



KS 816

Multi - Meßumformer Multi - Temperaturregler

KS 816

PROFIBUS-DP

PROFIBUS
PROCESS FIELD BUS

Schnittstellenbeschreibung
PROFIBUS - DP
9499 040 56218
gültig ab: 8377

SIMATIC® ist ein eingetragenes Warenzeichen der Siemens AG
STEP® ist ein eingetragenes Warenzeichen der Siemens AG
 ist ein eingetragenes Warenzeichen der
PROFIBUS Nutzerorganisation (PNO)

© PMA Prozeß- und Maschinen-Automation GmbH 2001 Printed in Germany (0104)
Alle Rechte vorbehalten. Ohne vorhergehendes schriftliche Genehmigung
ist der Nachdruck oder die auszugsweise fotomechanische oder
anderweitige Wiedergabe dieses Dokumentes nicht gestattet.

Dies ist eine Publikation von PMA Prozeß- und Maschinen Automation
Postfach 310229
D-34058 Kassel
Germany

Inhalt

1	Allgemeines	5
1.1	Lieferumfang.	6
2	Hinweise zum Betrieb	7
2.1	Anschluß der Schnittstelle	7
2.1.1	Verlegen von Leitungen	7
3	Prozeßdaten	8
3.1	Als Statusbyte sind definiert:	13
3.2	Status und Diagnosemeldungen	15
3.3	Verriegelungsmechanismus bei Änderungen	16
3.4	Übertragung von Prozeßdaten	16
3.5	Übertragung von Parametern	16
3.5.1	Nachrichtenelemente	16
3.5.2	Allgemeiner Übertragungsaufbau	18
3.5.3	Ablauf Schreiben der Daten	18
3.5.4	Ablauf Lesen der Daten.	19
3.6	Beispiele	19
3.6.1	Prinzipien des Funktionsblock-Protokolls	19
3.6.2	Einzelzugriff	19
3.6.3	Blockzugriff (Zehner-Block).	20
3.6.4	Blockzugriff (Gesamt-Block)	21
3.7	Datentypen	22
4	Schnelleinstieg mit S7	23
4.1	Beispiel einer Testumgebung:	23
5	Funktionsblock-Protokoll	25
5.1	Datenstrukturierung	25
5.2	CODE-Tabellen	26
5.2.1	Aufbau Konfigurationsworte (C.xxxx)	26
5.2.2	GERÄT	26
5.2.3	Frei konfigurierbar	29
5.2.4	INPUT	30
5.2.5	CONTR	31
5.2.6	ALARM.	35
6	Funktionsbaustein für SIMATIC® S7	36
6.1	Aufbau	36
7	Anhang	39
7.1	Begriffe	39
7.2	GSD Datei.	39

1 Allgemeines

Die Ausführung (9407-481-30001) des Multi-Meßumformers und Multi-Temperaturreglers KS 816 sind mit einer PROFIBUS-DP Schnittstelle ausgerüstet, über die eine Übertragung der Prozeß-, Parameter- und Konfigurationsdaten möglich ist. Der Anschluß erfolgt über die 9-polige Sub-D Buchse. Die serielle Kommunikationsschnittstelle ermöglicht Verbindungen zu übergeordneten Steuerungen, Visualisierungstools etc.

Eine weitere, standardmäßig immer vorhandene Schnittstelle ist die PC-Schnittstelle. Diese dient dem Anschluß eines Engineering Tools, das auf einem PC abläuft.

Die Kommunikation erfolgt nach dem Master/Slave-Prinzip. Der KS 816-DP ist immer Slave.

Das Leitungsmedium sowie die physikalischen und elektrischen Eigenschaften der Schnittstelle:

- W **Netzwerk Topologie**
Linearer Bus mit aktivem Busabschluß an beiden Enden. Stichleitungen sind möglich (abhängig vom verwendeten Kabeltyp ist eine maximale Gesamtstichleitungslänge bei 1,5Mbit/s von 6,6m und bei 3-12Mbit/s von 1,6m möglich).
- W **Übertragungsmedium**
geschirmte, verdrehte 2-Drahtleitung (r EN 50170 Vol.2).
- W **Baudraten und Leitungslängen (ohne Repeater)**
Die maximale Leitungslänge ist abhängig von der verwendeten Übertragungsrate. Die Baudrate wird durch die Masterkonfiguration vorgegeben.

- W **Automatische Baudratenerkennung**

Baudrate	Maximale Leitungslänge
9,6 / 19,2 / 93,75 kbit/s	1200 m
187,5 kbit/s	1000 m
500 kbit/s	400 m
1,5 Mbit/s	200 m
3 ... 12 Mbit/s	100m

- W **Schnittstelle**
RS485 mit Sub-D Stecker (9 polig) anschließbar.
- W **Adresseinstellungen**
Die Adresseinstellung kann auf zwei Arten erfolgen:
-Einstellung über Codierschalter, Bereich 00 ... 99, Default 00
-Einstellung per Software, Bereich 0 ... 126, Default 126
Stehen die Codierschalter auf '00', so gilt die eingestellte Softwareadresse.
Eine veränderte Codierschalter-Adresse wird erst nach Wiedereinschalten der Spannungsversorgung aktiv.
- W 32 Geräte in einem Segment. Mit Repeater auf 127 erweiterbar.

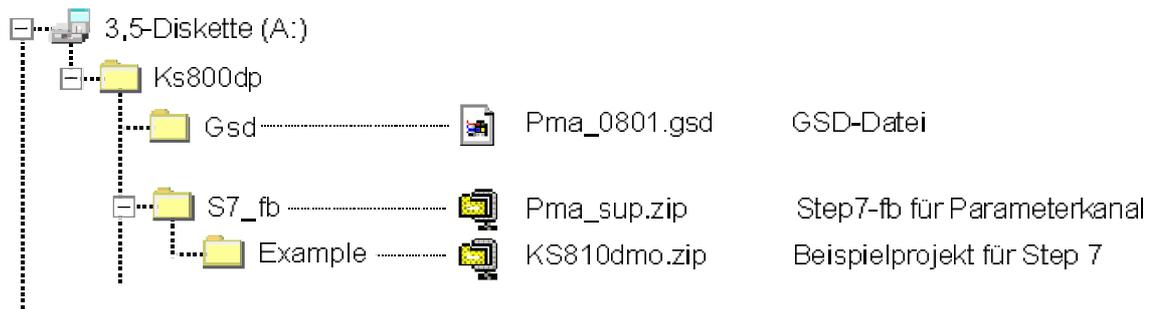
Der KS 816 mit PROFIBUS-DP Schnittstelle bietet hinsichtlich Handhabung und Integration in ein PROFIBUS Netzwerk viele Vorteile.

- W **Diagnose und Überwachung über COM-LED**
LED aus:Fehlerkennzeichnung für 'kein Buszugriff' (noch nicht vom Master angesprochen).
LED ein:OK, zyklischer Datenverkehr läuft
LED blinkt:(2Hz) Nutzdatenverkehr unterbrochen
LED blinkt:(4Hz) PROFIBUS- Parametrier- oder -Konfigurationsfehler.
- W **Besonderheiten**
Konfigurierbare Prozeßdatenmodule
Direktes Lesen und Schreiben von Ein- und Ausgängen
Einfache Anbindung an Steuerungen

1.1 Lieferumfang

Das Engineering Set besteht aus:

⌘ Diskette



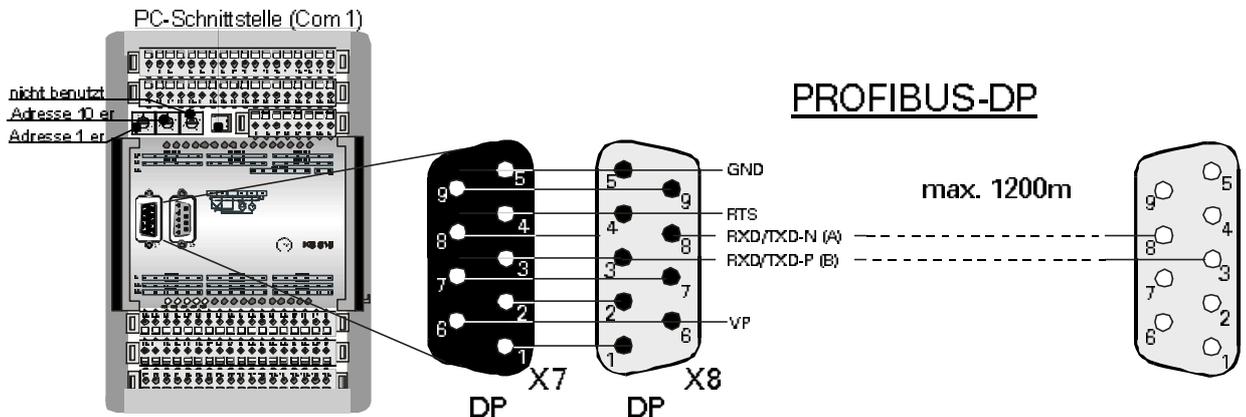
⌘ Schnittstellenbeschreibung für PROFIBUS-DP

2 Hinweise zum Betrieb

2.1 Anschluß der Schnittstelle

Der PROFIBUS wird an der 9-poligen Sub-D Buchse angeschlossen. Serielle Schnittstelle, physikalische Signale auf RS485-Basis.

Fig.: 1 Anschluß PROFIBUS-DP



Der Aufbau entspricht der Kabelführung vom Anwender durchzuführen. Dabei sind die allgemeinen Kabelspezifikationen nach EN 50170 Vol.2 zu beachten.

2.1.1 Verlegen von Leitungen

Bei der Leitungsverlegung sind die vom Lieferant der Masterbaugruppe gemachten allgemeinen Hinweise zum Verlegen von Leitungen zu beachten:

- W Leitungsführung innerhalb von Gebäuden (innerhalb und außerhalb von Schränken)
- W Leitungsführung außerhalb von Gebäuden
- W Potentialausgleich
- W Schirmung von Leitungen
- W Maßnahmen gegen Störspannungen
- W Länge der Stichleitung
- W Busabschlußwiderstände sind nicht im Lieferumfang des KS816-DP enthalten, sondern müssen über den externen Anschlußstecker realisiert werden.
- W Erdung

g Spezielle Hinweise zum Verlegen von PROFIBUS-Kabeln sind der PNO Technischen Richtlinie "**Aufbau Richtlinien für PROFIBUS-DP/FMS**" (Best-Nr. 2.111 [dt]; 2.112 [engl.]) zu entnehmen.

3 Prozeßdaten

Bei der Datenübertragung wird zwischen zyklisch zu übertragenden Prozeßdaten und azyklisch zu übertragenden Parameter- / Konfigurationsdaten unterschieden. Um das E/A Datenfeld den Anforderungen der Steuerungsaufgaben anpassen zu können, ist es modular aufgebaut.

Die Wahl des Prozeßdatenmoduls erfolgt über Konfigurationstools der Masteranschlüssen (z.B. bei Siemens S5 über COM PROFIBUS).

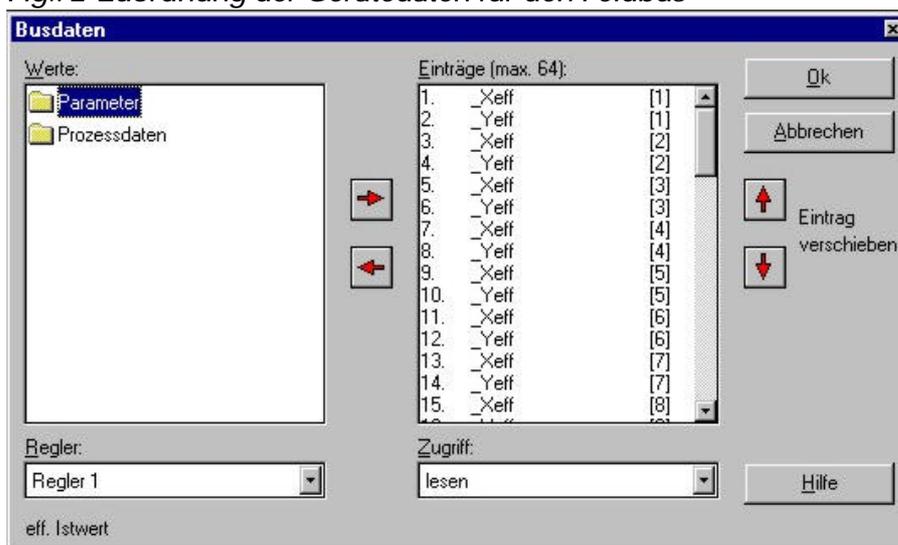
Folgende Prozeßdatenmodule können konfiguriert werden:

Prozeßdatenmodul A:	lesen (56 Byte) ¹⁾ Gerätestatus, 8 x (Istwert, Stellwert, Status, ...)	schreiben (8 Byte) ¹⁾ Gerätesteuerung, 8 x (Sollwert, Stellwert, ...)	mit Parameterkanal
Prozeßdatenmodul B:	lesen (106 Byte) ¹⁾ Gerätestatus, 8 x (Istwert, Stellwert, Status, ...)	schreiben (106 Byte) ¹⁾ Gerätesteuerung, 8 x (Sollwert, Stellwert, ...)	mit Parameterkanal
Prozeßdatenmodul C:	Nur Parameterkanal (8/8 Byte)		
Prozeßdatenmodul D:	lesen (114 Byte) ¹⁾ Gerätestatus, 8 x (Istwert, Stellwert, Status, ...)	schreiben (114 Byte) ¹⁾ Gerätesteuerung, 8 x (Sollwert, Stellwert, ...)	mit Parameterkanal
Prozeßdatenmodul E:	lesen (90 Byte) ¹⁾ Gerätestatus, (52 Variable Prozeßdaten)	schreiben (90 Byte) ¹⁾ Gerätesteuerung, (52 variable Prozeßdaten)	mit Parameterkanal
Prozeßdatenmodul F:	lesen (14 Byte) ¹⁾ Gerätestatus, (40 Variable Prozeßdaten)	schreiben (14 Byte) ¹⁾ Gerätesteuerung, (40 variable Prozeßdaten)	mit Parameterkanal

Der Parameterkanal dient zur sequentiellen Übertragung von Parameter- und Konfigurationsdaten. Die nachfolgenden Tabellen zeigen die einzustellenden Werte und Datenbezeichnungen an:

Für die Prozeßdatenmodule (Modul E + F) müssen, mit Hilfe des Engineering-Tools, die Daten der zyklischen Übertragung über allgemeine Geräteinstellungen r Kommunikation r Busdaten ausgewählt werden

Fig.: 2 Zuordnung der Gerätedaten für den Feldbus



Es können maximal 64 Daten zum Lesen und 64 Daten zum Schreiben ausgewählt werden. Je nach verwendetem Prozeßdatenmodul werden die ersten 90 Daten (Modul E), und die ersten 14 Daten (Modul F), verwendet.

1) Anzahl der benötigten Bytes im E/A-Feld

q **Modul A** (Prozeßdaten aller 16 Kanäle + Parameterkanal - Messwerverfassung)

Nr.	Bez.	L/S	FIX Point-Format			Bem.
			Anzahl Byte	Wert		
				Hex	COM PROFIBUS	
Eingänge			156/58			
0	Xeff_1	L	2	50	1AE	
1	Alarm_1	L	1	10	8DE	B
2	Xeff_2	L	2	50	1AE	
3	Alarm_2	L	1	10	8DE	B
...						
30	Xeff_16	L	2	50	1AE	
31	Alarm_16	L	1	10	8DE	B
Ein-/Ausgänge						
32	Parameterkanal	L/S	8/8	F3	4AX	

q **Modul B** (Prozeßdaten aller 16 Kanäle + Parameterkanal - Standardregler)

Nr.	Bez.	L/S	FIX Point-Format			Bem.
			Anzahl Byte	Wert		
				Hex	COM PROFIBUS	
Eingänge			106			
0	Unit_State	L	2	11	16DE	A
1	Xeff_1, Yeff_1, Alarm_1, Status1	L	6	52	3AE	B C
2	Xeff_2, Yeff_2, Alarm_2, Status_2	L	6	52	3AE	B C
...						
16	Xeff_16, Yeff_16, Alarm_16, Status_16	L	6	52	3AE	B C
Ausgänge			106			
17	Unit_Cntr	S	2	21	16DA	D
18	Wvol_1, Yman_1, Cntrl_1	S	6	62	3AA	E
19	Wvol_2, Yman_2, Cntrl_2	S	6	62	3AA	E
...						
34	Wvol_16, Yman_16, Cntrl_16	S	6	62	3AA	E
Ein- /Ausgänge						
35	Parameterkanal	L/S	8 / 8	F3	4AX	

q **Modul C** (Nur Parameterkanal)

Nr.	Bez.	L/S	FIX Point-Format		
			Anzahl Byte	Wert	
				Hex	COM PROFIBUS
Ein- /Ausgänge					
0	Parameterkanal	L/S	8 / 8	F3	4AX

Die Übertragung der Analogwerte erfolgt im 16 Bit Festpunkt Format (FIX). Im FIX-Format werden alle Werte mit einer Nachkommastelle interpretiert (Wertebereich -3000,0 bis 3200,0).

q **Modul D (50 Variable Prozeßdaten und Parameterkanal)**

Nr.	Bez.	L/S	FIX Point-Format			Bem.
			Anzahl Byte	Wert		
				Hex	COM PROFIBUS	
Eingänge			114			
0	Unit_State, Digital Outputs	L	6	15	16DE	A, F
1	IN_1 ... IN_8	L	16	57	8AE	
2	IN_9 ... IN_16	L	16	57	8AE	
...						
6	IN_41 ... IN_48	L	16	57	8AE	
7	IN_49 ... IN_50	L	4	51	4AE	
Ausgänge			114			
8	Unit_Cntrl	S	2	21	16DA	D
9	OUT_1 ... OUT_8	S	16	67	8AA	
10	OUT_9 ... OUT_16	S	16	67	8AA	
...						
14	OUT_41 ... OUT_48	S	16	67	8AA	
15	OUT_49 ... OUT_50	S	8	61	4AA	
Ein- /Ausgänge						
16	Parameterkanal	L/S	8 / 8	F3	4AX	

q **Modul E (40 Variable Prozeßdaten und Parameterkanal)**

Nr.	Bez.	L/S	FIX Point-Format			Bem.
			Anzahl Byte	Wert		
				Hex	COM PROFIBUS	
Eingänge			94			
0	Unit_State, Digital Outputs	L	6	15	4DE	A, F
1	IN_1 ... IN_8	L	16	57	8AE	
2	IN_9 ... IN_16	L	16	57	8AE	
...						
5	IN_33 ... IN_40	L	16	57	8AE	
Ausgänge			90			
6	Unit_Cntrl	S	2	21	16DA	D
7	OUT_1 ... OUT_8	S	16	67	8AA	
8	OUT_9 ... OUT_16	S	16	67	8AA	
...						
11	OUT_33 ... OUT_40	S	16	67	8AA	
Ein- /Ausgänge						
16	Parameterkanal	L/S	8 / 8	F3	4AX	

q **Modul F (Multiplexen aller 64 variablen Prozeßdaten und Parameterkanal)**

Nr.	Bez.	L/S	FIX Point-Format			Bem.
			Anzahl Byte	Wert		
				Hex	COM PROFIBUS	
Eingänge			18			
0	Unit_State, Digital Outputs	L	6	15	16DE	A, F
1	Index IN	Read	2	50	1AE	
2						
2	Read Value	L	2	50	1AE	
Ausgänge			14			
3	Unit_State	S	2	21	32DE	A
4	Index IN	Read	2	60	1AA	
5						
5	Read Value	S	2	60	1AA	
Ein-/Ausgänge						
6	Parameterkanal	L/S	8 / 8	F3	4AX	

Funktionsweise (lesen):

- W Eintrag der Indexnummer in 'Index OUT' (Read)
- W nachdem die Indexnummer in 'Index IN' (Read) gespiegelt erscheint, steht der gelesene Wert in 'Read Value'.

Funktionsweise (schreiben):

- W Eintrag der Indexnummer in 'Index OUT' (Write)
- W Eintrag des zu schreibenden Wertes in 'Write Value'
- W nachdem die Indexnummer in 'Index IN' (Write) gespiegelt erscheint, wurde der Wert übertragen.

g Um eine konsistente Datenübertragung zu gewährleisten, muß sichergestellt sein, daß 'Index OUT' (Write) und 'Write Value' vor einem PROFIBUS - Datenzyklus aktualisiert sind. Kann dies nicht sichergestellt werden, ist folgendermaßen vorzugehen: '0' in 'Index OUT' (Write) dann den zu übertragenden Wert in 'Write Value' schreiben und anschließend die Indexnummer in 'Index OUT' (Write). Eintrag einer '0' in 'Index OUT' (Read) / 'Index OUT' (Write) bewirkt keine Datenübertragung.

3.1 Als Statusbyte sind definiert:

Bem. A Unit_State

MSB							LSB	
D15	D14	D13	D2	D1	D0	

Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'
D0...D3		immer "0"		
D4		immer '0'		
D5	Dex	geänderte ComRead od. ComWrite Daten ¹⁾	nein	ja
D6, D7		immer '0'		
D8	Err1	Vorgabefehler Kanal 1 oder 9	nein	ja
D9	Err2	Vorgabefehler Kanal 2 oder 10	nein	ja
D10	Err3	Vorgabefehler Kanal 3 oder 11	nein	ja
D11	Err4	Vorgabefehler Kanal 4 oder 12	nein	ja
D12	Err5	Vorgabefehler Kanal 5 oder 13	nein	ja
D13	Err6	Vorgabefehler Kanal 6 oder 14	nein	ja
D14	Err7	Vorgabefehler Kanal 7 oder 15	nein	ja
D15	Err8	Vorgabefehler Kanal 8 oder 16	nein	ja

Bem. B Alarm_x

MSB					LSB			
D7	D6	D5	D4	D3	D2	D1	D0	

Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'
D0	Lim HH	Alarm HH	aus	ein
D1	Lim H	Alarm H	aus	ein
D2	Lim L	Alarm L	aus	ein
D3	Lim LL	Alarm LL	aus	ein
D4	Fail	Alarm Sensor Fail	nein	ja
D5		immer '0'		
D6		immer '0'		
D7		immer '0'		

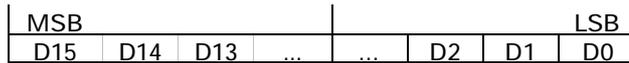
Bem. C Status_x

MSB							LSB
D7	D6	D5	D4	D3	D2	D1	D0

Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'
D0	w/w2	w/w2 Umschaltung	w	w2
D1	We/w	extern-/intern-Umschaltung	extern	intern
D2	w/Wanf	Umschaltung Anfahrswert	w	Wanf
D3	Orun	Optimierung aktiv	nein	ja
D4	A/M	Automatik/Hand-Umschaltung	Auto	Hand
D5	Coff	Regler abgeschaltet	nein	ja
D6	Y1	Schaltausgang 1	aus	ein
D7	Y2	Schaltausgang 2	aus	ein

1) Siehe Kapitel 3.3 Seite 14 'Verriegelungsmechanismus bei Änderungen'.

Bem. D Unit_Contrl



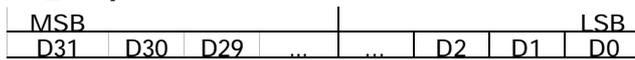
Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'
D0 ... D2		immer '0'		
D3	OStartG	Starten der Selbstoptim. aller Gruppenregler ¹⁾	kein Start	Start
D4	OStopG	Stoppen der Selbstoptim. aller Gruppenregler ¹⁾	kein Stop	Stop
D5	Dval	Daten gültig, Quitting ²⁾	Flanke '0' → '1'	
D6 .. D15		immer '0'		

Bem. E Cntrl_x



Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'
D0	A/M	Automatik/Hand-Umschaltung	Auto	Hand
D1	Coff	Reglerabgeschaltet	nein	ja
D2	w/w2	w/w2Umschaltung	w	w2
D3	We/w	extern-/intern-Umschaltung	extern	intern
D4	OStart	Optimierung starten ³⁾	kein Starten	starten
D5	OStop	Optimierung stoppen ¹⁾	kein Stoppen	stoppen
D6 .. D15		nicht benutzt, immer '0'		

Bem. F Digital_Outputs



Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'
D0	Y1_15	Y1-Ausgang des Kanals 15	aus	ein
D1	Y2_15	Y2-Ausgang des Kanals 15	aus	ein
D2	Y1_14	Y1-Ausgang des Kanals 14	aus	ein
D3	Y2_14	Y2-Ausgang des Kanals 14	aus	ein
D4	Y1_13	Y1-Ausgang des Kanals 13	aus	ein
D5	Y2_13	Y2-Ausgang des Kanals 13	aus	ein
D6	Y1_12	Y1-Ausgang des Kanals 12	aus	ein
D7	Y2_12	Y2-Ausgang des Kanals 12	aus	ein
D8	Y1_11	Y1-Ausgang des Kanals 11	aus	ein
D9	Y2_11	Y2-Ausgang des Kanals 11	aus	ein
D10	Y1_10	Y1-Ausgang des Kanals 10	aus	ein
D11	Y2_10	Y2-Ausgang des Kanals 10	aus	ein
D12	Y1_9	Y1-Ausgang des Kanals 9	aus	ein
D13	Y2_9	Y2-Ausgang des Kanals 9	aus	ein
D14	Y1_8	Y1-Ausgang des Kanals 8	aus	ein
D15	Y2_8	Y2-Ausgang des Kanals 8	aus	ein
D16	Y1_7	Y1-Ausgang des Kanals 7	aus	ein
D17	Y2_7	Y2-Ausgang des Kanals 7	aus	ein
D18	Y1_6	Y1-Ausgang des Kanals 6	aus	ein
D19	Y2_6	Y2-Ausgang des Kanals 6	aus	ein
D20	Y1_5	Y1-Ausgang des Kanals 5	aus	ein
D21	Y2_5	Y2-Ausgang des Kanals 5	aus	ein
D22	Y1_4	Y1-Ausgang des Kanals 4	aus	ein
D23	Y2_4	Y2-Ausgang des Kanals 4	aus	ein
D24	Y1_3	Y1-Ausgang des Kanals 3	aus	ein
D25	Y2_3	Y2-Ausgang des Kanals 3	aus	ein
D26	Y1_2	Y1-Ausgang des Kanals 2	aus	ein
D27	Y2_2	Y2-Ausgang des Kanals 2	aus	ein
D28	Y1_1	Y1-Ausgang des Kanals 1	aus	ein
D29	Y2_1	Y2-Ausgang des Kanals 1	aus	ein
D30	Y1_0	Y1-Ausgang des Kanals 0	aus	ein
D31	Y2_0	Y2-Ausgang des Kanals 0	aus	ein

1) Signale werden nur bei Wechsel von 0 → 1 aktiv. Das Signal muß so lange an stehen, bis ein Wechsel von 1 → 0 (siehe Status_x) stattgefunden hat.

2)

3)

3.2 Status und Diagnosemeldungen

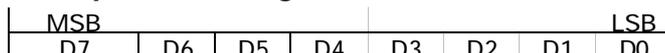
Zur Signalisierung von KS 816-Gerätezuständen ist die externe (anwenderspezifische) Diagnose zu verwenden. Das Format entspricht der gerätebezogenen Diagnose (EN50170 Volume 2 PROFIBUS)

Gerätespezifische Diagnose Octet 1



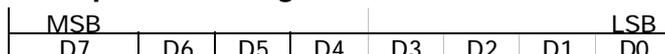
Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'	Typ
D0	Online/Conf	Online/Konfiguration	Online	Konfiguration	Status
D1 .. D7		immer '0'			

Gerätespezifische Diagnose Octet 2



Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'	Typ
D0	InpF1	Input Fail Kanal 1	nein	ja	Diagnose
D1	InpF2	Input Fail Kanal 2	nein	ja	Diagnose
D2	InpF3	Input Fail Kanal 3	nein	ja	Diagnose
D3	InpF4	Input Fail Kanal 4	nein	ja	Diagnose
D4	InpF5	Input Fail Kanal 5	nein	ja	Diagnose
D5	InpF6	Input Fail Kanal 6	nein	ja	Diagnose
D6	InpF7	Input Fail Kanal 7	nein	ja	Diagnose
D7	InpF8	Input Fail Kanal 8	nein	ja	Diagnose

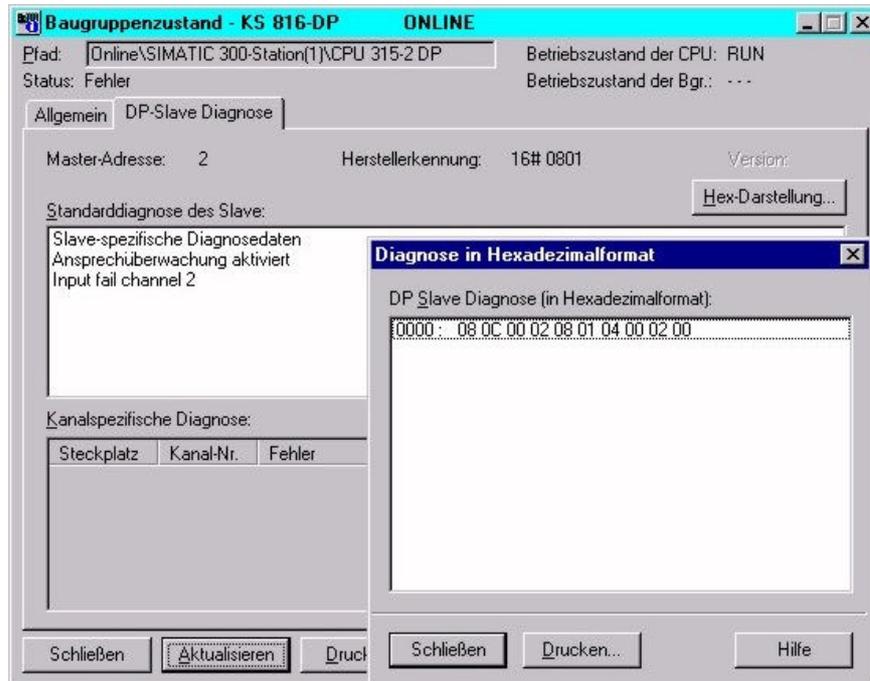
Gerätespezifische Diagnose Octet 3



Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'	Typ
D0	InpF9	Input Fail Kanal 9	nein	ja	Diagnose
D1	InpF10	Input Fail Kanal 10	nein	ja	Diagnose
D2	InpF11	Input Fail Kanal 11	nein	ja	Diagnose
D3	InpF12	Input Fail Kanal 12	nein	ja	Diagnose
D4	InpF13	Input Fail Kanal 13	nein	ja	Diagnose
D5	InpF14	Input Fail Kanal 14	nein	ja	Diagnose
D6	InpF15	Input Fail Kanal 15	nein	ja	Diagnose
D7	InpF16	Input Fail Kanal 16	nein	ja	Diagnose

Darstellung der Slave-Diagnose in STEP 7

Das folgende Bild zeigt den Baugruppenzustand des KS 816 und zusätzlich die Diagnoseinformation im Hexadezimalformat.



3.3 Verriegelungsmechanismus bei Änderungen

Wird während des Betriebes, z.B. online über Parameterkanal oder über die Engineering-schnittstelle, die Referenz auf ein zu übertragendes Datum geändert, so besteht die große Gefahr, daß Werte sowohl beim Busmaster als auch bei KS 816 misinterpretiert werden können. Dem soll ein Verriegelungsmechanismus abhelfen.

- W Wird eine Referenz verändert, setzt das Reglermodul das Bit Dex = 1.
- W Der Master muß das Bit Dex auswerfen
- W Quittierung und die Aussage, daß auch auf Masterseite nun gültige Schreibdaten vorliegen, werden über eine positive Flanke für das Bit Dval erzeugt.
- W Beim Empfang einer positiven Flanke setzt das Reglermodul Dex = 0 und übernimmt die gesendete Daten.
- W Das Rücksetzen Dex kann auch durch Aus- und Einschalten der Spannung erfolgen.

3.4 Übertragung von Prozeßdaten

An den KS 816 gesendete Ausgangsdaten werden mit den vorher gesendeten Werten verglichen und bei Abweichung vom Regler verarbeitet. Ist eine der Daten fehlerhaft, so wird im 'Unit_State' das Bit 8 bei Fehler im Kanal 1 oder 9, Bit 9 bei Fehler im Kanal 2 oder 10 ... oder Bit 15 bei Fehler im Kanal 8 oder 16 gesetzt, so lange bis keine fehlerhaften Zugriffe mehr anstehen.

3.5 Übertragung von Parametern

Für die Übertragung von Parametern steht der 'Parameterkanal' zu Verfügung, über den Daten transparent über das Funktionsblockprotokoll ausgetauscht werden können. Dabei werden alle möglichen Zugriffsarten des Protokolls unterstützt (Einzelzugriff, Zehnerblock und Gesamtblock). Die Kommunikation zum Regler erfolgt transparent, d.h. der Anwender ist für die Überwachung der Wertebereiche, Betriebsarten (auto/hand) usw. selbst verantwortlich. Der Parameterkanal ist für große Datenmengen mit geringen Anforderungen an die Übertragungsgeschwindigkeit ausgelegt.



3.5.1 Nachrichtenelemente

Im Folgenden werden einige Begriffe verwendet, die hier erläutert werden sollen:

Element	Beschreibung	Bem.
ID	Kennzeichnung der Telegrammart	A
ID1	Datenformat der zu übertragenden bzw. empfangenen Daten	B
Code	Adressierungsschlüssel einer Date	C
FB-Nr.	Funktionsblocknummer	D
Fkt-Nr.	Funktionsnummer	E
Type	d.c. (immer '0')	

Bem. A ID

Dieses Element identifiziert die Telegrammart:

- ID = 0x10 = Starttelegramm
- ID = 0x68 = Datentelegramm
- ID = 0x16 = Endtelegramm

Bem. B ID1

Dieses Element identifiziert das Dateiformat:

- ID1 = 0 = Integer
- ID1 = 1 = Realwert als Fixpoint

Bem. C Code

Die-Code-Kennung ist Dezimal und der Wertebereich umfaßt '00'...'99' sowie '178' = B2 und '179' = B3.

Bem. D FB-Nr. (Funktionsblocknummer)

Ein Funktionsblock wird mit einer Funktionsblocknummer adressiert. Sie umfaßt den Wertebereich '0' bis '250'. Über die Funktionsblocknummer erfolgt auch die Kanaladressierung. Funktionsblocknummernbereiche:

- W 0 - allgemeine Daten für das gesamte Gerät
- W 1 - 99 fest eingerichtete Funktionsblöcke

Bem. E Fkt-Nr. (Funktionsnummer)

Eine Funktion als Teiladresse eines Funktionsblock wird eben falls mit einer Funktionsnummer angesprochen. Sie umfaßt den Wertebereich '0' bis '99'.

Funktionsnummernbereiche:

- W 0 Funktion Allgemein
- W 1 - 99 andere Funktionen

1) 0x10 bedeutet 10 in Hexadezimal

3.5.2 Allgemeiner Übertragungsaufbau

Um über ein Datenfenster von 8 Byte, die für das Funktionsblockprotokoll benötigten Parameter übertragen zu können, besteht der Zugriff aus drei Teilen:

- ⊞ Auftragsheader mit Angabe des Codes, der FB-Nr., der Fkt-Nr., des Types sowie der folgenden Real- und Integer-Werte.

Aufbau des Starttelegramms:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
ID	ID1	Code	FB-Nr.	Fkt._Nr.	Type	Anz. Realwerte	Anz. Integerwerte

- ⊞ n Datenblöcke mit den zu übertragenden Nutzdaten

Aufbau des Datentelegramms:

Übertragung von Realdaten als Fixedpoint und von Integerwerten

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
ID	Count					Integer	

- ⊞ ein Endblock, liefert das Ergebnis der Operation

Aufbau des Endetelegramms:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
ID		Result					

Bedeutung von Result	
0	OK
4	NAK

Das Lesen oder Schreiben wird immer vom Master eingeleitet. Wenn die Anzahl der Real- und Integerwerte $\neq 0$ sind, so wird ein Write Dienst, sonst ein Read Dienst ausgelöst.

Der Code bestimmt die Zugriffsart:

- Code < 100, kein Vielfaches von 10 ⊞ Einzelzugriff
- Code < 100, Vielfaches von 10 ⊞ Zehnerblock Zugriff
- Code > 100 ⊞ Blockzugriff Gesamtblock

3.5.3 Ablauf Schreiben der Daten

Starttelegramm:

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Master sendet:	0x10	ID1	Code	FB-Nr.	Fkt._Nr	Type	Anzahl Realwerte	Anzahl Integerwerte
Regler antwortet:	0x10							

Datentelegramme:

Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
	0x68	count			Value		0x68	count			

Dabei wird bei Count = 1 der erste Wert gesendet, zur Flußkontrolle wird Count vom KS 816 gespiegelt (? 1 mal).

Die Werte werden in der Reihenfolge Real Integer übertragen.

Endetelegramm:

Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2-3	Byte 4 - 7
	0x16						0x16		Result	

3.5.4 Ablauf Lesen der Daten

Starttelegramm:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
sendet:	0x10	ID1	Code	FB-Nr.	Fkt._Nr	Type	0	0
Regler	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
antwortet:	0x10						Anzahl Realwerte	Anzahl Integerwerte

Datentelegramme:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
sendet:	0x68	count				antwortet:	0x68	count			Value

Dabei wird bei Count = 1 der erste Wert gesendet, zur Flußkontrolle wird Count vom KS 816 gespiegelt (? 1 mal). Die Werte werden in der Reihenfolge RealInteger übertragen.

Endetelegramm:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler	Byte 0	Byte 1	Byte 2-3	Byte 4 - 7
sendet:	0x16					antwortet:	0x16		Result	

3.6 Beispiele

3.6.1 Prinzipien des Funktionsblock-Protokolls

Ein Funktionsblock besitzt Ein- und Ausgangsdaten (Prozeßdaten) sowie Parameter und Konfigurationsdaten. Er ist adressierbar über eine Funktionsblocknummer.

Es werden folgende Zugriffsmechanismen unterschieden:

3.6.2 Einzelzugriff

Mit diesem Zugriff (Code xx) kann ein einzelner Wert einer Funktion gelesen bzw. geschrieben werden.

Gültige Werte für ID1:

Konfiguration als FixPoint:	0 = Integer	Realwerte werden als Integer (ohne Nachkommastelle) übertragen
	1 = Real	Realwerte werden als FixPoint (1 Nachkommastelle) übertragen

Beispiel 1: (Nachrichtenaufbau bei Datenvorgabe)

Übertragung der Parametersatznummer (ParNr = 1) zum Regler (Kanal 2).

Starttelegramm:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
sendet:	0x10	0	31	52	5	0	0	1
Regler	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
antwortet:	0x10							

Datentelegramme:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
sendet:	0x68	1			1	antwortet:	0x68	1			

Endetelegramm

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler	Byte 0	Byte 1	Byte 2 - 3	Byte 4 - 7
sendet:	0x16					antwortet:	0x16		0	

1) wurde ein Read Dienst abgewiesen, so sind diese Werte = 0

Beispiel 2: (Nachrichtenaufbau bei Datenanforderung)

Lesen des Fehlercodes der Selbstoptimierung Heizen (MSG1) vom Regler (Kanal 2).
Starttelegramm:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
sendet:	0x10	0	35	52	5	0	0	0
Regler	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
antwortet:	0x10						0	1

Datentelegramme:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
sendet:	0x68	1				antwortet:	0x68	1			2 (ok)

Endetelegramm:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler	Byte 0	Byte 1	Byte 2 - 3	Byte 4 - 7
sendet:	0x16					antwortet:	0x16		0	

3.6.3 Blockzugriff (Zehner-Block)

Mit diesem Zugriff (Code x0) können maximal neun Prozeßwerte (im mer als REAL-Werte) einer Funktion gelesen werden.

Beispiel: (Nachrichtenaufbau bei Datenanforderung)

Lesen der Sollwerte (Wnvol und wvol) vom Regler (Kanal3).
Starttelegramm:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
sendet:	0x10	0	30	53	1	0	0	0
Regler	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
antwortet:	0x10						2	0

Datentelegramme:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Rengler	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
sendet:	0x68	1				antwortet:	0x68	1			150
Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
sendet:	0x68	2				antworte:	0x68	2			140

Endetelegramm:

Master	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler	Byte 0	Byte 1	Byte 2 - 3	Byte 4 - 7
sendet:	0x16					antwort:	0x16		0	

3.6.4 Blockzugriff (Gesamt-Block)

Mit diesem Zugriff können alle Parameter- (Code 178) und Konfigurationsdaten (Code 179) einer Funktion gelesen bzw. geschrieben werden. Für diesen Zugriff gelten folgende Bedingungen:

- ⊘ **Um Daten mit 'Code B3 = 179'** schreiben zu können, muß vor her das Gerät in den Konfigurationsmodus (r siehe Seite 24 'OpMod') geschaltet werden. Wirksam werden alle neu eingegebenen Konfigurationsdaten und Parameter erst, wenn das Gerät wieder auf online zurückgeschaltet wurde.
- ⊘ Alle Daten einer Nachricht müssen definiert sein, Auslassungen sind nicht zulässig.
- ⊘ Sind Teile einer Nachricht im Gerät nicht in Benutzung (HW- und SW-Optionen), so ist dennoch die komplette Nachricht zu übertragen. Die Prüfung der nicht vorhandenen Daten entfällt.
- ⊘ Bei fehlerhaften Blockschreibzugriffen gilt: Eine Nachricht wird mit NAK beantwortet, wenn mindestens ein Datum fehlerhaft ist. Bereits gültige Werte werden übernommen.

Im folgenden ist anhand zweier Beispiele der Nachrichtenaufbau bei Blockzugriffen mit Code B2/B3 dargestellt. Die Reihenfolge der zu übertragenen Daten ist der jeweiligen Code-Tabelle zu entnehmen.

Gültige Werte für ID1:

Konfigurations als Fix-Point:	0, 1	Übertragung Realwerte jeweils als Fix-Point-Wert
-------------------------------	------	--

Beispiel 1: (Nachrichtenaufbau bei Datenanforderung)

Lesen der Sollwertparameter (~~W0~~, ~~W100~~, ~~W2~~, ~~Grw+~~, ~~Grw-~~ und ~~Grw2~~) vom Regler (Kanal 7).
Starttelegramm:

Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	0x10	0	0xB2	57	1	0	0	0
Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	0x10						6	0

Datentelegramme:

Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
	0x68	1					0x68	1			0
Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
	0x68	2					0x68	2			700
Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
	0x68	3					0x68	3			100
Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
	0x68	4					0x68	4			-32000
Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
	0x68	5					0x68	5			-32000
Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
	0x68	6					0x68	6			-32000

Endetelegramm:

Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2 - 3	Byte 4 - 7
	0x16						0x16		0	

Beispiel 2: (Nachrichtenaufbau bei Datenvorgabe)

Schreiben der Alarmkonfiguration (C600, C601) zum Regler (Kanal 1).

Starttelegramm:

Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	0x10	0	0xB3	70	0	0	0	2
Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	0x10						0	0

Datentelegramme:

Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
	0x68	1			0120		0x68	1			
Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7
	0x68	2			0110		0x68	2			

Endetelegramm:

Master sendet:	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 - 7	Regler antwortet:	Byte 0	Byte 1	Byte 2 - 3	Byte 4 - 7
	0x16						0x16		0	

3.7 Datentypen

Werte von Daten werden für die Übertragung in Datentypen gegliedert.

- W FP
Wert im Gerät als Floating Point Zahl (Real) vorhanden
Wertebereich: als Integer (im Einzelzugriff) -9999 ... 0 ... 9999
als Fixpoint -3000,0 ... 0,0 ... 3200,0
Ausnahme: Abschaltwert '-32000'
- W INT
positive ganze Integer-Zahl
Wertebereich: 0 ... 32767
Wertebereich bei Konfigurationsworten: 0000 ... 9999 (r Seite 24)
Ausnahme: Abschaltwert '-32000'
- W ST1
Status, bit-orientiert, 1 Byte Länge
Wertebereich: 00H ... 3FH, übertragen: 40H...7FH
Es können nur 6 Bits für die Informationsübertragung genutzt werden, nämlich Bit 0...5 (LSB = Bit 0). Bit 6 muß immer auf '1' gesetzt sein, um Verwechslungen mit den Steuerzeichen zu vermeiden. Bit 7 enthält das Parity Bit.
- W **ICMP (Integer Compact)**
Bitinformationen als Integerübertragung, max. 15 Bits
Wertebereich: 0...32767; Integerübertragung erfolgt im ASCII-Format.

	fest auf '0'	Bedeutung der Bits														
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wert	-	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

Beispiel:

Bit 13 = 1 und Bit 1 = 1, alle übrigen Bits sind '0'

interner Hex-Wert: 0x2002, als Integerwert: 8194, übertragenen ASCII-Wert: '8194'

4 Schnelleinstieg mit S7

Auf der dem Engineering Set bei lie gen den Dis ket te be fin det sich die GSD-Datei, Bei spie pro jekte für eine **SIMATIC® S5 / S7**, die Typ-Datei sowie Beispielkonfigurationen für COM PROFIBUS. Mit Hilfe der Konfiguration und des Projektes kann auf einfache Weise eine Kom munika tion mit einem KS 816-DP aufgebaut werden.

Testumgebung

Für den Testaufbau benötigen Sie folgende Komponenten:

- W Programmiergerät (empfohlen PG740)
- W Automatisierungsgerät
 - CPU315-2 DP
- W KS 816-DP
- W Engineering Set (Bestell Nr. 9407 999 09x11)
- W Kabel
 - PROFIBUS Kabel AG | KS 816-DP
 - PG | AG

4.1 Beispiel einer Testumgebung:

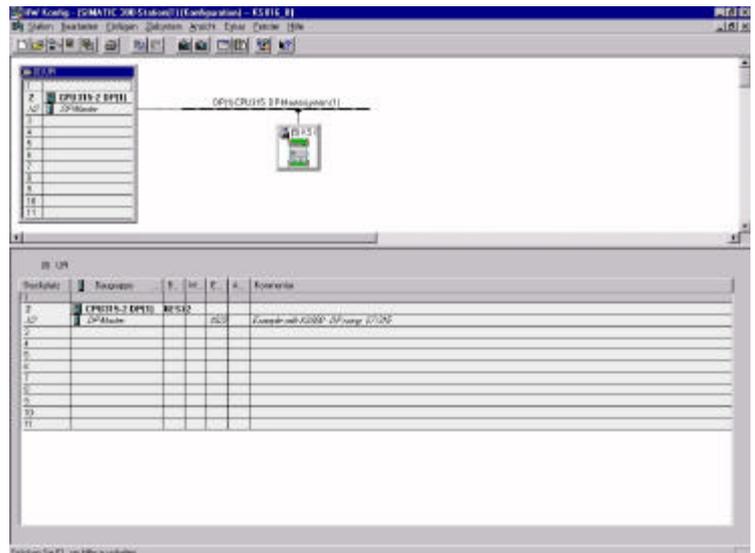
Ein KS 816-DP mit der Adresse 5 soll an eine CPU315-2 DP über PROFIBUS-DP angeschlossen werden. Es wird das Pro zeß da ten mo dul B ge wählt (16 Pro zeß da ten ka nä le und Pa ra me terkanal). Daten sollen in Fix Point-Format übertragen werden.

g

Bevor die Testumgebung in Betrieb genommen wird, sollten Sie sicherstellen, daß die Automatisierungsgeräte keine Anwendersoftware enthalten ("Urgelöscht").

Vorgehensweise:

- W Herstellender Verbindungen
- W Konfigurieren der Geräte
 - Am KS 816-DP die Adresse 5 einstellen (über Codierschalter oder Engineering Tool) und an Netz anschließen.
 - Busabschlußwiderstände am Reglerstecker und am Stecker der SPS (S7) aktivieren.
- W PROFIBUS-Netzkonfiguration
 - Diskette (Engineering Set) in PG stecken.
 - Beispielprojekt dearchivieren (A:\KS 816DP\S7_FB \EXAMPLE \KS 816dmo.arj)
 - Projekt KS 816dmo öffnen
 - Adressierungen und CPU Hardwarekonfiguration gegebenenfalls anpassen und in den DP-Master (CPU315-2 DP) übertragen.
 - AG auf Run schalten.



Nach Inbetriebnahme des Testaufbaus kann mit Hilfe der dem Projekt beigefügten Variablen ta bel len (VAT x) ein Test des E/A-Bereichs und der Auf ruf des Pa ra me terka nals durch ge führt werden.

VAT 1:

Hier werden die Prozeß- daten aller Kanäle dargestellt (Fix-Point). Im nebenstehenden Bild ist nur der Kanal 1 zu sehen.

Beispiel Kanal 1:

(Sollwertvorgabe = 30
Stellgröße = 40 %
Handbetrieb)

Operand	Symbol	Statuswert	Steuwert
PEW 0	"Unit_State"	2#0000_0000_0000_0000	
PEW 256	"Yeff_1"	226	
PEW 258	"Yeff_1"	76	
PEW 260	"Alarm_1"	2#0000_1100	
PEW 262	"Status_1"	2#0000_0010	
PEW 264	"Yeff_2"	226	
PEW 266	"Yeff_2"	0	
PEW 268	"Alarm_2"	2#0000_1100	
PEW 270	"Status_2"	2#0000_0010	
PEW 272	"Yeff_3"	227	
PEW 274	"Yeff_3"	0	
PEW 276	"Alarm_3"	2#0000_1100	
PEW 278	"Status_3"	2#0000_0010	
PEW 280	"Yeff_4"	227	
PEW 282	"Yeff_4"	0	
PEW 284	"Alarm_4"	2#0000_1100	
PEW 286	"Status_4"	2#0000_0010	
PEW 288	"Yeff_5"	227	

VAT 2:

Mit Hilfe dieser Variablen-Tabelle kann auf die Parameter des Funktionsbausteins zur Abbildung des Parameterkanals zugegriffen werden.

Vorzugeben sind z. B. beim Lesen von Fixpointwerten:

- W CodeNo, FBNo, FKTNo, Typ = 0 (r Kapitel)
- W Service = 0x 0001
- W Start_FixP = 1
- W ANZW_FixP zeigt den Status und das Ergebnis nach Abschluß der FB-Bearbeitung an.
- W DWLR, DWLI, zeigen die Anzahl der gelesenen Werte.

Operand	Symbol	Statuswert	Steuwert
MW 100	"Service"	16#16#0000	16#16#0000
MW 102	"CodeNo"	0	9
MW 104	"FBNo"	0	157
MW 106	"FKTNo"	0	0
MW 108	"Typ"	0	0
MW 110	"DWLR"	0	
MW 112	"DWLI"	0	
MW 114	"DWLC"	0	
MW 120	"ANZW_FixP"	2#0000_0000_0000_0000	
M 0.0	"Start_FixP"	2#0	2#1
M 121.4	"Reset"	2#0	//2#1
DB37.DBW 0	---	0	//300
DB37.DBW 2	---	0	
DB37.DBW 4	---	0	

Im unteren Bild sieht man die ersten Daten eines Datenbausteins, in den Daten des Parameterkanals geschrieben werden bzw. aus dem Werte gelesen werden.

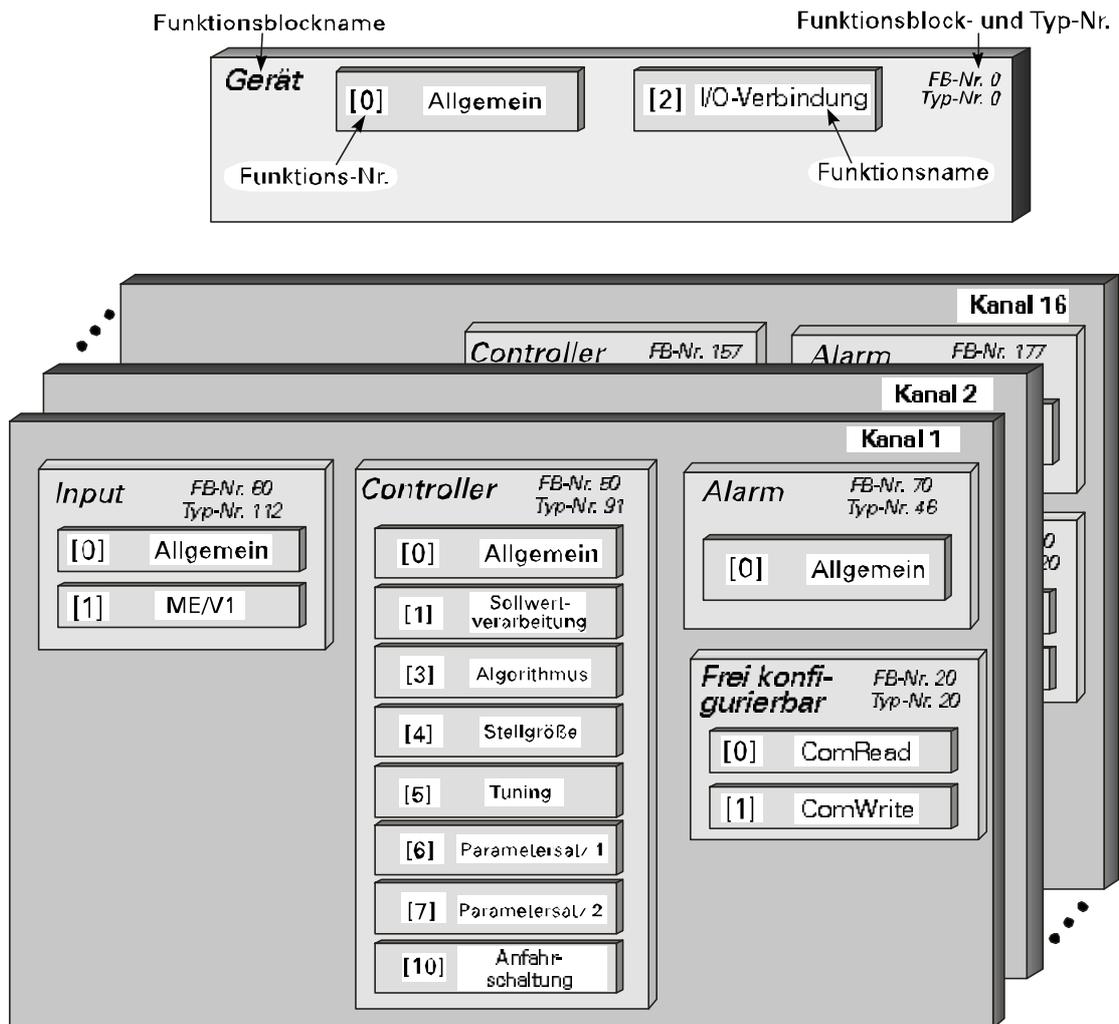
Operand	Symbol	Statuswert	Steuwert
PAW 0	"Unit_Cntrl"	Kein Statuswert vorhanden!	//2#0000_0000_0000_0000
PAW 256	"Yvol_1"	Kein Statuswert vorhanden!	//1000
PAW 258	"Yvol_1"	Kein Statuswert vorhanden!	//0
PAW 260	"Cntrl_1"	Kein Statuswert vorhanden!	//2#0000_0000_0000_0000
PAW 262	"Yvol_2"	Kein Statuswert vorhanden!	//1000
PAW 264	"Yvol_2"	Kein Statuswert vorhanden!	//0
PAW 266	"Cntrl_2"	Kein Statuswert vorhanden!	//2#0000_0000_0000_0000
PAW 268	"Yvol_3"	Kein Statuswert vorhanden!	//1000
PAW 270	"Yvol_3"	Kein Statuswert vorhanden!	//0
PAW 272	"Cntrl_3"	Kein Statuswert vorhanden!	//2#0000_0000_0000_0000
PAW 274	"Yvol_4"	Kein Statuswert vorhanden!	//1000
PAW 276	"Yvol_4"	Kein Statuswert vorhanden!	//0
PAW 278	"Cntrl_4"	Kein Statuswert vorhanden!	//2#0000_0000_0000_0000
PAW 280	"Yvol_5"	Kein Statuswert vorhanden!	//1000
PAW 282	"Yvol_5"	Kein Statuswert vorhanden!	//0

5 Funktionsblock-Protokoll

5.1 Datenstrukturierung

Durch die Vielfalt der zu verarbeitenden Informationen in KS 816 sind logisch zusammenhängende Daten und Aktionen zu Funktionsblöcken zusammengefasst. Ein Funktionsblock besitzt Ein-, Ausgangsdaten, Parameter und Konfigurationsdaten. Sie werden über feste Blockadressen (FB-Nr.) angesprochen. Jeder Block ist wiederum in einzelne Funktionen aufgeteilt. Funktionen werden über Funktionsnummern (Fkt-Nr.) angesprochen. Funktionsnummer 0 adressiert funktionsblockspezifische Daten.

Fig.: 3 Übersicht der Funktionsblöcke und Funktionen des KS 816



5.2 CODE-Tabellen

5.2.1 Aufbau Konfigurationsworte (C.xxxx)

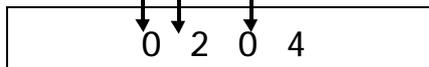
Die in den folgenden Code-Tabellen aufgeführten Konfigurationsworte bestehen aus mehreren Teilkomponenten, die nur gemeinsam übertragen werden können. Die Daten in der Tabelle sind folgendermaßen zu interpretieren:

Beispiel (C100):

Code	Bez.	L/S	Typ	Beschreibung	Bereich
71	C100	L/S	INT	CFunc: Reglerfunktion (T,H) WFunc: Sollwertfunktion (E)	0..xx0z

Beschreibung	CFunc		WFunc
	Tausender	Hunderter	Einer
Bereich	x	x	z
	00 ... 07		0...1

Beispiel: 2-Pkt- Regler;
Festwert/ Folge



- g - Die Einstellmöglichkeiten der Konfigurationswörter entnehmen Sie der KS 816 Funktionsbeschreibung (Best. Nr.: 9499 040 55918)

5.2.2 GERÄT

(FB-Nr.: 0 Typ-Nr.: 0)

In dem Funktionsblock 'GERÄT' sind alle Daten, die für das gesamte Gerät gelten, zusammengefaßt.

Prozeßdaten

(Funktions-Nr: 0)

Allgemein					Bereich	Be m.
Code	Bez.	L/S	Typ	Beschreibung		
01	Unit_State1	L	ST1	Status1		A
10	Block 13..15, 18	L	Block			
13	WriteError	L	INT	Fehler des letzten Schreibzugriffs	0, 100... 127	
14	WriteErrorPosition	L	INT	Position des letzten Schreibzugriffsfahlers	0...99	
15	ReadError	L	INT	Fehler des letzten Lesezugriffs	0, 100... 127	
16	DPErr	L	INT	Fehlermeldungen vom DP-Modul		B
17	DPAdr_eff	L	INT	Wirksame PROFIBUS Adresse	0...126	
18	Type	L	INT	Typnr. des Funktionsblocks	0	
20	Block 21...27	L	Block			
21	HWbas	L	INT	Basic HW Optionen: Modul A, P		C
23	SWopt	L	INT	SW-Optionen 1		D
24	SWcod	L	INT	SW-Codenr. 7.-10. Stelle der 12NC	wxyz	E
25	SWvers	L	INT	SW-Codenr. 11.-12. Stelle der 12NC	00xy	F
26	OPVers ¹⁾	L	INT	Bedienversion		
27	EEPVers ¹⁾	L	INT	Versionsstand des EEPROMs		
31	OpMod	L/S	INT	Gerät in Konfigurationsmode umschalten(nur nach 1)	0	
				Gerät in Online-Mode umschalten(nur nach 0)	1	
				Abbruch des Konfigurationsmodus (nur nach 0)	2	
32	Ostartg	L/S	INT	Stoppen/Startender Selbstoptimierung aller Gruppenregler	0..1	
33	UPD	L/S	INT	Quittierender lokalen Datenänderung	0..1	G

1) Daten sind für zukünftige Verwendung vorgehalten, um interne Versionen unterscheiden zu können.

Bem. A Unit_State1



Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'
D0	'0'	immer '0'		
D1	CNF	Gerätezustand	online	configuration
D2...D4	'0'	immer '0'		
D5	UPD	ParameterUpdate	nein	ja
D6	'1'	immer '1'		
D7		Parity		

Bem. B DPErr



Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'
D0		Buszugriffnichterfolgreich	kein Fehler	Fehler
D1		fehlerhaftes Parametriertelegramm	kein Fehler	Fehler
D2		fehlerhafteKonfiguration	kein Fehler	Fehler
D3		KeinNutzdatenverkehr mehr	kein Fehler	Fehler
D4...D15		immer '0'		

Bem. C HWbas

COM2		0	0
T	H	Z	E

GrundausführungohneCOM2	0	0	0	0
COM2mitCANopen	0	1	0	0
COM2mitPROFIBUS-DP	0	2	0	0
COM2mitISO1745	0	3	0	0

Beispiel: Der Wert 'HWbas = 0200' bedeutet, daß das angesprochene Geräte eine COM2-Schnittstelle mit PROFIBUS-Anschluß besitzt.

Bem. D SWOpt

Ausführung		0	0
T	H	Z	E

Grundausführung	0	0	0	0
Wasserkühlen (noch nicht verfügbar)	0	1	0	0

Bem. E SWCod

T	H	Z	E
7. Stelle	8. Stelle	9. Stelle	10. Stelle

Beispiel: Der Wert 'SWCod= 7239' bedeutet, daß für das angesprochene Gerät die Software die Codenummer 4012 157 239xx enthält.

Bem. F SWVers

T	H	Z	E
0	0	11. Stelle	12. Stelle

Beispiel: Der Wert 'SWVers= 11' bedeutet, daß für das angesprochene Gerät die Software die Codenummer 4012 15x xxx11 enthält.

Bem. G UPD

Wird ein Parameterwert oder ein Konfigurationswert über eine Schnittstelle geändert, so wird dies im UPD-Flag angezeigt. Ebenso nach der Wiederkehr der Spannungsversorgung ist dieses Bit ge setzt. Das Flag, das auch über Code UPD gelesen werden kann, kann zurück-gesetzt werden (Wert =0).

I/O-Verbindung					(Funktions-Nr: 2)	
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
20	Block 21...24	L	Block			
21	SnOEMOpt	L	INT	Seriennummer OEM-Feld		
22	SnFabMonth	L	INT	Seriennummer Fabrikationsmonat		
23	SnCntHi	L	INT	Seriennummer Zähler High		
24	SnCntLo	L	INT	Seriennummer Zähler Low		

Parameter- u. Konfigurationsdaten

Allgemein					(Funktionsnr: 0)		
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.	
B2	41	lim_wk_enable	L/S	INT	Freigabe der Kühlfunktion für alle Kanäle	-999,9 ... 999,9	
B3	71	C900 ¹⁾ COM1	L/S	INT	Prot: Protokollart (T) Baud: Baudrate (H,Z)	0..xyy0	1)
	72	Adr1 ¹⁾	L/S	INT	COM1:Geräteadresse:	0..99	1)
	73	C904	L/S	INT	Freq: Netzfrequenz 50/60 (T)	0..x000	
	74	C902 ¹⁾ COM2	L/S	INT	Prot: Protokollart (T) Baud: Baudrate (entfällt bei PROFIBUS) (H,Z)	0..wxyz	
	75	Adr2 ¹⁾	L/S	INT	COM2:Geräteadresse: ISO1745 (def. 0) CAN-BUS PROFIBUS (def. 126)	0..99 0..255 0..126	

1) Baudrate u. Adreßeinstellung werden erst nach einer Initialisierung wirksam, z.B. Protokollumschaltung.

5.2.3 Frei konfigurierbar

(FB-Nr.: 20 ... 27 für Regler 1 - 8; 120 ... 127 für Regler 9 - 16 Typ-Nr.: 20)

Über diese Zugriffe werden die Daten für die frei definierbaren Module D ... F definiert. Mit dem ComRead- Block werden die zu lesenden und mit dem ComWrite- Block die zu schreibenden Daten spezifiziert,

Parameter- u. Konfigurationsdaten

ComRead							(Funktionsnr: 0)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.	
B2	41	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 1	0 ... 177	A
	42	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 1	0 ... 2999	
	43	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 2	0 ... 177	
	44	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 2	0 ... 2999	
	45	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 3	0 ... 177	
	46	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 3	0 ... 2999	
	47	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 4	0 ... 177	
	48	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 4	0 ... 2999	

ComWrite							(Funktionsnr: 1)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.	
B2	41	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 1	0 ... 177	B
	42	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 1	0 ... 2999	
	43	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 2	0 ... 177	
	44	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 2	0 ... 2999	
	45	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 3	0 ... 177	
	46	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 3	0 ... 2999	
	47	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 4	0 ... 177	
	48	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 4	0 ... 2999	

Bem. A Datenstruktur

Für die Definition, auf welche Date zugriffen wird, müssen folgende Einträge vorgenommen werden:

- ⌘ Funktionsblocknummer r ComReadBlock bzw. ComWriteBlock
- ⌘ Funktionsnummer + Einzel-Code r ComReadFctKey bzw. ComWriteFctKey

Beispiel:

Soll der wvol-Wert des Reglers 2 (Reglerbezeichnung 1 - 8) für ComRead ausgewählt werden, so setzen sich die Wertefolgendermaßen zusammen:

Funktionsblocknummer	Regler 2	= 51	ComReadBlock	= 51
Funktionsnummer	wvol	= 01	ComReadFctKey	= 0132
Einzel-Code	wvol	= 32		

5.2.4 INPUT

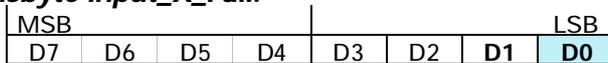
(FB-Nr.: 60 ... 67 für Regler 1 - 8; 160 ...167 für Regler 9 - 16 Typ-Nr.: 112)

In dem Funktionsblock 'INPUT' sind alle Daten, welche die Erfassung und Verarbeitung aller Eingangswerte (analog/digital) betreffen, zusammengefaßt. Die Daten sind je Reglerkanal einmal vorhanden.

Prozeßdaten

Allgemein		Eingangsverarbeitung analoger Signale				(Funktionsnr: 0)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
00	Block	L	Block	Blockzugriff (1, 3)		
1	Input_x_Fail	L	ST1	Signal Input x Fail		A
3	x1	L	FP	Hauptregelgröße		
10	Block	L	Block	Blockzugriff (13, 18)		
13	INP1	L	FP	Rohmeßwert vor Meßwertkorrektur		
18	Function Type	L	INT	Typnr. des Funktionsblocks	112	

Bem. A Statusbyte Input_X_Fail:



Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'
D0	INP1F	Input 1 Fail	nein	ja
D1...D5	'0'	immer '0'		
D6	'1'	immer '1'		
D7		Parity		

Parameter- u. Konfigurationsdaten

ME/V1		Meßwert INP1 : Erfassung u. Verarbeitung				(Funktionsnr: 1)	
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.	
B2	41 X1 _{in}	L/S	FP	Meßwertkorrektur X1 Input	-999..9999		
	42 X1 _{out}	L/S	FP	Meßwertkorrektur X1 Output	-999..9999		
	43 X2 _{in}	L/S	FP	Meßwertkorrektur X2 Input	-999..9999		
	44 X2 _{out}	L/S	FP	Meßwertkorrektur X2 Output	-999..9999		
B3	71 X0	L/S	FP	phys. Wert bei 0%	-999..9999		
	72 X100	L/S	FP	phys. Wert bei 100%	-999..9999		
	73 X _{Fail}	L/S	FP	Ersatzwert bei Sensorfail	-999..9999		
	74 T _{fm}	L/S	FP	Filterzeitkonst. Meßwertverarb.	0.0 .. 999.9		
	75 T _{kref}	L/S	FP	angenommene TK	0...60 °C / 32...140°F		
	76 C200	L/S	INT	Typ: Sensortyp Unit: Einheit	(T,H) (Z)	0..xxy0	
	77 C205	L/S	INT	Fail: Fühlerbruchverh. STk: Quelle Tk XKorr: Freig. Istwertkorr.	(T) (H) (Z)	1..wxy0	
78 C190	L/S	INT	Signalzuordnung digitaler Signale: Regleraus w/w2	(Z) (E)	0...00xy		

5.2.5 CONTR

(FB-Nr.: 50 ... 57 für Regler 1 - 8; FB-Nr. 150 ... 157 für Regler 9 - 16 Typ-Nr.: 91)
 In dem Funktionsblock 'CONTR' sind alle Daten, die den Regler betreffen, zusammengefaßt.
 Sie sind für jeden Reglerkanal einmal vorhanden.

Prozeßdaten

Allgemein					(Funktions-Nr.: 0)	
Code	Bez.	L/S	Typ	Beschreibung	Bereich.	Bem
00	Block	L	Block	Blockzugriff (1...9)		
1	Status 1	L	ST1	Status 1		A
3	W	L	FP	eff. Soll wert		
4	X	L	FP	eff. Ist wert		
5	Y	L	FP	wirksame Stellgröße		
6	xw	L	FP	Regelabweichung		
13	Status_Alarm_x	L	INT	Status_x und Alarm_x		B
18	Type	L	INT	Typnr. des Funktions blocks	90	
20	Block	L	Block	Blockzugriff (21...26)		
21	Xeff	L	FP	eff. Ist wert		
22	Yeff	L	FP	wirksame Stellgröße		
24	Unit_State	L	ICMP	Vorgabefehler Kanal 1 ... 16	r Seite 11	
25	Alarm_x	L	ICMP	Alarmwerte	r Seite 11	
26	Status_x	L	ICMP	Statusinformationen	r Seite 11	
30	Block	L	Block	Blockzugriff (31...38)		
33	A/M	L/S	INT	Automatik/Hand-Umschaltung	0..1	
34	OStart	L/S	INT	Startender Selbstoptimierung	0..1	
35	We/i	L/S	INT	Umschaltung Wext/Wint	0..1	
36	w/w2	L/S	INT	Umschaltung w/w2	0..1	
38	Coff	L/S	INT	Regler aus/ein	0..1	
39	Cntrl_x	L/S	INT	Steuerwort	0..65	C

Bem. A Status1: (Code 01)

		MSB							LSB	
		D7	D6	D5	D4	D3	D2	D1	D0	
Bit-Nr.	Name	Belegung							Zustand '0'	Zustand '1'
D0	Y1	Schaltausgang							aus	ein
D1	Y2	Schaltausgang							aus	ein
D2	A/M	Autom/Manual							Auto	Manual
D3	CFail	Zustand Regler							ok	nicht ok
D4	Coff	Regler abgeschaltet							nein	ja
D5	XFail	Sensor Fail							nein	ja
D6	'1'	immer '1'								
D7		Parity								

Bem. B Status_Alarm_x: (Code 13)

		MSB													LSB			
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	
Bit-Nr.	Name	Belegung															Zustand '0'	Zustand '1'
D0	w/w2	w/w2 Umschaltung															w	w2
D1	We/w	extern-/intern-Umschaltung															extern	intern
D2	w/Wanf	Umschaltung Anfahrswert															w	Wanf
D3	Orun	Optimierung aktiv															nein	ja
D4	A/M	Automatik/Hand-Umschaltung															Auto	Hand
D5	Coff	Regler abgeschaltet															nein	ja
D6	Y1	Schaltausgang 1															aus	ein
D7	Y2	Schaltausgang 2															aus	ein
D8	Lim HH	Alarm HH															aus	ein
D9	Lim H	Alarm H															aus	ein
D10	Lim L	Alarm L															aus	ein
D11	Lim LL	Alarm LL															aus	ein
D12	Fail	Alarm Sensor Fail															nein	ja
D13	'0'	immer '0'																
D14, D15	'0'	Immer '0'																

Bem. C Cntrl_x: (Code 39)

MSB								LSB							
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Bit-Nr.	Name	Belegung	Zustand '0'		Zustand '1'										
D0	A/M	Autom/Manual	Auto		Manual										
D1	Coff	Reglerabschalten	nein		ja										
D2	w/w2	w/w2-Umschaltung	w		w2										
D3	We/w	Wext/Wint	Wext		Wint										
D4	OStart	Optimierungstarten ¹⁾	kein starten		starten										
D5	OStop	Optimierungstoppen ¹⁾	kein stoppen		stoppen										
D6..D15	'0'	Immer '0'													

Sollwert				Sollwertverarbeitung (Funktions-Nr:1)		
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
00	Block	L	Block	Blockzugriff (1, 3)		
01	WState	L	ST1	Sollwertstatus		D
03	Wint	L	FP	wirksamer interner Sollwert		
30	Block	L	Block	Blockzugriff (31...32)		
31	Wnvol	L/S	FP	int. Sollwert, nicht flüchtig	-999..9999	
32	wvol	L/S	FP	int. Sollwert, flüchtig	-999..9999	

Bem. D WState: (Code 01)

MSB				LSB				
D7	D6	D5	D4	D3	D2	D1	D0	
Bit-Nr.	Name	Belegung	Zustand '0'		Zustand '1'			
D0	w/w2	w/w2-Umschaltung	w		w2			
D1	We/Wi	Wext/Wint	Wext		Wint			
D2	w/Wanf	w/Wanfah	w		Wanf			
D3	GRW	Gradientenfunktionaktiv	nein		ja			
D4	Weff_fail	Fehler effektiver Sollwert	nein		ja			
D5	'0'	immer '0'						
D6	'1'	immer '1'						
D7		Parity						

Stellgröße				Stellgrößenverarbeitung (Funktions-Nr:4)		
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
30	Block	L	Block	Blockzugriff (31, 35)		
31	dYman	L/S	FP	differenz. Stellgrößenvorgabe	-210..210	
32	Yman	L/S	FP	absolute Stellgrößenvorgabe	-105..105	
33	Yinc	L/S	INT	increment. Stellgrößenvorgabe	0, 1	
34	Ydec	L/S	INT	decrement. Stellgrößenvorgabe	0, 1	
35	Ygrw_Is	L/S	INT	Geschw. für incr./decr. Stellgrößenverschiebung	0, 1	

Tuning				Selbstoptimierung (Funktions-Nr:5)		
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
00	Block	L	Block	Blockzugriff (1, 3)		
1	State_Tune1	L	ST1	Status Tuning		E
3	ParNeff	L	INT	eff. Parametersatznummer	0..1	
30	Block	L	Block	Blockzugriff (31...39)		
31	ParNr	L/S	INT	Parametersatznummer wirksam	0..1	
32	Tu1	L	FP	Verzugszeit Heizen	0..9999 s	
33	Vmax1	L	FP	Anstiegsgeschwindigkeit Heizen	0,000...9,999 %/s	
34	Kp1	L	FP	Prozeßverstärkung Heizen	0,000...9,999	
35	MSG1	L	INT	Fehlercode der Selbstoptimierung Heizen	0..8	
36	Tu2	L	FP	Verzugszeit Kühlen	0..9999 s	
37	Vmax2	L	FP	Anstiegsgeschwindigkeit Kühlen	0,000...9,999 %/s	
38	Kp2	L	FP	Prozeßverstärkung Kühlen	0,000...9,999	
39	MSG2	L	INT	Fehlercode der Selbstoptim. Kühlen	0..8	

1) Signale werden nur bei Wechsel von 0 r 1 aktiv. Das Signal muß solange anstehen, bis ein Wechsel von Orun (siehe Status_Alarm_x) stattgefunden hat.

Bem. E Status 1 Tuning 'State_Tune1'

MSB				LSB			
D7	D6	D5	D4	D3	D2	D1	D0
Bit-Nr.	Name	Belegung	Zustand '0'	Zustand '1'			
D0	OStab	Prozeß in Ruhe	nein	ja			
D1	Orun	BetriebSelbstoptimierung	aus	ein			
D2	Oerr	ErgebnisSelbstoptimierung	Ok	Fehler			
D3...D5	'0'	immer '0'					
D6	'1'	immer '1'					
D7		Parity					

Parameter u. Konfigdaten

Allgemein							(Funktions-Nr: 0)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.	
B3	71	C100	L/S	INT	CFunc: Reglerfunktion CType: Reglertyp WFunc: Sollwertfunktion	(T,H) (Z) (E)	0..xyz
	72	C101	L/S	INT	CMode: Reglerwirkungs r. CDiff: x/x-w Differenzier. CFail: Verhalten bei Sensor Fail CANf: Anfahr schaltung	(T) (H) (Z) (E)	0..wxyz
	73	C700	L/S	INT	OMode: Art der Selbstoptimierung OCond: Prozeß in Ruhe. OGrp: Zuordnung Gruppen opt. OCntr: Betriebsart gest. Adapt.	(T) (H) (Z) (E)	0..wxyz
	74	C180	L/S	INT	SWext: Quelle für Wext	(T)	0..x000

Sollwert							Sollwertverarbeitung(Funktions-Nr: 1)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.	
B2	41	W0	L/S	FP	untere Sollwertgrenze f. Weff	-999..9999	
	42	W100	L/S	FP	obere Sollwertgrenze f. Weff	-999..9999	
	43	w2	L/S	FP	Zusatzsollwert	-999..9999	
	44	Grw+	L/S	FP	Sollwertgradient plus	>0..9.999	
	45	Grw-	L/S	FP	Sollwertgradient minus	>0..9.999	
	46	Grw2	L/S	FP	Sollwertgradient w2	>0..9.999	

Algo							Regelalgorithmus (Funktions-Nr: 3)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.	
B2	41	Xsh	L/S	FP	Neutrale Zone	0.2 .. 20,0 %	
	42	Tpuls	L/S	FP	Mindestimpulslänge	0.1..2,0 s	
	43	Tm	L/S	FP	Motorlaufzeit des Stellmotors	10..300 s	
	44	Xsd1	L/S	FP	Schaltdifferenzsignalgerät	0,1..9999 %	
	45	LW	L/S	FP	Schaltpunkt abstand Zusatzk.	-999..9999	
	46	Xsd2	L/S	FP	Schaltdifferenz Zusatzk.	0,1..9999 %	
	47	Xsh1	L/S	FP	Neutrale Zone	0.0 .. 999.9%	
	48	Xsh2	L/S	FP	Neutrale Zone	0.0 .. 999.9 %	

Stellgröße							Stellgrößenbearbeitung(Funktionsnr: 4)
Code	Bez.	L(S)	Typ	Beschreibung	Bereich	Bem.	
B2	41	Y _{min}	L/S	FP	untere Stellgrößenbegrenzung	-105..105 %	
	42	Y _{max}	L/S	FP	ober Stellgrößenbegrenzung	-105..105 %	
	43	Y0	L/S	FP	Arbeitspunkt f. Stellgröße	-105..105 %	
	44	Yh	L/S	FP	maximaler mittelwert der Stellgröße	5..100%	
	45	LYh	L/S	FP	Grenze für Mittelwertbildung	0,1 .. 10,0	

1) Datum besitzt Abschaltfunktion; zusätzlicher Datenwert '-32000'

Tuning					Selbstoptimierung(Funktionsnr: 5)			
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.		
B2	41	YOptm	L/S	FP	Stellgrö Be während Prozeß in Ruhe	-105..105		
	42	dYopt	L/S	FP	Sprunghöhebeidentifikation	5..100		
	43	OXsd	L/S	FP	HystereseebeiParamumschalt	0.0..9999		
	44	Trig1	L/S	FP	Umschaltpunkt 1	0.0..9999		
	45	POpt	L/S	INT	Parametersatzder optimiert werden soll	0..1		

Paramset x					Regelparametersatz 1 / 2(Funktionsnr: 6,7)			
Code	Bez.	L/S	Typ	Beschreibung	Wertebereich	Bem.		
B2	41	Xp1	L/S	FP	Proportionalbereich1	0.1..999.9		
	42	Tn1	L/S	FP	Nachstellzeit 1	0..9999		
	43	Tv1	L/S	FP	Vorhaltezeit1	0..9999		
	44	T1	L/S	FP	min.Periodendauer 1	0.4..999.9		
	45	Xp2	L/S	FP	Proportionalbereich2	0.1..999.9		
	46	Tn2	L/S	FP	Nachstellzeit2	0..9999		
	47	Tv2	L/S	FP	Vorhaltezeit2	0..9999		
	48	T2	L/S	FP	min.Periodendauer 2	0.4..999.9		

Anfahrerschaltung					(Funktionsnr: 10)			
Code	Bez.	L/S	Typ	Beschreibung	Wertebereich	Bem.		
B2	41	Ya	L/S	FP	maximaler Stellwert	5 .. 100 %		
	42	Wa	L/S	FP	Anfahrswert	-999 .. 9999		
	43	TPa	L/S	FP	Anfahrhaltezeit	0 .. 9999 min		

5.2.6 ALARM

(FB-Nr.: 70 ... 777 für Regler 1 - 8; FB-Nr. 170 ... 177 für Regler 9 - 16 Typ-Nr.: 91)
 Der Funktionsblock 'ALARM' definiert die gesamte Alarmverarbeitung des zugehörigen Controllers. Die Daten sind je Reglerkanal einmal vorhanden.

Prozeßdaten

Allgemein		(Funktionsnr: 0)				
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
00	Block	L	Block	Blockzugriff (1..3)		
1	Status_AI1	L	ST1	Alarmstatus1		A
18	Type	L	INT	Typnr. des Funktionsblocks	46	

Bem. A Status_AI1

		MSB			LSB				
		D7	D6	D5	D4	D3	D2	D1	D0
Bit-Nr.	Name	Belegung			Zustand '0'		Zustand '1'		
D0	Lim HH	Alarm HH			aus		ein		
D1	Lim H	Alarm H			aus		ein		
D2	Lim L	Alarm L			aus		ein		
D3	Lim LL	Alarm LL			aus		ein		
D4	Fail	Fail			nein		ja		
D5	'0'	immer '0'							
D6	'1'	immer '1'							
D7		Parity							

Parameter u. Konfigurationsdaten

Allgemein		(Funktionsnr: 0)					
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.	
B2	41	LimL	L/S	FP	unterer Voralarm	-999..9999	.
	42	LimH	L/S	FP	oberer Voralarm	-999..9999	
	43	xsd1	L/S	FP	Schalt differenz Voralarme	0..9999	
	44	LimLL	L/S	FP	unterer Hauptalarm	-999..9999	
	45	LimHH	L/S	FP	oberer Hauptalarm	-999..9999	

1) Datum besitzt Abschaltfunktion; zusätzlicher Datenwert '-32000'

6 Funktionsbaustein für SIMATIC® S7

Die prinzipielle Handhabung des S7-FB entspricht der S5 Variante. Der FB ist bedingt aufzurufen bei Auslösung eines Auftrages und solange der Auftrag aktiv ist.

Je nach S7-CPU und eingesetztem DP-Master ergeben sich Unterschiede im E/A-Handling. Bei einer CPU315-2 DP mit on-board DP-Schnittstelle sind die SFC-Bausteine 14 und 15 zu benutzen, um Daten konsistent zu übertragen. Die SFC-Bausteine 14 und 15 kopieren die E/A-Bereiche in den Merker- oder Datenbausteinbereich. Bei Benutzung eines externen CP's (CP 342-5 DP), sind die entsprechenden DP-SEND und DP-RECEIVE FB's am Anfang und Ende des Zyklusses aufzurufen.

Der FB besitzt eine Instanz-DB, der bei FB-Aufruf mit anzugeben ist.

6.1 Aufbau

Der Funktionsbaustein besitzt folgende Aufrufparameter:

Name	Typ	Beschreibung / Funktion																																
A-Anfang	Pointer	Anfang Adressbereich der Ausgangsworte (z. B. Adresse Datenbereich 'RECORD' des SFC 15, Ax, y bei Nutzung eines externen CP's). Bei Angabe eines Datenwortes muß die DB-Nr. mit übergeben werden (z. B. DB4.DBX0.0)																																
E-Anfang	Pointer	Anfang Adressbereich der Eingangsworte (z. B. Adresse Datenbereich 'RECORD' des SFC 15, Ex, y bei Nutzung eines externen CP's). Bei Angabe eines Datenwortes muß die DB-Nr. mit übergeben werden (z. B. DB4.DBX0.0)																																
DB-Para	Pointer	Angabe des Datenbausteins mit den Parametern. Die Eingabe umfaßt die Datenbaustein-Nr. und die Datenwort-Nr. wo die Parameter datenbeginnen. Es ist dabei kein Offset zu berücksichtigen. Die Daten werden von der angegebenen Adresse als Parameterdaten (Nutzdaten) interpretiert. Die Angabe des DB muß in folgender Form erfolgen z. B. DB6.DBX10.0																																
Service	WORD	Service (Read/Write)																																
Code_nr	WORD	Code																																
FB_nr	WORD	Funktionsblock-Nr. (Kanaladressierung)																																
FKT_nr	WORD	Funktions-Nr.																																
Typ	WORD	keine Funktion (immer '0')																																
Timeout	DWORD	Timeout-Wert, wird bei je dem Aufruf dekrementiert. Ist der Wert = 1, wird der Auftrag mit der Fehlermeldung 'timeout' abgebrochen.																																
DWLR	WORD	Länge der Real-Werte																																
DWLI	WORD	Länge der Integer-Werte																																
DWLC	WORD	d.c. immer '0'																																
ANZW	W	Im Anzeigewort ist der aktuelle Zustand der Übertragung für den gewählten Datenbereich ersichtlich. Das Anzeigewort besitzt folgenden Aufbau:																																
		<table border="1"> <thead> <tr> <th>15</th> <th>14</th> <th>13</th> <th>12</th> <th>11</th> <th>10</th> <th>9</th> <th>8</th> <th>7</th> <th>6</th> <th>5</th> <th>4</th> <th>3</th> <th>2</th> <th>1</th> <th>0</th> </tr> </thead> <tbody> <tr> <td>Timeout (FB)</td> <td>Service falsch</td> <td></td> <td></td> <td></td> <td>NAK (Zugriff vom Regler nicht akzeptiert)</td> <td>Parity error</td> <td>Timeout intern (Regler)</td> <td></td> <td>warten auf Endelegramm</td> <td>Service (0=Read; 1=Write)</td> <td>Reset Auftrag</td> <td>wartet auf Quittung</td> <td>Auftrag fertig mit Fehler</td> <td>Auftrag fertig ohne Fehler</td> <td>Auftrag läuft</td> </tr> </tbody> </table>	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Timeout (FB)	Service falsch				NAK (Zugriff vom Regler nicht akzeptiert)	Parity error	Timeout intern (Regler)		warten auf Endelegramm	Service (0=Read; 1=Write)	Reset Auftrag	wartet auf Quittung	Auftrag fertig mit Fehler	Auftrag fertig ohne Fehler	Auftrag läuft
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																			
Timeout (FB)	Service falsch				NAK (Zugriff vom Regler nicht akzeptiert)	Parity error	Timeout intern (Regler)		warten auf Endelegramm	Service (0=Read; 1=Write)	Reset Auftrag	wartet auf Quittung	Auftrag fertig mit Fehler	Auftrag fertig ohne Fehler	Auftrag läuft																			

Der Funktionsbaustein liest bzw. schreibt Parameter-/Konfigurationsdaten des KS816.

- W **A-Anfang, E-Anfang**
In diese Parameter werden die Eingangsadressen bzw. Ausgangsadressen des Parameterkanals eingegeben. Die Adressen werden bei der Konfiguration des PROFIBUS Teilnehmers festgelegt (STEP 7 - Hardware Konfiguration)
- W **DB-Para**
DB-Para ist ein Zeiger auf den Datenbaustein, in den gelesene Daten geschrieben werden bzw. aus dem beim Schreiben Daten entnommen werden.
- W **Service**
Dieser Parameter bestimmt die Zugriffsart (Schreiben / Lesen) r ID1.

Schreibzugriff:	F0 = Integer	Lesezugriff:	0 = Integer
	F1 = Real		1 = Real

Einzelzugriff

Mit diesem Zugriff (Code xx) kann ein einzelner Wert einer Funktion gelesen bzw. geschrieben werden.

Gültige Werte für ID1:

Konfiguration als Fix-Point:	0	Realwerte werden als Integer (ohne Nachkommastelle) übertragen
	1	Realwerte werden als FixPoint (1 Nachkommastelle) übertragen

Blockzugriff (Zehner-Block)

Mit diesem Zugriff (Code x0) können maximal neun Prozeßwerte (immer als REAL-Werte) einer Funktion gelesen bzw. geschrieben werden.

Blockzugriff (Gesamt-Block)

Mit diesem Zugriff können alle Parameter- (Code 178) und Konfigurationsdaten (Code 179) einer Funktion gelesen bzw. geschrieben werden. Für diesen Zugriff gelten folgende Bedingungen:

- Um Daten mit 'Code B3 = 179' schreiben zu können, muß vorher das Gerät in den Konfigurationsmodus (r siehe Seite 24 'OpMod') geschaltet werden. Wirksam werden alle neu eingegebenen Konfigurationsdaten und Parameter erst, wenn das Gerät wieder auf online zurückgeschaltet wurde.
- Alle Daten einer Nachricht müssen definiert sein, Auslassungen sind nicht zulässig.
- Sind Teile einer Nachricht im Gerät nicht in Benutzung (HW- und SW-Optionen), so ist dennoch die komplette Nachricht zu übertragen. Die Prüfung der nicht vorhandenen Daten entfällt.
- Bei fehlerhaften Blockschreibzugriffen gilt: Eine Nachricht wird mit NAK beantwortet, wenn mindestens ein Datum fehlerhaft ist. Bereits gültige Werte werden übernommen.

Die Reihenfolge der zu übertragenen Daten ist der jeweiligen Code-Tabelle zu entnehmen.

Gültige Werte für ID1:

Konfiguration als Fix-Point:	0, 1	Übertragung Realwerte jeweils als FixPoint-Wert
------------------------------	------	---

- W **Code_nr**
Die-Code-Kennung ist Dezimal und der Wertebereich umfaßt '00'...'99' sowie '178' = B2 und '179' = B3.
- W **FB_nr. (Funktionsblocknummer)**
Ein Funktionsblock wird mit einer Funktionsblocknummer adressiert. Sie umfaßt den Wertebereich '0' bis '250'.
Funktionsblocknummernbereiche:
0 allgemeine Daten für das gesamte Gerät
1 - 99 fest eingerichtete Funktionsblöcke
- W **FKT_nr (Funktionsnummer)**
Eine Funktion als Teiladresse eines Funktionsblock wird ebenfalls mit einer Funktionsnummer angesprochen. Sie umfaßt den Wertebereich '0' bis '99'.
Funktionsnummernbereiche:

0 Funktion Allgemein
1 - 99 andere Funktionen

W Typ (Funktionstyp)

Je dem Funktionsblock ist auch eine Funktions typ nummer zu ge ord net. Sie um faßt den Wertebereich '0' bis '111'.

Funktionstypenbereiche:

0 Funktionstyp Allgemein
1 - 111 andere Funktionstypen

W Timeout

Timeoutzähler: Bereich 0x0000 β TIME β 0x7FFF

- wird bei jedem SPS-Zyklus dekrementiert (max. 32767)
- bei 0 Timeout.

Sollte die CPU zu schnell sein, FB206/FB207 über Timerbaustein verzögert aufrufen.

W DWLR (Real), DWLI (Integer)

Diese Parameter enthalten nach einem Lesezugriff die jeweilige Anzahl der empfangenen Daten. Bei einem Schreibzugriff ist die jeweilige Anzahl der zu übertragenden Daten eingetragen. DWLC wird im KS 816 nicht benötigt, Wert ist auf 0 zu setzen.

W ANZW

Dieses Anzeigewort bildet den aktuellen Zustand der Übertragung ab. Das Bit 4 kann als Eingang zum Rücksetzen (Reset) des FB 206 / FB 207 verwendet werden.

Die Auswahl des Reglerkanals erfolgt über die FB_nr.

7 Anhang

7.1 Begriffe

COM PROFIBUS	Konfigurations tool (früher COM ET200) der Fa. Siemens für PROFIBUS
FB	Abk. f. Funktionsblock
Fkt	Abk. für Funktion
ET	Abk. f. Engineering Tool
Funktion	eine aus Sicht der Schnittstelle geschlossene Teilfunktion eines Funktionsblocks
Funktionsblock	geschlossene Abarbeitungseinheit
GSD-Datei	Geräte Stamm Datei
HW	Abk. f. Hardware
ISO1745	Genormtes Kommunikationsprotokoll ISO 1745, ASCII basiert
PC-Schnittstelle	frontseitige Schnittstelle am Regler zum Anschluß eines Engineering Tools
PCI	Process Control Instrument
PCI-Protokoll	Protokoll auf Basis ISO 1745, implementiert für PMA Regler
PNO	PROFIBUS Nutzer Organisation
PROFIBUS-DP	Genormtes Kommunikationsprotokoll nach EN50170 Vol.2 (DP: Dezentrale Peripherie)
RS422	Genormte 4 Drahtverbindung, Full duplex, (EIA RS 422); hier: getrennte Sendempfangskanäle mit bis zu 32 Teilnehmern
RS485	Genormte 2 Drahtverbindung, Half duplex, (EIA RS 485)
S5 / S7	Steuerungsfamilien der Siemens AG
Serielle Schnittstelle	Rückseitige Busfähige Schnittstelle des Reglers
SW	Abk. f. Software
Typdatei	Konfigurationsdatei für COM ET200

7.2 GSD Datei

```

=====
; Device Database File for product K S 8 1 6 - D P
; Copyright (C) PMA Prozeß-und Maschinen-Automation GmbH 2001
; D-34123 Kassel, Miramstr. 87, Tel. +49 (0) 561/ 505 -1307
; Release : V1.0
; File: PMA_0801.gsd
=====
#Profibus_DP
GSD_Revision = 1
Vendor_Name = "PMA GmbH"
Model_Name = "KS 816-DP"
Revision = "V 1.1 "
Ident_Number = 0x0801
Protocol_Ident = 0 ; DP
Station_Type = 0 ; Slave
FMS_supp = 0
Hardware_Release = "HV 01.00"
Software_Release = "SV 01.00"
;supported baud rates:
9.6_supp = 1
19.2_supp = 1
45.45_supp = 1
93.75_supp = 1
187.5_supp = 1
500_supp = 1
1.5M_supp = 1
3M_supp = 1
6M_supp = 1
12M_supp = 1
;max. time to answer after a request
MaxTsdr_9.6 = 60
MaxTsdr_19.2 = 60
MaxTsdr_45.45 = 60

```

```

MaxTsdr_93.75 = 60
MaxTsdr_187.5 = 60
MaxTsdr_500 = 100
MaxTsdr_1.5M = 150
MaxTsdr_3M = 250
MaxTsdr_6M = 450
MaxTsdr_12M = 800
Redundancy = 0 ; not supported
Repeater_Ctrl_Sig = 2 ; TTL
24V_Pins = 0 ; not available
Bitmap_Device = "PMA0816N"
Bitmap_Diag = "PMA0816D"
Bitmap_SF = "PMA0816F"
;
;—DP-Slave related key words——-
;
Freeze_Mode_supp = 1 ; supported
Sync_Mode_supp = 1 ; supported
Auto_Baud_supp = 1
Set_Slave_Add_supp = 0
User_Prm_Data_Len = 0 ; no user prm data
;minimum slave poll cycle (Basis 100us):
Min_Slave_Intervall = 1
Modular_Station = 1 ; modular device
Max_Module = 0x01 ; max. number of modules
Max_Input_Len = 116
Max_Output_Len = 116
Max_Data_Len = 232
; Module description
; 1. measuring values for 16 channels + parameter channel
Module = "A:Measured data(16)+ parameter" \
    0x50,0x10,0x50,0x10,0x50,0x10,0x50,0x10,\
    0x50,0x10,0x50,0x10,0x50,0x10,0x50,0x10,\
    0x50,0x10,0x50,0x10,0x50,0x10,0x50,0x10,\
    0x50,0x10,0x50,0x10,0x50,0x10,0x50,0x10,\
    0xF3
EndModule
;
; 2. Controller values + parameter channel
Module = "B:Process data(16)+parameter" \
    0x11,\
    0x52, 0x52, 0x52, 0x52, 0x52, 0x52, 0x52, 0x52,\
    0x52, 0x52, 0x52, 0x52, 0x52, 0x52, 0x52, 0x52,\
    0x21,\
    0x62, 0x62, 0x62, 0x62, 0x62, 0x62, 0x62, 0x62,\
    0x62, 0x62, 0x62, 0x62, 0x62, 0x62, 0x62, 0x62,\
    0xF3
EndModule
;
; 3. Only parameter channel
Module = "C: Parameter" 0xF3
EndModule
;
; 4. Process data for 52 Variable data + parameter channel
Module = "D: 52 Variable data + parameter" \
    0x15,\
    0x57, 0x57, 0x57, 0x57, 0x57, 0x57, 0x51,\
    0x21,\
    0x67, 0x67, 0x67, 0x67, 0x67, 0x67, 0x61,\

```

```

    0xF3
EndModule
;
; 5. Process data for 40 Variable data + parameter channel
Module = "E: 40 Variable data + parameter" \
    0x15,\
    0x57, 0x57, 0x57, 0x57, 0x57,\
    0x21,\
    0x67, 0x67, 0x67, 0x67, 0x67,\
    0xF3
EndModule
;
; 6. Multiplexing of Process data for 1 Variable data + parameter channel
Module = "F: Multiplexed data + parameter" \
    0x15,\
    0x50, 0x50,\
    0x21,\
    0x60, 0x60,\
    0xF3
EndModule
;
; Device related diagnostic data
Unit_Diag_Bit(0) = "Configuration state"
Unit_Diag_Bit(8) = "Input fail channel 1"
Unit_Diag_Bit(9) = "Input fail channel 2"
Unit_Diag_Bit(10) = "Input fail channel 3"
Unit_Diag_Bit(11) = "Input fail channel 4"
Unit_Diag_Bit(12) = "Input fail channel 5"
Unit_Diag_Bit(13) = "Input fail channel 6"
Unit_Diag_Bit(14) = "Input fail channel 7"
Unit_Diag_Bit(15) = "Input fail channel 8"
Unit_Diag_Bit(16) = "Input fail channel 9"
Unit_Diag_Bit(17) = "Input fail channel 10"
Unit_Diag_Bit(18) = "Input fail channel 11"
Unit_Diag_Bit(19) = "Input fail channel 12"
Unit_Diag_Bit(20) = "Input fail channel 13"
Unit_Diag_Bit(21) = "Input fail channel 14"
Unit_Diag_Bit(22) = "Input fail channel 15"
Unit_Diag_Bit(23) = "Input fail channel 16"
;
Slave_Family=5
Max_Diag_Data_Len = 10
Fail_safe = 0

```

