

Unter Subindex 2 wird der Transfer-Typ konfiguriert

- ⇒ Transfertype data[1] (0x01...0xF0=synch, 0xFF=async)

Im folgenden Abschnitt werden die Dateninhalte der PDOs beschrieben.

3.1 Information Record (Sende-PDO vom KS 800):

Der Information Record ist wie folgt aufgebaut:

Index	Subindex	Field	Data Type
42	0	number of supported entries in the record	unsigned8
	1	channel number	unsigned8
	2	Xeff	FixedPoint1
	3	device status information	unsigned8
	4	channel status information	unsigned16
	5	Ypid	FixedPoint1

Ein Send-PDO vom KS 800 enthält im Detail folgende Daten:

- Kanal → data[0]
- Xeff → data[1, 2]
 - low byte → data[1]
 - high byte → data[2]
- Geräte-Statusinformation: → data[3]
 - Bit 0 ^{*}) - offline[0], online [1]
 - Bit 1 ^{*}) - DO1...12 fail
 - Bit 2 ^{*}) - DO13...16 fail
 - Bit 3 ^{*}) - Heizstrom Kurzschluß
 - Bit 4 - di1 | Signalisieren der "Input State",
 - Bit 5 - di2 | wenn als Input konfiguriert,
 - Bit 6 - di3 | sonst "0".
 - Bit 7 - di4 | Erzeugt keinen asynchr. PDO-Trigger!
- Kanal-Statusinformation: → data[4, 5]
 - Bit 0 ^{*}) - Alarm HH → data[4]
 - Bit 1 ^{*}) - Alarm H
 - Bit 2 ^{*}) - Alarm L
 - Bit 3 ^{*}) - Alarm LL
 - Bit 4 ^{*}) - Alarm Sensor Fail
 - Bit 5 ^{*}) - Alarm Heizstrom
 - Bit 6 ^{*}) - Alarm Leckstrom
 - Bit 7 ^{*}) - Alarm DO_x
 - Bit 8 ^{*}) - W2 aktiv → data[5]
 - Bit 9 ^{*}) - Wint aktiv
 - Bit 10 ^{*}) - Wanfahr aktiv
 - Bit 11 ^{*}) - Optimierung aktiv
 - Bit 12 ^{*}) - Fehler bei Optimierung
 - Bit 13 ^{*}) - Regler A/M
 - Bit 14 ^{*}) - Regler abgeschaltet (coff)
- Ypid: → data [6, 7]
 - low Byte ^{*}) - data [6]
 - high Byte ^{*}) - data [7]

^{*}) Eine Änderung erzeugt einen asynchronen PDO-Event

3.2 Control Record (Empfangs-PDO des KS 800):

Der Control Record ist wie folgt aufgebaut:

Index	Subindex	Field	Data Type
43	0	number of supported entries in the record	unsigned8
	1	channel number	unsigned8
	2	Wvol	FixedPoint1
	3	Yman	FixedPoint1
	4	control byte	unsigned8
	5	update byte	unsigned8

Über Empfangs-PDOs können im Detail folgende Daten übernommen werden:

- Kanal → data[0]
- Sollwert (Wvol) → data[1, 2]
 - low byte data[1]
 - high byte data[2]
- Reglerausgangswert (Yman) → data[3, 4]
 - low byte data[3]
 - high byte data[4]
- Kontrollinformation: → data[5]
 - Bit 0 - A[0] / M [0/1]
 - Bit 1 - COFF [1]
 - Bit 2 - W/W2 [0/1]
 - Bit 3 - Wext/Wint [0/1]
 - Bit 4 - Ostart [1]
- Updateinformation: → data[6]
 - Bit 0 - A/M [0/1]
 - Bit 1 - COFF [1]
 - Bit 2 - W/W2 [0/1]
 - Bit 3 - Wext/Wint [0/1]
 - Bit 4 - Ostart [1]
 - Bit 6 - Yman [1]
 - Bit 7 - Wvol [1]

Über die Updateinformation kann selektiert werden, welche Daten aus dem Control-Record vom KS 800 übernommen werden. Ist eines der Bits gesetzt, werden die zugehörigen Daten auf dem KS 800 aktualisiert.

3.3 PDO Übertragungsverhalten

3.3.1 Asynchrones Sende PDO

Wenn ein PDO als asynchrones PDO konfiguriert ist, wird bei Änderung von Statusinformationen, bzw. Ypid eines Kanals ein PDO gesendet. Die Reihenfolge der Meldungen ergibt sich wie folgt: Die Überprüfung ob gesendet werden muß beginnt mit dem ersten Kanal. Wurde seit dem letzten Senden eine Statusänderung registriert (Flag fSendEvent gesetzt), wird ein PDO erzeugt, gesendet und das Flag zurückgesetzt. Nach erfolgreichem Senden wird der nächste Kanal bearbeitet. Sind beide Sende PDOs als asynchrone PDOs konfiguriert, werden zwei Kanäle gleichzeitig überprüft und gesendet.

Beispiel:

- Kanal 1: fSendEvent = TRUE
→ PDO für Kanal 1 senden
- Kanal 2: fSendEvent = FALSE
- Kanal 3: fSendEvent = TRUE
→ PDO für Kanal 3 senden
- Kanal 4: fSendEvent = FALSE
- Kanal 5: fSendEvent = FALSE
- Kanal 6: fSendEvent = FALSE
- Kanal 7: fSendEvent = TRUE
→ PDO für Kanal 7 senden
- Kanal 8: fSendEvent = FALSE
- Kanal 1: fSendEvent = FALSE

Ein asynchrones PDO wird vom KS 800 immer bei Zustandswechsel eines der Status-Bits oder Ypid-Änderung gesendet. Die vom KS 800 gesendete Nachricht sieht bei Default-Konfiguration wie folgt aus:

Beispiel:

Nachricht vom KS 800 (ASYNCR-PDO):

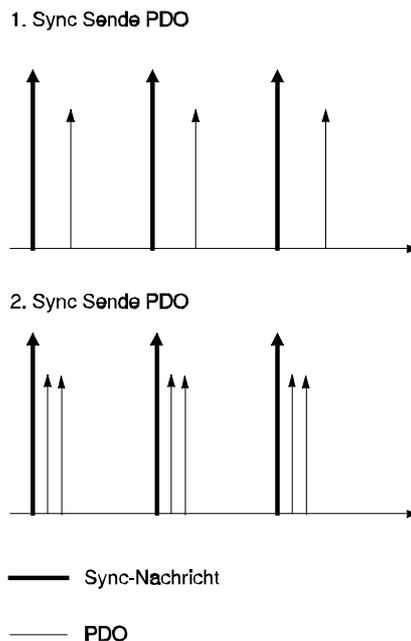
COB-ID	8 data bytes					
	LEN	Channel	Xeff	Dev. st.	Channel st.	Ypid
0x180 +node	8	0x05	0xBC 0x02	0x00	0x00 0x42	0xF4 0x01

- ⇒ Xeff = 0x02BC = 700 = 70,0°C
- ⇒ Device Status = 0x00 = keine Fehler
- ⇒ Channel Status = 0x4200 = Coff = 1, Wint = 1
- ⇒ Ypid = 0x01f4 = 500 = 50%

3.3.2 Synchrones Senden PDO

Der Master sendet zum Refresh von Istwerten SYNC-Impulse. Nach wieviel Sync-Nachrichten der KS 800 reagieren soll, kann vom Master konfiguriert werden. Die Defaulteinstellung ist "1". (siehe Transfertype, Kap. 3 Prozeß-Daten-Objekte)

Im synchronen Betrieb wird nach jeder Sync-Nachricht (wenn Transfertype = 1) ein Kanal, beginnend mit dem ersten, gesendet. Danach werden alle weiteren Kanäle der Reihe nach gesendet. Nach dem letzten Kanal wird beim ersten ein neuer Durchlauf gestartet. Sind beide Senden PDOs als synchrone PDOs konfiguriert, werden zwei Kanäle gleichzeitig übertragen. Das folgende Diagramm erläutert die Zusammenhänge zwischen Sync-Nachricht und PDOs.



3.3.3 Asynchrones Empfangen PDO

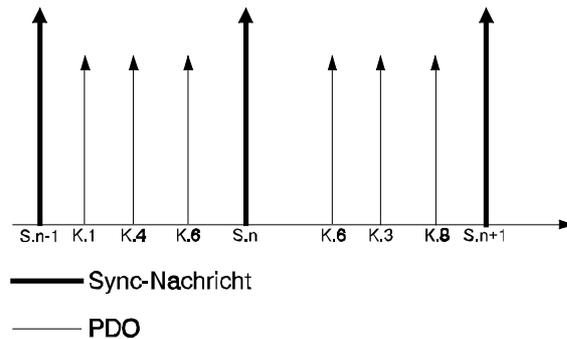
Wenn ein asynchrones PDO empfangen wird, werden die enthaltenen Daten mit dem nächsten Kommunikations Modul-Zyklus an das Reglermodul weitergereicht. Die Daten werden entsprechend der Kanalnummer in den Übergabebereich geschrieben. Dabei wird das Flag fReceived gesetzt, dies signalisiert dem Regler, daß neue Daten für ihn bereitstehen. Das Flag fReceived wird durch den Regler zurückgesetzt.

Achtung: Das Kommunikationsmodul überprüft nicht, ob der Regler die zuletzt empfangenen Daten bereits übernommen hat, wenn es neue Daten an den Regler übergibt. (Abtastrate beachten!) D.h. PDO's können überschrieben werden, ältere Informationen gehen verloren!

3.3.4 Synchrones Empfangs PDO

Wird ein synchrones PDO empfangen, verbleiben die Daten zunächst im Rx-Buffer (CAN-Controller) und werden noch nicht an den Regler übergeben. Bei Empfang der nächsten gültigen Sync-Nachricht werden sie an den Regler übergeben. Dazu wird das Flag fReceived für den entsprechenden Kanal gesetzt. Dies signalisiert dem Regler das neue Daten für ihn bereitstehen. Das Flag fReceived wird durch den Regler zurückgesetzt. Folgendes Diagramm zeigt den Zusammenhang zwischen Sync-Nachricht und Synchronen PDOs.

Achtung: Wird ein weiterer synchroner PDO vor der nächsten Sync-Nachricht empfangen, so wird das zuletzt empfangene PDO an den Regler übergeben. D.h. PDO's können überschrieben werden, ältere Informationen gehen verloren!



Erklärung des Diagramms:

Wenn nur 1 PDO auf sync konfiguriert ist, wird Kanal 6 (S.n), bzw. Kanal 8 (S.n+1) übergeben.

Bei 2 sync PDOs sind es Kanal 4+6, bzw. 3+8. Der Erste der 3er Gruppe geht immer verloren, da der/die Rx-Buffer überschrieben werden.

4 OP-Mode / Pre-OP-Mode

Nach dem Aufstart des CAN-Netzwerkes ist der KS 800 im Pre-OP-Mode. Das heißt, daß er nur über SDOs angesprochen werden kann. Wird der KS 800 in den OP-Mode gesetzt, kann auch über die freigegebenen PDOs kommuniziert werden. Bei Default Konfiguration sind die asynchronen PDOs nach Umschalten in den OP-Mode aktiv, synchrone PDOs müssen, falls erforderlich, über entsprechende SDO-Nachrichten freigegeben werden.

KS 800 in OP-Mode setzen:

		2 data bytes	
COB-ID	LEN	Command	Knoten
0x00	2	0x01	<Nummer> *)

KS 800 in Pre-OP-Mode setzen:

		2 data bytes	
COB-ID	LEN	Command	Knoten
0x00	2	0x80	<Nummer> *)

Wird als Knotenadresse "0" angegeben, werden alle im Netz befindlichen Teilnehmer in den Pre-OP-Mode gesetzt. Die Regler senden keine Antwort auf diese Nachricht.

Weitere NMT (Netzwerk-Management) - Kommandos:

Reset Node:

		2 data bytes	
COB-ID	LEN	Command	Knoten
0x00	2	0x81	<Nummer> *)

Reset Communication:

		2 data bytes	
COB-ID	LEN	Command	Knoten
0x00	2	0x82	<Nummer> *)

*) Knoten-Nummer-Bereich 1 ...127

Achtung: Reset-Node führt zu einem Hardware-Reset des KS 800, d.h. er ist erst nach einigen Sekunden wieder über den CAN-Bus erreichbar. Alle Command-Parameter werden auf die Default-Werte zurückgesetzt.

5 SYNC

Wie bereits beschrieben werden synchrone PDO-Daten mittels SYNC-Nachrichten vom KS 800 ausgewertet oder gesendet. Die entsprechenden KS 800 müssen im OP-Mode sein und synchrone PDOs müssen konfiguriert sein. (Siehe 3. Prozeß-Daten-Objekte)

Eine SYNC-Nachricht ist ein sogenannter "Broadcast" (Nachricht an alle Teilnehmer im Netz) und wird wie folgt vom Master gesendet:

		0 data bytes	
COB-ID	LEN		
0x80	2		

Antwort vom KS 800:

z.B. folgender PDO

		8 data bytes				
COB-ID	LEN	Channel	Xeff	Dev. st.	Channel st.	Ypid
0x80 +node [^])	8	0x05	0x2BC 0x02	0x00	0x00 0x42	0xF4 0x01

- ⇒ Xeff = 0x0BC = 700 = 700°C
- ⇒ Device Status = 0x00 = kein Fehler
- ⇒ Channel Status = 0x4200 = Coff = 1, Wint = 1
- ⇒ Ypid = 0x01F4 = 500 = 50%

Nach einem erneuten SYNC vom Master werden die Daten des nächsten Kanals, in diesem Fall des Kanals 6, übertragen usw. Nach Übertragung des achten Kanals wird wieder bei Kanal 1 begonnen. Sind mehrere KS 800 im Netz, werden pro SYNC-Event von jedem Regler die Daten eines Kanals gesendet (Max. 2 Kanäle wenn 2 sync. PDOs konfiguriert.)

[^]) Der Default-Wert des COB_ID von 0x80 lässt sich mit dem Objekt 1005 ändern.

6 Nodeguarding / Lifeguarding

Mittels "Nodeguarding" testet ein Master zyklisch, ob die Kommunikation zu den im Netz konfigurierten Slaves (z.B. KS 800) noch möglich ist. Bei "Lifeguarding" prüft ein Slave ob er noch regelmäßig von einem Master angesprochen wird.

Über die Parameter "guard time * life time" kann festgelegt werden, in welchen Zeitabständen der Master die Nodeguard-Nachricht aktivieren muß (Timeout für den Master).

Ist "guard time * life time" = "0", wird kein Lifeguarding durchgeführt und der Master ist an kein Zeitraster gebunden. Ist "guard time * life time" definiert und der KS 800 wird nicht im angegebenen Zeitraster abgefragt, fällt der KS 800 vom OP-Mode in den Pre-OP-Mode zurück. Es werden dann keine PDOs mehr übertragen.

Beim Nodeguarding antwortet der KS 800 durch "toggeln" des höchstwertigen Bits in der Antwortnachricht:

- Ist der KS 800 im Op-Mode, antwortet er wie folgt: 0x05 → 0x85 → 0x05 → 0x85 → ...
- Ist der KS 800 im Pre-Op-Mode, antwortet er wie folgt: 0x7F → 0xFF → 0x7F → 0xFF → ...

Node-Guarding Identifier → Objekt 100E Default: 6E0 + node ID

Beispiel:

Nodeguard-Nachricht vom Master:

		0 data bytes	
COB-ID	LEN		
0x6E0 + node	0		

Antwort vom KS 800 (Pre-OP-Mode):

		1 data bytes	
COB-ID	LEN	Answer	
0x6E0 +node	1	0x7F	
0x6E0 +node	1	0x7F	
0x6E0 +node	1	0x7F	

Antwort vom KS 800 (OP-Mode):

		1 data bytes	
COB-ID	LEN	Answer	
0x6E0 +node	1	0x05	
0x6E0 +node	1	0x05	
0x6E0 +Node	1	0x05	

Anmerkung: Der Default-Value für den Node-Guarding-Identifier bleibt bei **0x6E0+node-ID** (nicht konform zum DS301 Version 3.0, aber abwärtskompatibel). Der Wert kann über das Objekt **0x100E** gelesen und verändert werden.

7 LMT-Dienste (CMS-Standard und Erweiterungen)

Es stehen die LMT-Dienste gemäß CiA/DS205 Version 1.1 zur Verfügung. Einschränkungen, bzw. spezielle Erweiterungen sind den nachfolgenden Service-Beschreibungen zu entnehmen. Die zugehörigen COB-ID's sind 2021 (Master => Slave), bzw. 2020 (Slave => Master), wobei der KS 800 jeweils Slave ist.

7.1 Switch Mode Services

7.1.1 Switch Mode Global

M => S

0	1	2	3	4	5	6	7
cs = 04	mode	r	r	r	r	r	r

cs LMT command specifier
mode 0: switches to operation mode
 1: switches to configuration mode
r reserved

7.1.2 Switch Mode Selective

M => S

0	1	2	3	4	5	6	7
cs = 01	m1	m2	m3	m4	m5	m6	m7

0	1	2	3	4	5	6	7
cs = 02	p1	p2	p3	p4	p5	p6	p7

0	1	2	3	4	5	6	7
cs = 02	s1	s2	s3	s4	s5	s6	s7

cs LMT command specifier
m1..7 manufacturer name part of LMT address
p1..7 product name part of LMT address
s1..7 serial number part of LMT address

7.2 Configuration Services

Configure NMT Address besteht aus den Diensten Configure Module ID und Configure Modul Name. Der Service Configure Modul Name wird nicht unterstützt. Ein entsprechender Request liefert error_code: 255 und specific_error_code: 01 (service not supported).

7.2.1 Configure Module ID

M => S

0	1	2	3	4	5	6	7
cs = 17	Mld	r	r	r	r	r	r

S => M

0	1	2	3	4	5	6	7
cs = 17	error code	spec error	r	r	r	r	r

cs LMT command specifier
Mld new module_id to configure
error_code 0: successful
1..254: reserved
255: implementation specific error
specific_error only valid if error_code equal 255
3: Mld out of range
4: command execution error
r reserved

7.2.2 Configure Bit Timing

M => S

0	1	2	3	4	5	6	7
cs = 19	table selector	table index	r	r	r	r	r

S => M

0	1	2	3	4	5	6	7
cs = 19	error code	spec error	r	r	r	r	r

cs LMT command specifier
table_selector 0: standard CiA bit timing table
table_index 0: 1 Mbit
1: 800 kbit
2: 500 kbit
3: 250 kbit
4: 125 kbit
5: 50 kbit
6: 20 kbit
7: 10 kbit
error_code 0: successful
1: bit timing not supported
2..254: reserved
255: implementation specific error
specific_error only valid if error_code equal 255
3: value out of range
4: command execution error
r reserved

Der Service *Activate Bit Timing* wird nicht unterstützt (unconfirmed service !), die Aktivierung erfolgt über einen Reset (siehe Adress-/BR-Schalter für die 3 Reset-Möglichkeiten).

Der Service *Store Configuration* wird nicht unterstützt Ein entsprechender Request liefert error_code: 01 (service not supported).

7.3 Inquire LMT Address

Inquire LMT Address besteht aus den folgenden 3 Diensten:

7.3.1 Inquire Manufacturer Name

M => S

0	1	2	3	4	5	6	7
cs = 36	r	r	r	r	r	r	r

S => M

0	1	2	3	4	5	6	7
cs = 36	m1	m2	m3	m4	m5	m6	m7

cs LMT command specifier
m1..m7 manufacturer name of the module (if m1 is a valid alpha-num character).
If m1 is 255, m2 contains error_code and m3 optional error-reason.
(liefert "Firmen-Name" (PMA))

7.3.2 Inquire Product Name

M => S

0	1	2	3	4	5	6	7
cs = 37	r	r	r	r	r	r	r

S => M

0	1	2	3	4	5	6	7
cs = 37	p1	p2	p3	p4	p5	p6	p7

cs LMT command specifier
p1..p7 product name of the module (if p1 is a valid alpha-num character).
If p1 is 255, p2 contains error_code and p3 optional error-reason.
(liefert "System-Name" (Popen))
r reserved

7.3.3 Inquire Serial Number

M => S

0	1	2	3	4	5	6	7
cs = 38	r	r	r	r	r	r	r

S => M

0	1	2	3	4	5	6	7
cs = 38	s1	s2	s3	s4	s5	s6	s7

cs LMT command specifier
s1..s7 serial number of the module (if s1 is a valid BCD-pair).
If s1 is 255, s2 contains error_code and s3 optional error-reason.
(liefert Geräte-Code-Nummer)
r reserved

7.3.4 Serial Number Format (14 BCD-digit => 7 byte)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	0	m1	m2	m3	m4	c1	c2	c3	c4	c5	c6	c7	c8

m1..m4 FabMonth, Herstellungsmonat in verschlüsselter Schreibweise
c1..c8 FabCount, "unique" Zählnummer, einmalige Vergabe

7.4 Identification Services

7.4.1 LMT Identify Remote Slaves

M => S

0	1	2	3	4	5	6	7
cs = 05	manu- fact. name						

M => S

0	1	2	3	4	5	6	7
cs = 06	product name						

M => S

0	1	2	3	4	5	6	7
cs = 07	serial number low						

M => S

0	1	2	3	4	5	6	7
cs = 08	serial num- ber. high						

cs LMT command specifier
manufact._name manufacturer name part of LMT address
product_name product name part
serial_number_low lower boundary of requested range
serial_number_high higher boundary
('boundaries' sind im Intervall enthalten)

7.4.2 LMT Identify Slave

S => M

0	1	2	3	4	5	6	7
cs = 09	r	r	r	r	r	r	r

cs LMT command specifier
r reserved

Hinweis: Bei mehreren Geräten am Bus kann es hier zu Problem, speziell dem Verlust von Frames, kommen ! (bedingt durch die Anzahl der gleichzeitig antwortenden Teilnehmer mit identischen Frames)

7.5 Herstellerspezifische LMT-Dienste

Der Service *Activate Bit Timing* (cs = 131) wird nicht unterstützt (siehe auch gleichnamigen Std.-Service).

7.6 Inquire Node-ID

M => S

0	1	2	3	4	5	6	7
cs = 128	CAN	r	r	r	r	r	r

S => M

0	1	2	3	4	5	6	7
cs = 128	error code	spec error	Node ID	r	r	r	r

cs LMT command specifier
CAN number of CAN-controller (must be 0)
error_code 0: successful
 1..254: reserved
 255: implementation specific error
specific_error only valid if error_code equal 255
 6: illegal CAN-controller selected
Node ID module Node ID

7.6.1 Configure Module-ID

M => S

0	1	2	3	4	5	6	7
cs = 129	Mid	CAN	r	r	r	r	r

S => M

0	1	2	3	4	5	6	7
cs = 129	error code	spec error	r	r	r	r	r

cs LMT command specifier
 CAN number of CAN-controller (must be 0)
 Mid new module_id to configure
 error_code 0: successful
 1..254: reserved
 255: implementation specific error
 specific_error only valid if error_code equal 255
 3: Mid out of range
 4: command execution error
 6: illegal CAN-controller selected
 r reserved

7.6.2 Configure Bit Timing

M => S

0	1	2	3	4	5	6	7
cs = 130	table selector	table index	CAN	r	r	r	r

S => M

0	1	2	3	4	5	6	7
cs = 130	error code	spec error	r	r	r	r	r

cs LMT command specifier
 CAN number of CAN-controller (must be 0)
 table_selector 0: standard CiA bit timing table
 table_index 0: 1 Mbit
 1: 500 kbit
 2: 250 kbit
 3: 125 kbit
 4: -
 5: 50 kbit
 6: 20 kbit
 7: 10 kbit
 8: 800 kbit
 error_code 0: successful
 1: bit timing not supported
 2..254: reserved
 255: implementation specific error
 specific_error only valid if error_code equal 255
 3: value out of range
 4: command execution error
 6: illegal CAN-controller selected
 r reserved

7.6.3 Inquire Bit Timing

M => S

0	1	2	3	4	5	6	7
cs = 132	r	r	r	r	r	r	r

S => M

0	1	2	3	4	5	6	7
cs = 132	error code	spec error	table selector	table index	CAN	BTR0	BTR1

cs LMT command specifier
 CAN number of CAN-controller (allways 0)
 BTR0/BTR1 bit timing register 0 and 1 of CAN-controller
 table_selector standard CiA bit timing table (allways 0)
 table_index 0: 1 Mbit
 1: 500 kbit
 2: 250 kbit
 3: 125 kbit
 4: -
 5: 50 kbit
 6: 20 kbit
 7: 10 kbit
 8: 800 kbit
 error_code 0: successful
 1: bit timing not supported
 2..254: reserved
 specific_error only valid if error_code equal 255
 r reserved

8 Objektverzeichnis

Im folgenden werden die herstellerspezifischen listenorientierten Einträge des Objektverzeichnisses erläutert.

8.1 Übersicht Objektverzeichnis herstellerspezifischer Teil

Hinweis: Das gesamte Objektverzeichnis ist doppelt ausgelegt (ab Index 2001; ab Index 3001). Dies ist vorgesehen um gleiche Objekte mit unterschiedlichen Datentypen zu ermöglichen.

Aufbau der Indextabelle:

Der Index ist eine 4-stellige Zahl in hexadezimaler Notation mit folgendem Aufbau.

MSB		LSB	
Format	Funktionsblock	Funktion	Parameter-Nr.
2 = Fixed Point Format	0 = Gerät	laufende Nummer	laufende Nummer
3 = Float Format	1 = Input		
	2 = Controller		
	3 = Alarm		

Konfigurationsmode:

Änderungen der Konfiguration wie Ausgangszuordnungen oder Messbereichsendwerte werden nicht Online übernommen. Zur Änderung solcher Parameter muß der KS 800 in den Konfigurationsmode geschaltet werden. Die Mode-Umschaltung erfolgt über einen SDO-Zugriff.

Index 2008 (OPMod): Wert = 0 Konfigurationsmode
 Wert = 1 Normalmode

Zur Änderung von im Objektverzeichnis mit rw* gekennzeichneten Daten muß OPMod auf 0 gesetzt werden. Die neuen Parameter werden beim anschließenden Rückändern des OPMod von 0 -> 1 übernommen.

Verwendete Datenformate:

Unsigned8: 8-Bit-Wert, ohne Vorzeichen, Bereich 0...256
 Unsigned 16: 16-Bit-Wert, ohne Vorzeichen, Bereich 0...65535
 Fixedpoint1: 16-Bit Integer-Wert mit 1 festen Kommastelle, Bereich -32767...+32767. Bei der Interpretation dieser Werte wird die letzte Stelle als Nachkommastelle gewertet. z.B. 12345 bedeutet 1234,5 (°C), oder 873 sind 87,3 (%).

Spalte "ab Firmware": Ä = Diese Date gibt es bereits ab der Bedienversion 1, hat aber im Laufe der Entwicklung eine Änderung (Erweiterung) erfahren.
 N = Diese Date ist ab der angegebenen Bedienversion neu hinzu gekommen.

Index (hex)	Object (Symbolic name)	Name	Type	Attribut	ab Firmware
FB Gerät					
<i>Prozeßdaten Fktnr.0</i>					
2001	VAR	Status 1Unit-State1	Unsigned8	ro	1.0
2002	VAR	Basic Hardware Options HWbas	Unsigned16	ro	1.0
2003	VAR	SW Optionen SWopt	Unsigned16	ro	1.0
2004	VAR	SW Code-Nr. SWCode	Unsigned16	ro	1.0
2005	VAR	SW Version SWVersion	Unsigned16	ro	2.1
2006	VAR	Bedienversion OPVers	Unsigned16	ro	1.0
2007	VAR	Versionsstand des EEPROMS EEPVers	Unsigned16	ro	1.0
2008	VAR	Konfigurationsmode Umschaltung OPMo	Unsigned8	rw	1.0
2009	VAR	Stoppen/Starten der RSE aller Gruppenregler OStartg	Unsigned8	rw	1.0
200A	VAR	Zurücksetzen des lokalen Datenänderungsflags	Unsigned8	rw	1.0
<i>Parameter und Konfigurationsdaten Fktnr 0</i>					
200B	VAR	Baudrate COM 1 C900	Unsigned16	rw*	1.0
200C	VAR	Geräteadresse Adr1	Unsigned16	rw*	1.0
200D	VAR	C904	Unsigned16	rw*	1.0 Ä
200E	VAR	Baudrate COM 2 C902 (CAN Baudrate)	Unsigned16	rw*	1.0
200F	VAR	Geräteadresse Adr2 (CAN Node ID)	Unsigned16	rw*	1.0
2010	VAR	Freigabetemperatur der Kühlen-Funktion für Wasserkühlung	FixedPoint	rw	2.0
2011	VAR	Heizstromreset/-Schnelltest	Unsigned8	rw	4.0
<i>Prozeßdaten Fktnr. 2</i>					
2020	VAR	Status Alarmausgänge State_alarm_out	Unsigned8	ro	1.0
2021	VAR	Status dig. Ein./Ausgänge State_dio	Unsigned8	ro	1.0
<i>Parameter und Konfigurationsdaten Fktnr. 2</i>					
2022	VAR	HauptKonfig. C500	Unsigned16	rw*	1.0
2023	VAR	Hauptkonfig. C530	Unsigned16	rw*	1.0
2024	VAR	Zuordnung HC/Leckstrom C151	Unsigned16	rw*	1.0 Ä
2025	VAR	Heizstromzyklus Hccycl	Unsigned16	rw*	1.0

2026	VAR	Meßbereichsende für HC-Eingang HC 100	FixedPoint1	rw*	1.0
2027	VAR	Forced digital output OUT1...OUT8	Unsigned8	rw	2.1
2028	VAR	Forced digital output OUT9...OUT16	Unsigned8	rw	2.1
2029	VAR	Forced dig. output OUT17...OUT19 (FW 2.1) + Ableichrelais 1+2 (FW 3.0)	Unsigned8	rw	siehe Name
FB Input					
<i>Prozeßdaten Fktnr 0</i>					
2100	ARRAY	Singnal Input Fail Input_X_Failed	Unsigned8	ro	1.0
2101	ARRAY	Hauptregelgröße x1	FixedPoint1	ro	1.0
2102	ARRAY	Rohmeßwert vor Meßwertkorrektur INP1	FixedPoint1	ro	1.0
<i>Parameter und Konfigurationsdaten Fktnr 1</i>					
2110	ARRAY	Meßwertkorrektur X1 Input X1in	FixedPoint1	rw	1.0
2111	ARRAY	Meßwertkorrektur X1 Output X1out	FixedPoint1	rw	1.0
2112	ARRAY	Meßwertkorrektur X2 X2in	FixedPoint1	rw	1.0
2113	ARRAY	Meßwertkorrektur X2 X2out	FixedPoint1	rw	1.0
2114	ARRAY	Sensortyp und Einheit (T,H) C200	Unsigned16	rw*	1.0
2115	ARRAY	Fail: Fühlerbruch (T) C205	Unsigned16	rw*	1.0
2116	ARRAY	phys. Wert bei 0% X0	FixedPoint1	rw*	1.0
2117	ARRAY	phys. Wert bei 100% X100	FixedPoint1	rw*	1.0
2118	ARRAY	Ersatzwert bei Sensorfail XFail	FixedPoint1	rw*	1.0
2119	ARRAY	Filterzeitkonstante Meßwertverarbeitung Tfm	FixedPoint1	rw*	1.0
211A	ARRAY	angenommene TK Tkref	FixedPoint1	rw*	1.0
211B	ARRAY	Signalzuordnung dig. Signale C190	Unsigned16	rw*	1.0
FB Anal. Out					
<i>Prozeßdaten Fktnr. 0</i>					
2130	ARRAY	Forced analog output AOUT1...AOUT8	FixedPoint1	rw	3.0
2140	ARRAY	X0	FixedPoint1	rw*	3.0
2141	ARRAY	X100	FixedPoint1	rw*	3.0
2142	ARRAY	C540 Mode/Scr A-Out	Unsigned16	rw*	3.0
FB Controller					
<i>Prozeßdaten Fktnr 0</i>					
2200	ARRAY	Status 1	Unsigned8	ro	1.0
2201	ARRAY	eff. Sollwert Weff	FixedPoint1	ro	1.0
2202	ARRAY	eff. Istwert Xeff	FixedPoint1	ro	1.0
2203	ARRAY	wirksame Stellgröße Ypid	FixedPoint1	ro	1.0

2204	ARRAY	Regelabweichung xw	FixedPoint1	ro	1.0
2205	ARRAY	Automatik/Hand Umschaltung	Unsigned8	rw	1.0
2206	ARRAY	Starten der RSE OStart	Unsigned8	rw	1.0
2207	ARRAY	Umschalten Wext/Wint (Kaskadenregelung)	Unsigned8	rw	1.0
2208	ARRAY	Umschalten W/W2	Unsigned8	rw	1.0
2209	ARRAY	Regler ein/aus Coff	Unsigned8	rw	1.0
Parameter und Konfigurationsdaten Fktnr 0					
220A	ARRAY	Hauptkonfiguration 1, Regelung C100	Unsigned16	rw*	1.0
220B	ARRAY	Hauptkonfiguration 2, Regelung C101	Unsigned16	rw*	1.0
220C	ARRAY	Konfiguration Tuning C700	Unsigned16	rw*	1.0
220D	ARRAY	Signalzuordnung anal. C180	Unsigned16	rw*	1.0 Ä
Prozeßdaten Fktnr 1					
2210	ARRAY	Sollwertstatus WState	Unsigned8	ro	1.0
2211	ARRAY	wirksamer interner Sollwert Wint	FixedPoint1	ro	1.0
2212	ARRAY	int. Sollwert, nicht flüchtig Wnvol	FixedPoint1	rw	1.0
2213	ARRAY	int. Sollwert, flüchtig Wvol	FixedPoint1	rw	1.0
Parameter und Konfigurationsdaten Fktnr 1					
2214	ARRAY	Untere Sollwertgrenze für Weff W0	FixedPoint1	rw	1.0
2215	ARRAY	obere Sollwertgrenze für Weff W100	FixedPoint1	rw	1.0
2216	ARRAY	Zusatzsollwert W2	FixedPoint1	rw	1.0
2217	ARRAY	Sollwertgradient plus Grw+	FixedPoint1	rw	1.0
2218	ARRAY	Sollwertgradient minus Grw-	FixedPoint1	rw	1.0
2219	ARRAY	Sollwertgradient W2 Grw2	FixedPoint1	rw	1.0
221A	ARRAY	Loop-alarm ein/aus C102	Unsigned16	rw*	4.0 N
Parameter und Konfigurationsdaten Fktnr 2					
2220	ARRAY	Adaptions-Mode C710	Unsigned16	rw*	4.0 N
Parameter und Konfigurationsdaten Fktnr 3					
2230	ARRAY	Neutrale Zone Xsh	FixedPoint1	rw	1.0 Ä
2231	ARRAY	Mindestimpulslänge Tpuls	FixedPoint1	rw	1.0 Ä
2232	ARRAY	Motorlaufzeit des Stellmotors Tm	FixedPoint1	rw	1.0
2233	ARRAY	Schaltdifferenz Signalgerät Xsd1	FixedPoint1	rw	1.0
2234	ARRAY	Schaltpunktabstand Zusatzkontakt LW	FixedPoint1	rw	1.0
2235	ARRAY	Schaltdifferenz Zusatzkontakt Xsd2	FixedPoint1	rw	1.0
2236	ARRAY	Neutrale Zone Xsh1	FixedPoint1	rw	1.0
2237	ARRAY	Neutrale Zone Xsh2	FixedPoint1	rw	1.0

Prozeßdaten Fktnr 4					
2240	ARRAY	Differenz Stellgrößenvorgabe dYman	FixedPoint1	rw	1.0
2241	ARRAY	absolute Stellgrößenvorgabe Yman	FixedPoint1	rw	1.0
2242	ARRAY	increm. Stellgrößenverstellung Yinc	Unsigned8	rw	1.0
2243	ARRAY	decrem. Stellgrößenverstellung Ydec	Unsigned8	rw	1.0
2244	ARRAY	Geschwindigkeit für increm. und decrem. Stellgrößenverstellung Ygrw_is	Unsigned8	rw	1.0
Parameter und konfigurationsdaten Fktnr 4					
2245	ARRAY	untere Stellgrößenbegrenzung Ymin	FixedPoint1	rw	1.0
2246	ARRAY	obere Stellgrößenbegrenzung Ymax	FixedPoint1	rw	1.0
2247	ARRAY	Arbeitspunkt für Stellgröße Y0	FixedPoint1	rw	1.0
2248	ARRAY	max. Mittelwert der Stellgröße Yhm	FixedPoint1	rw	1.0
2249	ARRAY	Grenze für Mittelwertbildung LYh	FixedPoint1	rw	1.0
Prozeßdaten Fktnr 5					
2250	ARRAY	Status Tuning State_Tune1	Unsigned8	ro	1.0
2251	ARRAY	eff. Parameterzusatznummer ParNeff	Unsigned8	ro	1.0
2252	ARRAY	Parameterzusatznummer wirksam ParNr	Unsigned8	rw	1.0
2253	ARRAY	Verzugszeit Heizen Tu1	FixedPoint1	ro	1.0
2254	ARRAY	Anstiegsgeschwindigkeit Heizen Vmax1	FixedPoint1	ro	1.0
2255	ARRAY	Prozeßverstärkung Heizen Kp1	FixedPoint1	ro	1.0
2256	ARRAY	Fehlercode der RSE Heizen MSG1	Unsigned8	ro	1.0
2257	ARRAY	Verzugszeit Kühlen Tu2	FixedPoint1	ro	1.0
2258	ARRAY	Anstiegsgeschwindigkeit Kühlen Vmax	FixedPoint1	ro	1.0
2259	ARRAY	Prozeßverstärkung Kühlen Kp2	FixedPoint1	ro	1.0
225A	ARRAY	Fehlercode der RSE Kühlen MSG2	Unsigned8	ro	1.0
Parameter und Konfigurationsdaten Fktnr 5					
225B	ARRAY	Stellgröße während Prozeß in Ruhe YOpm	FixedPoint1	rw	1.0
225C	ARRAY	Sprunghöhe bei Identifikation dYOpm	FixedPoint1	rw	1.0
225D	ARRAY	Parametersatz, der optimiert werden soll POpm	Unsigned8	rw	1.0
225E	ARRAY	Hysterese bei Parameterumschaltung OXsd	FixedPoint1	rw	1.0
225F	ARRAY	Umschaltpunkt 1 Trig1	FixedPoint1	rw	1.0
Parameter und Konfigurationsdaten Fktnr 6					
2260	ARRAY	Proportionalbereich 1 Xp1_1	FixedPoint1	rw	1.0
2261	ARRAY	Nachstellzeit 1 Tn1_1	FixedPoint1	rw	1.0
2262	ARRAY	Vorhaltzeit 1 Tv1_1	FixedPoint1	rw	1.0
2263	ARRAY	min. Periodendauer 1 T1_1	FixedPoint1	rw	1.0

2264	ARRAY	Proportionalbereich 2 Xp2_1	FixedPoint1	rw	1.0
2265	ARRAY	Nachstellzeit 2 Tn2_1	FixedPoint1	rw	1.0
2266	ARRAY	Vorhaltzeit 2 Tv2_1	FixedPoint1	rw	1.0
2267	ARRAY	min. Periodendauer 2 T2_1	FixedPoint1	rw	1.0
Parameter und Konfigurationsdaten Fktnr 7					
2270	ARRAY	Proportionalbereich 1 Xp1_2	FixedPoint1	rw	1.0
2271	ARRAY	Nachstellzeit 1 Tn1_2	FixedPoint1	rw	1.0
2272	ARRAY	Vorhaltezeit 1 Tv1_2	FixedPoint1	rw	1.0
2273	ARRAY	min. Periodendauer 1 T1_2	FixedPoint1	rw	1.0
2274	ARRAY	Proportionalbereich 2 Xp2_2	FixedPoint1	rw	1.0
2275	ARRAY	Nachstellzeit 2 Tn2_2	FixedPoint1	rw	1.0
2276	ARRAY	Vorhaltezeit 2 Tv2_2	FixedPoint1	rw	1.0
2277	ARRAY	min. Periodendauer 2 T2_2	FixedPoint1	rw	1.0
Parameter und Konfigurationsdaten Fktnr 10					
22A0	ARRAY	max. Stellwert Ya	FixedPoint1	rw	1.0
22A1	ARRAY	Anfahrswert Wa	FixedPoint1	rw	1.0
22A2	ARRAY	Anfahrhaltezeit TPa	FixedPoint1	rw	1.0
FB Alarm					
Prozeßdaten Fktnr 0					
2300	ARRAY	Alarmstatus 1 Status_AL1	Unsigned8	ro	1.0
2301	ARRAY	Alarmstatus 2 Status_AL2	Unsigned8	ro	1.0
2302	ARRAY	Heizstrommeßwert HC	FixedPoint1	ro	1.0
Parameter und Konfigurationsdaten Fktnr 0					
2303	ARRAY	unterer Voralarm LimL	FixedPoint1	rw	1.0
2304	ARRAY	oberer Voralarm LimH	FixedPoint1	rw	1.0
2305	ARRAY	Alarmhysterese, Haupt- und Voralarme xsd	FixedPoint1	rw	1.0
2306	ARRAY	unterer Hauptalarm LimLL	FixedPoint1	rw	1.0
2307	ARRAY	oberer Hauptalarm LimHH	FixedPoint1	rw	1.0
2308	ARRAY	Heizstromgrenzwert LimHC	FixedPoint1	rw	1.0
2309	ARRAY	Alarmquelle/-funktion LL, C600	Unsigned16	rw*	1.0
230A	ARRAY	Alarmziel C 601	Unsigned16	rw*	1.0
230B	ARRAY	Alarmquelle/-funktion L, C602	Unsigned16	rw*	3.0 N
230C	ARRAY	Alarmquelle/-funktion H, C603	Unsigned16	rw*	3.0 N
230D	ARRAY	Alarmquelle/-funktion HH, C604	Unsigned16	rw*	3.0 N

Spalte "ab Firmware": Ä = Diese Date gibt es bereits ab der Bedienversion 1, hat aber im Laufe der Entwicklung eine Änderung (Erweiterung) erfahren.
 N = Diese Date ist ab der angegebenen Bedienversion neu hinzu gekommen.

Index (hex)	Object (Symbolic name)	Name	Type	Attribut	ab Firmware
FB Gerät					
<i>Prozeßdaten Fktnr.0</i>					
3001	VAR	Status 1Unit-State1	Unsigned8	ro	1.0
3002	VAR	Basic Hardware Options HWbas	Unsigned16	ro	1.0
3003	VAR	SW Optionen SWopt	Unsigned16	ro	1.0
3004	VAR	SW Code-Nr. SWCode	Unsigned16	ro	1.0
3005	VAR	SW Version SWVersion	Unsigned16	ro	2.1
3006	VAR	Bedienversion OPVers	Unsigned16	ro	1.0
3007	VAR	Versionsstand des EEPROMS EEPVers	Unsigned16	ro	1.0
3008	VAR	Konfigurationsmode Umschaltung OPMo	Unsigned8	rw	1.0
3009	VAR	Stoppen/Starten der RSE aller Gruppenregler OStartg	Unsigned8	rw	1.0
300A	VAR	Zurücksetzen des lokalen Datenänderungsflags	Unsigned8	rw	1.0
<i>Parameter und Konfigurationsdaten Fktnr 0</i>					
300B	VAR	Baudrate COM 1 C900	Unsigned16	rw*	1.0
300C	VAR	Geräteadresse Adr1	Unsigned16	rw*	1.0
300D	VAR	C904	Unsigned16	rw*	1.0 Ä
300E	VAR	Baudrate COM 2 C902 (CAN Baudrate)	Unsigned16	rw*	1.0
300F	VAR	Geräteadresse Adr2 (CAN Node ID)	Unsigned16	rw*	1.0
3010	VAR	Freigabetemperatur der Kühlen-Funktion für Wasserkühlung	Float	rw*	2.0
3011	VAR	Heizstromreset/-Schnelltest	Unsigned8	rw	4.0
<i>Prozeßdaten Fktnr. 2</i>					
3020	VAR	Status Alarmausgänge State_alarm_out	Unsigned8	ro	1.0
3021	VAR	Status dig. Ein./Ausgänge State_dio	Unsigned8	ro	1.0
<i>Parameter und Konfigurationsdaten Fktnr. 2</i>					
3022	VAR	HauptKonfig. C500	Unsigned16	rw*	1.0
3023	VAR	Hauptkonfig. C530	Unsigned16	rw*	1.0
3024	VAR	Zuordnung HC/Leckstrom C151	Unsigned16	rw*	1.0 Ä
3025	VAR	Heizstromzyklus Hccycl	Unsigned16	rw*	1.0
3026	VAR	Meßbereichsende für HC-Eingang HC 100	Float	rw*	1.0

3027	VAR	Forced digital output OUT1...OUT8	Unsigned8	rw	2.1
3028	VAR	Forced digital output OUT9...OUT16	Unsigned8	rw	2.1
3029	VAR	Forced dig. output OUT17...OUT19 (FW 2.1) + Ableichrelais 1+2 (FW 3.0)	Unsigned8	rw	siehe Name
FB Input					
<i>Prozeßdaten Fktnr 0</i>					
3100	ARRAY	Singnal Input Fail Input_X_Failed	Unsigned8	ro	1.0
3101	ARRAY	Hauptregelgröße x1	Float	ro	1.0
3102	ARRAY	Rohmeßwert vor Meßwertkorrektur INP1	Float	ro	1.0
<i>Parameter und Konfigurationsdaten Fktnr 1</i>					
3110	ARRAY	Meßwertkorrektur X1 Input X1in	Float	rw	1.0
3111	ARRAY	Meßwertkorrektur X1 Output X1out	Float	rw	1.0
3112	ARRAY	Meßwertkorrektur X2 X2in	Float	rw	1.0
3113	ARRAY	Meßwertkorrektur X2 X2out	Float	rw	1.0
3114	ARRAY	Sensortyp und Einheit (T,H) C200	Unsigned16	rw*	1.0
3115	ARRAY	Fail: Fühlerbruch (T) C205	Unsigned16	rw*	1.0
3116	ARRAY	phys. Wert bei 0% X0	Float	rw*	1.0
3117	ARRAY	phys. Wert bei 100% X100	Float	rw*	1.0
3118	ARRAY	Ersatzwert bei Sensorfail XFail	Float	rw*	1.0
3119	ARRAY	Filterzeitkonstante Meßwertverarbeitung Tfm	Float	rw*	1.0
311A	ARRAY	angenommene TK Tkref	Float	rw*	1.0
311B	ARRAY	Signalzuordnung dig. Signale C190	Unsigned16	rw*	1.0
FB Anal. Out					
<i>Prozeßdaten Fktnr. 0</i>					
3130	ARRAY	Forced analog output AOUT1...AOUT8	Float	rw	3.0
3140	ARRAY	X0	Float	rw*	3.0
3141	ARRAY	X100	Float	rw*	3.0
3142	ARRAY	C540 Mode/Scr A-Out	Unsigned16	rw*	3.0
FB Controller					
<i>Prozeßdaten Fktnr 0</i>					
3200	ARRAY	Status 1	Unsigned8	ro	1.0
3201	ARRAY	eff. Sollwert Weff	Float	ro	1.0
3202	ARRAY	eff. Istwert Xeff	Float	ro	1.0
3203	ARRAY	wirksame Stellgröße Ypid	Float	ro	1.0
3204	ARRAY	Regelabweichung xw	Float	ro	1.0

3205	ARRAY	Automatik/Hand Umschaltung	Unsigned8	rw	1.0
3206	ARRAY	Starten der RSE OStart	Unsigned8	rw	1.0
3207	ARRAY	Umschalten Wext/Wint (Kaskadenregelung)	Unsigned8	rw	1.0
3208	ARRAY	Umschalten W/W2	Unsigned8	rw	1.0
3209	ARRAY	Regler ein/aus Coff	Unsigned8	rw	1.0
Parameter und Konfigurationsdaten Fktnr 0					
320A	ARRAY	Hauptkonfiguration 1, Regelung C100	Unsigned16	rw*	1.0
320B	ARRAY	Hauptkonfiguration 2, Regelung C101	Unsigned16	rw*	1.0
320C	ARRAY	Konfiguration Tuning C700	Unsigned16	rw*	1.0
320D	ARRAY	Signalzuordnung anal. C180	Unsigned16	rw*	1.0 Ä
Prozeßdaten Fktnr 1					
3210	ARRAY	Sollwertstatus WState	Unsigned8	ro	1.0
3211	ARRAY	wirksamer interner Sollwert Wint	Float	ro	1.0
3212	ARRAY	int. Sollwert, nicht flüchtig Wnvol	Float	rw	1.0
3213	ARRAY	int. Sollwert, flüchtig Wvol	Float	rw	1.0
Parameter und Konfigurationsdaten Fktnr 1					
3214	ARRAY	Untere Sollwertgrenze für Weff W0	Float	rw	1.0
3215	ARRAY	obere Sollwertgrenze für Weff W100	Float	rw	1.0
3216	ARRAY	Zusatzsollwert W2	Float	rw	1.0
3217	ARRAY	Sollwertgradient plus Grw+	Float	rw	1.0
3218	ARRAY	Sollwertgradient minus Grw-	Float	rw	1.0
3219	ARRAY	Sollwertgradient W2 Grw2	Float	rw	1.0
321A	ARRAY	Loop-alarm ein/aus C102	Unsigned16	rw*	4.0 N
Parameter und Konfigurationsdaten Fktnr 2					
3220	ARRAY	Adaptions-Mode C710	Unsigned16	rw*	4.0 N
Parameter und Konfigurationsdaten Fktnr 3					
3230	ARRAY	Neutrale Zone Xsh	Float	rw	1.0 Ä
3231	ARRAY	Mindestimpulslänge Tpuls	Float	rw	1.0 Ä
3232	ARRAY	Motorlaufzeit des Stellmotors Tm	Float	rw	1.0
3233	ARRAY	Schaltdifferenz Signalgerät Xsd1	Float	rw	1.0
3234	ARRAY	Schaltpunktabstand Zusatzkontakt LW	Float	rw	1.0
3235	ARRAY	Schaltdifferenz Zusatzkontakt Xsd2	Float	rw	1.0
3236	ARRAY	Neutrale Zone Xsh1	Float	rw	1.0
3237	ARRAY	Neutrale Zone Xsh2	Float	rw	1.0
Prozeßdaten Fktnr 4					

3240	ARRAY	Differenz Stellgrößenvorgabe dYman	Float	rw	1.0
3241	ARRAY	absolute Stellgrößenvorgabe Yman	Float	rw	1.0
3242	ARRAY	increm. Stellgrößenverstellung Yinc	Unsigned8	rw	1.0
3243	ARRAY	decrem. Stellgrößenverstellung Ydec	Unsigned8	rw	1.0
3244	ARRAY	Geschwindigkeit für increm. und decrem. Stellgrößenverstellung Ygrw_is	Unsigned8	rw	1.0
Parameter und konfigurationsdaten Fktnr 4					
3245	ARRAY	untere Stellgrößenbegrenzung Ymin	Float	rw	1.0
3246	ARRAY	obere Stellgrößenbegrenzung Ymax	Float	rw	1.0
3247	ARRAY	Arbeitspunkt für Stellgröße Y0	Float	rw	1.0
3248	ARRAY	max. Mittelwert der Stellgröße Yhm	Float	rw	1.0
3249	ARRAY	Grenze für Mittelwertbildung LYh	Float	rw	1.0
Prozeßdaten Fktnr 5					
3250	ARRAY	Status Tuning State_Tune1	Unsigned8	ro	1.0
3251	ARRAY	eff. Parameterzusatznummer ParNeff	Unsigned8	ro	1.0
3252	ARRAY	Parameterzusatznummer wirksam ParNr	Unsigned8	rw	1.0
3253	ARRAY	Verzugszeit Heizen Tu1	Float	ro	1.0
3254	ARRAY	Anstiegsgeschwindigkeit Heizen Vmax1	Float	ro	1.0
3255	ARRAY	Prozeßverstärkung Heizen Kp1	Float	ro	1.0
3256	ARRAY	Fehlercode der RSE Heizen MSG1	Unsigned8	ro	1.0
3257	ARRAY	Verzugszeit Kühlen Tu2	Float	ro	1.0
3258	ARRAY	Anstiegsgeschwindigkeit Kühlen Vmax	Float	ro	1.0
3259	ARRAY	Prozeßverstärkung Kühlen Kp2	Float	ro	1.0
325A	ARRAY	Fehlercode der RSE Kühlen MSG2	Unsigned8	ro	1.0
Parameter und Konfigurationsdaten Fktnr 5					
325B	ARRAY	Stellgröße während Prozeß in Ruhe YOpm	Float	rw	1.0
325C	ARRAY	Sprunghöhe bei Identifikation dYopt	Float	rw	1.0
325D	ARRAY	Parametersatz, der optimiert werden soll POpt	Unsigned8	rw	1.0
325E	ARRAY	Hysterese bei Parameterumschaltung OXsd	Float	rw	1.0
325F	ARRAY	Umschaltpunkt 1 Trig1	Float	rw	1.0
Parameter und Konfigurationsdaten Fktnr 6					
3260	ARRAY	Proportionalbereich 1 Xp1_1	Float	rw	1.0
3261	ARRAY	Nachstellzeit 1 Tn1_1	Float	rw	1.0
3262	ARRAY	Vorhaltzeit 1 Tv1_1	Float	rw	1.0
3263	ARRAY	min. Periodendauer 1 T1_1	Float	rw	1.0
3264	ARRAY	Proportionalbereich 2 Xp2_1	Float	rw	1.0

3265	ARRAY	Nachstellzeit 2 Tn2_1	Float	rw	1.0
3266	ARRAY	Vorhaltzeit 2 Tv2_1	Float	rw	1.0
3267	ARRAY	min. Periodendauer 2 T2_1	Float	rw	1.0
Parameter und Konfigurationsdaten Fktnr 7					
3270	ARRAY	Proportionalbereich 1 Xp1_2	Float	rw	1.0
3271	ARRAY	Nachstellzeit 1 Tn1_2	Float	rw	1.0
3272	ARRAY	Vorhaltezeit 1 Tv1_2	Float	rw	1.0
3273	ARRAY	min. Periodendauer 1 T1_2	Float	rw	1.0
3274	ARRAY	Proportionalbereich 2 Xp2_2	Float	rw	1.0
3275	ARRAY	Nachstellzeit 2 Tn2_2	Float	rw	1.0
3276	ARRAY	Vorhaltezeit 2 Tv2_2	Float	rw	1.0
3277	ARRAY	min. Periodendauer 2 T2_2	Float	rw	1.0
Parameter und Konfigurationsdaten Fktnr 10					
32A0	ARRAY	max. Stellwert Ya	Float	rw	1.0
32A1	ARRAY	Anfahrswert Wa	Float	rw	1.0
32A2	ARRAY	Anfahrhaltezeit TPa	Float	rw	1.0
FB Alarm					
Prozeßdaten Fktnr 0					
3300	ARRAY	Alarmstatus 1 Status_AL1	Unsigned8	ro	1.0
3301	ARRAY	Alarmstatus 2 Status_AL2	Unsigned8	ro	1.0
3302	ARRAY	Heizstrommeßwert HC	Float	ro	1.0
Parameter und Konfigurationsdaten Fktnr 0					
3303	ARRAY	unterer Voralarm LimL	Float	rw	1.0
3304	ARRAY	oberer Voralarm LimH	Float	rw	1.0
3305	ARRAY	Alarmhysterese, Haupt- und Voralarme xsd	Float	rw	1.0
3306	ARRAY	unterer Hauptalarm LimLL	Float	rw	1.0
3307	ARRAY	oberer Hauptalarm LimHH	Float	rw	1.0
3308	ARRAY	Heizstromgrenzwert LimHC	Float	rw	1.0
3309	ARRAY	Alarmquelle/-funktion LL, C600	Unsigned16	rw*	1.0
330A	ARRAY	Alarmziel C 601	Unsigned16	rw*	1.0
330B	ARRAY	Alarmquelle/-funktion L, C602	Unsigned16	rw*	3.0 N
330C	ARRAY	Alarmquelle/-funktion H, C603	Unsigned16	rw*	3.0 N
330D	ARRAY	Alarmquelle/-funktion HH, C604	Unsigned16	rw*	3.0 N

9 Konfigurierung und Parametrisierung über den CAN-Bus.

Die nachfolgenden Ausführungen sind nur von Bedeutung, wenn der KS 800 über den CAN-Bus konfiguriert und parametrisiert wird. Normalerweise wird das Engineering-Tool hierfür verwendet, das bereits die gegenseitige Prüfung der Konfigurationsdaten und die richtige Reihenfolge der Daten beim Download berücksichtigt.

Die unten stehende Aufstellung gibt die Reihenfolge der Daten an, in der sie an den KS 800 gesendet werden müssen, damit eine korrekte Konfigurierung und Parametrisierung erfolgt. Damit ist sichergestellt, daß Werte, die sich gegenseitig beeinflussen, den passenden Inhalt bekommen.

z.B. Eine Änderung der Fühlerart setzt die Sollwertgrenzen W0 und W100 auf die Default-Werte zurück (physikalische Meßgrenzen der jeweiligen Fühlerart). Würde man zuerst die Sollwertgrenzen auf W0 = 50°C und W100 = 350°C setzen und anschließend die Thermoelement-Art auf Typ K festlegen, würden die Sollwertgrenzen auf W0 = 0°C und W100 = 1350°C zurückgesetzt.

Außerdem ist zu beachten, daß bei den Parametern par_w0 und par_w100, bzw. par_ymin und par_ymax eine bestimmte Reihenfolge bei der Übertragung eingehalten werden muß, da diese beiden Wertepaare auf Plausibilität geprüft werden. Es muß immer zuerst der größere Wert gesendet werden.

Sind die Werte par_w0 auf 100°C und par_w100 auf 150°C gesetzt (Begrenzung der Sollwertstellungen zwischen 100 und 150°C) und sollen auf par_w0 = 160°C, bzw par_w100 = 200°C geändert werden, können die Daten nicht in der unten angegebenen Reihenfolge gesendet werden (zuerst par_w0 und anschließend par_w100), weil die neue untere Sollwertgrenze von 160°C über der alten oberen Grenze (150°C) liegt. In diesem Fall muß zuerst die neue obere Grenze (par_w100) auf 200°C angehoben werden und anschließend die untere Grenze auf den neuen Wert nachgefahren werden.

9.1 KS 800 über CAN-Bus-Schnittstelle konfigurieren:

Die in den Abschnitten 9.1 bis 9.3 beschriebenen Angaben brauchen bei der Verwendung des Engineering-Tools nicht berücksichtigt zu werden, da sie automatisch berücksichtigt werden, bzw. die Menues entsprechend gestaltet sind.

1. das Gerät offline schalten
2. Konfigurationsdaten (in der richtigen Reihenfolge) senden
3. das Gerät online schalten (die Kanäle arbeiten mit den alten Parameterwerten)
4. Parameterdaten senden (sie werden während des laufenden Betriebes übernommen)

9.1.1 Abspeichern des nichtflüchtigen Sollwertes.

Der KS 800 verfügt über zwei Arten der Speicherung des Sollwertes:

1: Der flüchtige Sollwert, der im RAM gespeichert wird. Auf den dort gespeicherten Wert greift der Regler im normalen Regelbetrieb zurück.

2: Der nicht-flüchtige Sollwert, der im EEPROM gespeichert wird. Dieser Sollwert dient den Reglern als "Start-Sollwert" nach der Initialisierung (Spannung ein, oder neue Konfiguration). Er wird in das RAM kopiert und dann als "Arbeits-" Sollwert verwendet, bis von einer Steuerung der RAM-Sollwert überschrieben wird.

Wird der nicht-flüchtige Sollwert (EEPROM) im Zuge eine Neukonfiguration geändert, erfolgt das

Schreiben in das EEPROM (Speichern) erst wenn der betreffende Regler "an der Reihe" ist. Dies kann max. 2 Gerätezyklen dauern (~1,2s). Während der Zeit, vom Ende des Schreibens ins RAM bis zum Neustart des Gerätes werden die Daten aus dem RAM verwendet. Bevor die Daten nicht im EEPROM gespeichert sind, darf das Gerät nicht "online" geschaltet werden, denn es könnte sein, daß manche neuen Daten noch nicht im EEPROM gespeichert sind und nach der Neuinitialisierung, bzw. nach einem Neustart wieder die alten (Soll-)Werte verwendet werden.

9.2 Plausibilitätsprüfung der Daten bei der Übertragung

Achtung!

Die Konfigurationsdaten werden von dem Gerät nicht überprüft. (Diese Überprüfung erfolgt im E-Tool.)

Die Parameterdaten werden auf ihre Grenzen überprüft.

9.3 Zuordnungstabelle Firmwareversion, Bedienversion, Serien-Nummer und E-Tool

Datum	Serien-Nummer	Firmware-Version	Bedien-Version	E-Tool-Version
3.8.97	8330	1.0	1	1.0
10.9.97	8334	1.1	1	1.1
28.10.97	8335	1.2	1	1.1
16.12.97	8337	1.3	1	1.1
8.4.98	8341	1.3	1	1.3
11.5.98	8341	1.3	1	1.3
23.9.99	8358	2.0	2	1.4
15.12.99	8361	2.1	2	2.0
7.3.00	8363	2.1	2	3.0SR1
7.3.00	8363	2.2	2	3.0SR1
7.3.00	8364	2.3	3	3.1SR1
10.7.00	8366	2.4	3	3.1SR1
30.3.01	8376	2.5	4	3.2SR3
13.11.01	8384	3.0	5	3.4SR1
19.8.02	8393	4.0	6	1.4SR1 (BlueControl)

9.4 Reihenfolge der Daten zur Konfigurierung und Parametrisierung

In der Spalte "Funktion" sind die einzelnen Konfigurationsdaten aufgeführt, deren Bedeutungen in der Funktionsbeschreibung 9499 040 49218 im Einzelnen beschrieben sind. Die Ziffern hinter dem (Unterstrich) besagen, welche Digit in dem Konfigurationswort verwendet werden.

16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Digit 4				Digit 3				Digit 2				Digit 1			

In der Spalte "Einführung" ist die Bedienversion der Einführung der Konfigurationsdate angegeben. Neue Konfigurationsdaten sind mit "ab BV x" und Erweiterungen mit "zus. ab BV x"

Konfiguration Gerät

<i>CAN Index</i>	<i>Funktion</i>	<i>Einführ.</i>	<i>CAN Index</i>	<i>Funktion</i>	<i>Einführ.</i>
200B	c900_32	BV 1	2115	c205_432	BV 1
200C	c901_4321	BV 1	211B	c190_21	BV 1
200D	c904_4	BV 1	2116	c201_4321	BV 1
	c904_1	zus. ab BV ≥ 2	2117	c202_4321	BV 1
	c904_32	zus. ab BV ≥ 5			
200E	c902_32	BV1	2118	c213_4321	BV 1
200F	c903_4321	BV1	2119	c214_4321	BV 1
2026	c302_4321	BV1	211A	c210_4321	BV 1
2022	c500_4321	BV 1	2309	c600_4321	BV 1
2023	c530_432	BV 1	230A	c601_4321	BV 1
2024	c151_432	BV 1	230B	c602_4321	ab BV ≥ 6
	c151_1	zus. ab BV ≥ 3	230C	c603_4321	ab BV ≥ 6
2025	c150_4321	BV 1	230D	c604_4321	ab BV ≥ 6

Konfiguration Kanäle

220A	c100_4321	BV 1	2140	x0	ab BV ≥ 5
220B	c101_4321	BV 1	2141	x100	ab BV ≥ 5
220C	c700_4321	BV 1	2142	c540_31	ab BV ≥ 5
220D	c180_3	BV 1			
	c180_4	zus. ab BV ≥ 5			
221A	c102_1	ab BV ≥ 6			
2220	c710_21	ab BV ≥ 6			
2114	c200_432	BV 1			

Parameter Gerät2010 par_wk_enable ab BV ≥ 2 **Parameter Kanäle**

2214	par_w0	BV 1	2262	par_tv1_0	BV 1
2215	par_w100	BV 1	2263	par_t1_0	BV 1
2216	par_w2	BV 1	2264	par_xp2_0	BV 1
2217	par_grwp	BV 1	2265	par_tn2_0	BV 1
2218	par_grwm	BV 1	2266	par_tv2_0	BV 1
2219	par_grw2	BV 1	2267	par_t2_0	BV 1
2230	par_xsh	BV 1	2270	par_xp1_1	BV 1
(2230	par_xsh_wk)	BV 1	2271	par_tn1_1	BV 1
2231	par_tpuls	BV 1	2272	par_tv1_1	BV 1
(2231	par_tpuls_wk)	BV 1	2273	par_t1_1	BV 1
2232	par_tm	BV 1	2274	par_xp2_1	BV 1
2233	par_xsd1	BV 1	2275	par_tn2_1	BV 1
2234	par_lw	BV 1	2276	par_tv2_1	BV 1
2235	par_xsd2	BV 1	2277	par_t2_1	BV 1
2236	par_xsh1	BV 1	22A0	par_ya	BV 1
2237	par_xsh2	BV 1	22A1	par_wa	BV 1
2245	par_ymin	BV 1	22A2	par_tpa	BV 1
2246	par_ymax	BV 1	2110	par_x1in	BV 1
2247	par_y0	BV 1	2111	par_x1out	BV 1
2248	par_yh	BV 1	2112	par_x2in	BV 1
2249	par_lyh	BV 1	2113	par_x2out	BV 1
225B	par_yoptm	BV 1	2303	par_lim_l	BV 1
225C	par_dyopt	BV 1	2304	par_lim_h	BV 1
225E	par_oxsd	BV 1	2305	par_lim_xsd	BV 1
225F	par_trig1	BV 1	2306	par_lim_ll	BV 1
225D	par_popt	BV 1	2307	par_lim_hh	BV 1
2260	par_xp1_0	BV 1	2308	par_lim_hc	BV 1
2261	par_tn1_0	BV 1			

Anschließend können die Prozeßdaten in beliebiger Reihenfolge gelesen und geschrieben werden.

Wird Wasserkühlung verwendet, gelten die Parameter 2230 und 2231 mit der Erweiterung _wk.

10 Applikationsbeispiele

10.1 Definition der Beispiele

Nach Auslieferung des KS 800 ist der Regler mit den in der Reglerbeschreibung angegebenen Default-Parametern eingestellt. Die folgenden Beispiele sollen den Ablauf von Umkonfigurierung und Inbetriebsetzung des Reglers verdeutlichen. In Beispiel 1 soll ein Heizen/Kühlen-Regler mit PT100 Eingang betrieben werden. Während in Beispiel 1 die Default-Regelparameter genutzt werden soll im 2. Beispiel die Selbstoptimierung zu den optimalen Parametern führen.

10.2 Beispiel 1 (Heizen/Kühlen-Regler)

Nachfolgend wird der Regler von der Defaulteinstellung (Heizen) auf Heizen/Kühlen umparametrisiert. Dann werden die Istwerte zyklisch (via SDOs) gelesen. Im Beispiel wird Kanal 1 genutzt. Der Subindex ist daher entsprechend der Kanalnummer bei allen Zugriffen auf "1" zu setzen.

- Konfigurationsmode wählen
→ **Index 0x2008** (OPMod=0)
Verweis: Prozeßdaten, Einzelcode 31

- Reglerhauptkonfiguration C100=0x0300 setzen (H/K-Regler)
→ **Index 0x200A**
Verweis: C100 - Reglerbeschreibung

- Hauptkonfiguration C200=0x2010 setzen (PT100, Einheit: C)
→ **Index 0x2114**
Verweis: C200 - Reglerbeschreibung

- Konfigurationsmode verlassen und Werte übernehmen
→ **Index 0x2008** (OPMod=1)
Verweis: Prozeßdaten, Einzelcode 31

- Sollwert Weff setzen
→ **Index 0x2313** (Weff=250)

- Regler in AUTO setzen
→ **Index 0x2205** (A/H=0)
Verweis: Controller, Einzelcode 33

- Schleife für Istwerte:
→ **for(;;)**
 {
 - Effektiver Istwert Kanal 1 lesen
 → **Index 0x2202**
 - wait(1sec)
 }

10.3 Beispiel 2 (H/K-Regler einstellen und Selbstoptimierung aktivieren)

Nachfolgend wird nach der Konfigurierung als H/K-Regler eine Selbstoptimierung gestartet. Nach erfolgreicher Optimierung werden die Istwerte zyklisch (via SDOs) gelesen. Wie in Beispiel 1 ist der Subindex immer auf "1" zu setzen (Regler 1).

- Konfigurationsmode wählen
 - **Index 0x2008** (OPMod=0)
Verweis: Prozeßdaten, Einzelcode 31
- Reglerhauptkonfiguration C100=0x0300 setzen (H/K-Regler)
 - **Index 0x200A**
Verweis: C100 - Reglerbeschreibung
- Hauptkonfiguration C200=0x2010 setzen (PT100, Einheit: C)
 - **Index 0x2114**
Verweis: C200 - Reglerbeschreibung
- Konfigurationsmode verlassen und Werte übernehmen
 - **Index 0x2008** (OPMod=1)
Verweis: Prozeßdaten, Einzelcode 31
- Sollwert Weff setzen
 - **Index 0x2313** (Weff=250)
- Optimierung starten
 - **Index 0x2206** (OStart=1)
Verweis: Controller, Einzelcode 34
 - OPTIMIERUNG = GESTARTET
- Optimierungsstatus auswerten
 - **while(OPTIMIERUNG <> BEENDET)**
 - {
 - Optimierungsstatus lesen
 - **Index 0x2250** (State_Tune_1)
Verweis: Controller, Einzelcode 1
 - fallende Flanke des Orun-bits abwarten
 - if(State_Tune_1[Orun] == 1)
OPTIMIERUNG = AKTIV
 - if(State_Tune_1[Orun] == 0 AND OPTIMIERUNG == AKTIV)
OPTIMIERUNG = BEENDET
 - wait(1sec)
 - }
- Fehlermeldungen prüfen
 - **if(State_Tune1[Oerr == 0])**
 - {
 - Regler läuft mit neuen Parametern, da Optimierung erfolgreich
 - Schleife für Istwerte:
 - **for(;;)**
 - {
 - Effektiver Istwert Kanal 1 lesen
 - **Index 0x2202**

```
        - wait( 1sec)
      }
}
else
{
- Fehlermeldung Heizen lesen
  → Index 0x2256 (MSG1)
    Verweis: Controller, Einzelcode 35

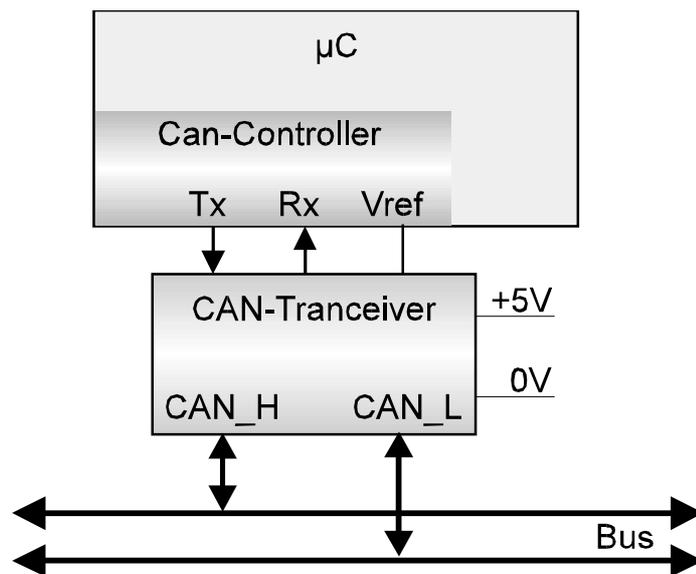
- Fehlermeldung Kühlen lesen
  → Index 0x225A (MSG2)
    Verweis: Controller, Einzelcode 39
}
```

11 CAN Physical Layer

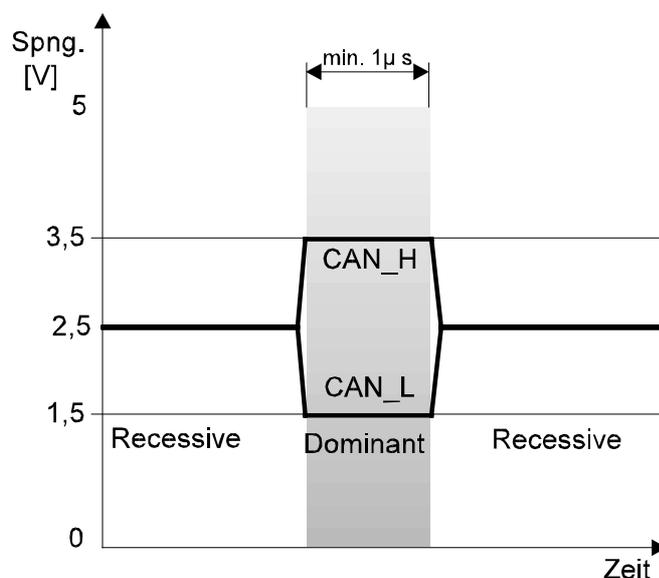
Es gibt eine Reihe von genormten Standards bezüglich des CAN Physical Layers. Der wichtigste für allgemeine Anwendungen ist der "CAN High-Speed Standard ISO 11898-2". Die nachfolgenden Empfehlungen basieren primär auf diesem Standard und gelten unabhängig vom verwendeten CAN-Protokoll (CANopen / DeviceNet).

11.1 ISO 11898-2 Knoten:

Ein ISO 11898-2 konformer Knoten besteht aus einem μC mit CAN-Controller (evtl. auch integriert), der über Rx- und Tx-Line mit einem CAN-Transceiver verbunden ist. Der Transceiver wiederum ist mit den differentiellen CAN-H und CAN-L Leitungen am CAN-Bus angeschlossen. Dieser (Transceiver-) Anschluß ist beim KS 800 galvanisch getrennt ausgeführt.



Die nominellen CAN-Buspegel werden beim CAN-Bus mit "Recessive" (nominelle Spannung von 2,5 V für CAN-H und CAN-L) und "Dominant" (nominell 3,5V für CAN-H und 1,5V für CAN-L) bezeichnet.



11.2 Baudraten und Buslängen

Die maximale, nutzbare Buslänge in einem CAN-Netzwerk wird durch eine Vielzahl von Einflüssen bestimmt, vor allem durch die folgenden physikalischen Effekte:

- Verzögerungszeiten der angeschlossenen Bus-Knoten (mit/ohne Opto-Koppler) und Verzögerungszeit des Bus-Kabels (propagation delays)
- unterschiedliche Abtastzeitpunkte innerhalb einer CAN-Bit-Zelle, bedingt durch Oszillatortoleranzen der einzelnen Bus-Knoten
- Signal-Amplituden Dämpfung, bedingt durch den ohmschen Widerstand des Bus-Kabels und den Eingangs-Widerständen der Bus-Knoten

Die im folgenden aufgeführten praktischen Buslängen können bei Verwendung von ISO 11898-2 konformen Transceivern mit Standard Buskabeln erreicht werden.

Bei den hohen Baudraten (1 MBd / 800 kBd) kann es allerdings durch die Anzahl/Geschwindigkeit der evtl. vorhandenen Opto-Koppler (galv. Trennung) zu erheblich kürzeren Buslängen kommen !

11.3 Praktische Buslängen

CAN-Profil(e)	Baud-Rate	Buslänge	Nominelle Bit-Time
CANopen	1 MBd	30 m	1 μ s
CANopen	800 kBd	50 m	1,25 μ s
CANopen/DeviceNet	500 kBd	100 m	2 μ s
CANopen/DeviceNet	250 kBd	200 m	4 μ s
CANopen/DeviceNet	125 kBd	500 m	8 μ s
CANopen	50 kBd	1000 m *)	20 μ s
CANopen	20 kBd	2500 m *)	50 μ s
CANopen	10 kBd	5000 m *)	100 μ s

*) Bei großen Kabellängen ist der Einsatz von galv. Trennungen und Repeatern zwingend notwendig

Weitere Hinweise zu den Buslängen können auch den Standards CiA „DS-102“ (CANopen) bzw. den ODVA "DeviceNet Specifications Volume I, Release 2.0", speziell Appendix A und B entnommen werden.

11.4 Kabel-Parameter

ISO 11898-2 definiert einige DC- bzw. AC-Parameter für die in CAN-Bus Netzwerken einsetzbaren Kabel (typischerweise kommen paarweise verdrehte Kabel mit definierten elektrischen Eigenschaften zum Einsatz). Die wichtigen AC Parameter sind 120 Ohm Kabelimpedanz und eine nominelle "propagation delay" von 5 ns/m ! Empfehlungen für die zu verwendenden Buskabel und Abschlußwiderstände können der folgenden Tabelle entnommen werden:

Bus-Länge	Buskabel (Z: 120 Ohm, tp: 5ns/m)		Abschlußwiderstand	Max. Bit-Rate
	Spez. Widerstand	Kabelquerschnitt		
0 ..40 m	70 mOhm/m	0,25mm ² , 0,34mm ² AWG 23, AWG 22	124 Ohm, 1%	1 MBd @ 40m
40 m .. 300 m	<60 mOhm/m	0,34mm ² , 0,6mm ² AWG 22, AWG 20	127 Ohm, 1% *)	> 500 kBd @ 100m
300 m .. 600 m	<40 mOhm/m	0,5mm ² , 0,6mm ² AWG 20	127 Ohm, 1% *)	> 100 kBd @ 500m
600 m ..1 km	<26 mOhm/m	0,75mm ² , 0,8mm ² AWG 18	127 Ohm, 1% *)	> 50 kBd @ 1 km

*) Bei großen Kabellängen ist ein höherer Wert für den Abschlußwiderstand (150 .. 300 Ohm) hilfreich, zur Reduzierung der Dämpfung .

Weitere Empfehlungen für CAN Netzwerke (speziell auch mit großer Ausdehnung):

- galv. Trennungen sind notwendig bei großen Längen (z.B. bei 400m Buskabel)
- separate Ground-Leitung ist sinnvoll
- der Spannungseinbruch (Potentialdifferenz) zwischen den Ground-Potentialen der Transceiver sollte gering sein (kleiner 2V). Einspeisung des Netzteils evtl. in der Mitte des Kabels)
- der Gesamt-Eingangswiderstand der Bus-Knoten sollte > 500 Ohm sein
- evtl. notwendige Stichleitungen sollten so kurz wie möglich sein, um Reflektionen zu vermeiden/verringern, z.B. < 6m @ 500kBd (DeviceNet). Bei höheren Baudraten < 1m !

Weitergehende Informationen sind bei der ODVA (DeviceNet), dem CiA (CANopen), den diversen Chip-Herstellern und im Internet zu bekommen.

