



Multi-temperature controller KS 800

The background features a large, light gray 'KS 800' watermark. Overlaid on this is a schematic diagram of a PID control loop. It shows a feedback loop with a controller block labeled 'PID' and a process block. The text 'ISO 1745' is prominently displayed in the center of the diagram.

ISO 1745

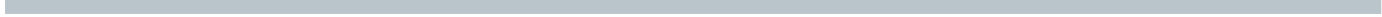
Interface description
ISO 1745 protocol
9499 040 49411
valid from: 8357

© PMA Prozeß- und Maschinen-Automation GmbH 1999 Printed in Germany (9909)
All rights reserved. No part of this document may be
reproduced or published in any form or by any means without prior written permission
from the copyright owner.

A publication of PMA Prozeß- und Maschinen Automation
P.O. Box 310229
D.34058 Kassel
Germany

Inhalt

1	Hints on operation	5
1.1	Connecting the interface	5
2	Interface protocol	6
2.1	Protocol layer 1	6
2.1.1	Data format	6
2.1.2	Baud rate	6
2.1.3	Parity	6
2.1.4	Addressing	6
2.2	Protocol layer 2	6
2.2.1	Transmission control characters	7
2.2.2	Character format	7
3	Message structure	8
3.1	Message elements	8
3.2	Basic message structure	9
3.3	Data types	10
4	Standard protocol	11
4.1	CODE table	11
5	Function block protocol	13
5.1	Data structure	13
5.2	CODE tables	14
5.2.1	Structure of configuration words (C.xxxx)	14
5.2.2	INSTRUMENT (FB no.: 0 type no.: 0)	14
5.2.3	Sonderzugriffe (FB-Nr.: 10 ... 17 Typ-Nr.: 10)	20
5.2.4	Frei konfigurierbar (FB-Nr.: 20 ... 27 Typ-Nr.: 20)	21
5.2.5	INPUT (FB no.: 60 ... 67 Type no.: 112)	23
5.2.6	CONTR (FB no.: 50 ... 57 Type no.: 91)	24
5.2.7	ALARM (FB-Nr.: 70 ... 77 Typ-Nr.: 46)	27
6	Examples	28
6.1	Message examples in standard protocol	28
6.2	Principles of the function block protocol	28
6.2.1	Single access	28
6.2.2	Block access (tens block)	28
6.2.3	Block access (overall block)	29
6.3	Message structure in function block protocol	30
6.3.1	INSTRUMENT	30
6.3.2	INPUT	30
6.3.3	CONTR	30
6.3.4	ALARM	30



7	Annex	31
7.1	Terms	31
8	Index	32

1 Hints on operation

Multi-temperature controller version KS800-RS is provided with a serial, bussable RS485 interface which can be used for transmission of process, parameter and configuration data. Connection is via (a) 9-pole sub-D socket(s)(connector(s)). The serial communication interface permits connections to supervisory PLCs, visualization tools, etc.

An RS485/422 hardware interface is realized. The protocol available on this hardware is:

- the PCI protocol, which is based on an ISO 1745 frame,

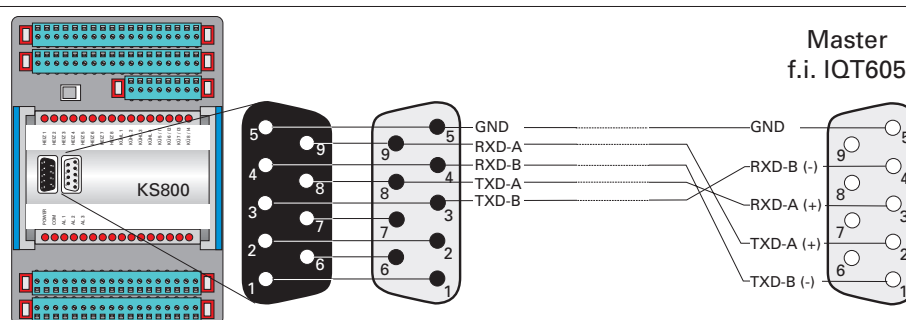
Communication is according to the master/slave principle. KS800 is always slave. The software of the serial interface is implemented as standard in the firmware.

Another standard interface is the PC interface. This interface is used for connecting an engineering tool, which runs on an external PC.

1.1 Connecting the interface

Version KS800-RS offers an RS485 or RS422 interface. 'RS422' as available in this product means an RS485 4-wire interface. A driver for reception and a driver for sending are available.

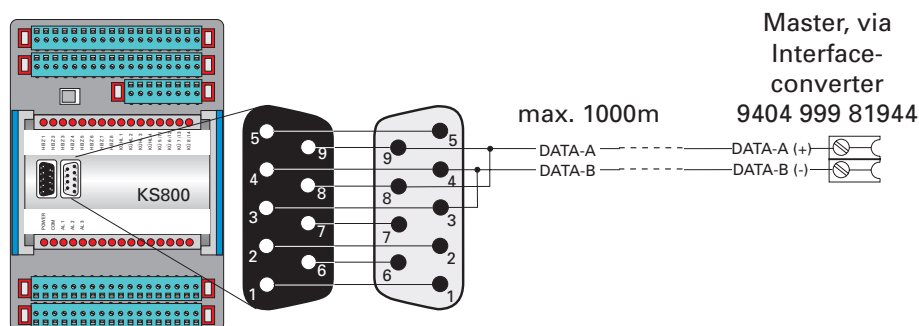
Fig.: 1 Connection examples RS422 interface



Attention! No galvanic isolation.

On the 2-wire RS 485 interface, reception and transmission lines must be galvanically connected by the user.

Fig.: 2 Connection examples RS485 interface



If an RGND connection is required with an RS485 adjustment, a 100 Ohm resistor must be mounted across terminal 5 (RGND) and terminal 5 of the interface converter by the user.

The outputs are galvanically isolated.

The interface mode is half-duplex.

Installing appropriate cables must be done by the user, whereby the general cable specifications according to EIA RS485 must be taken into account.

2 Interface protocol

2.1 Protocol layer 1

Bus connection is physical:

- via the PC interface as a TTL signal (COM 1)
- via an RS485/422 connection (COM 2) with version KS800-RS.

2.1.1 Data format

The following transmission format, fixed, must be used:

- 1 start bit,
- 7 bits ASCII value or 7 bits binary
- 1 parity bit (EVEN)
- 1 stop bit

LSB is transmitted first, MSB is parity bit.

2.1.2 Baud rate

The Baud rate for the serial interface is adjustable. The following Baud rates are available:

- 2400 Baud
- 4800 Baud
- 9600 Baud
- 19200 Baud

2.1.3 Parity

Parity detection is fixed to EVEN.

2.1.4 Addressing

KS 800 can be operated together with KS40, KS50, KS90, KS92, KS94, KS98, KS4580, DIGITAL 280/380 and PRO 96 and the ICS 90 and ITS 90 systems at the same bus. Decisive for instrument selection is the address (2 bytes).

The KS 800 (0...99) address is adjusted via the “KS800 Engineering Tool” (general instrument settings → communication → address).

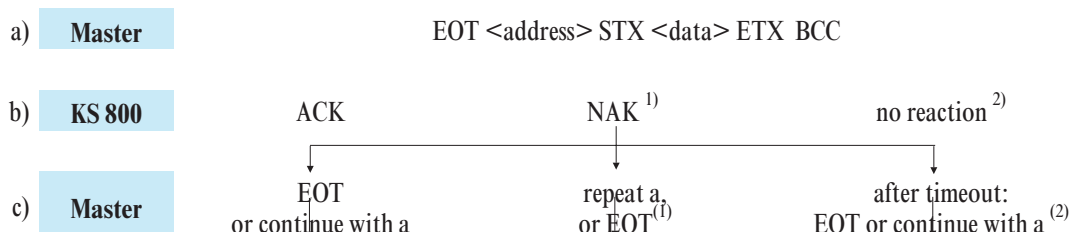
2.2 Protocol layer 2

A fixed master/slave principle is used, whereby KS800 is always slave. Transmission control (communication start and cancelation by EOT) is always by the master.

Two communication services are available:

- for data sending: SDA (Send Data with Acknowledge), acknowledged by KS 800

Data flow direction : master → KS 800

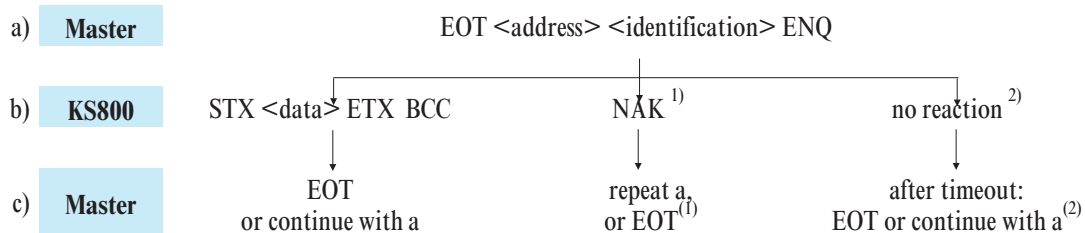


1) Possible after disturbance of transmission or after sending inadmissible data.

2) Possible after KS 800 failure, bus failure or faulty address specification.

- for data request: RDR (Request Data with Reply)
data request with reply in one message cycle.

Data flow direction : KS800 → Master



2.2.1 Transmission control characters

The following transmission control characters are used:

Abbreviation	HEX	Description
STX	02	Start of Text - data introduction
ETX	03	End of Text - end of data
EOT	04	End of Transmission - reset the interface units or transmission cancelation
ENQ	05	Enquiry - request for reply
ACK	06	Acknowledge - confirmation
NAK	15	Not Acknowledge - no confirmation

2.2.2 Character format

Numbers and characters in the address, identification or data field are always transmitted as ASCII characters.

The following 7-bit ASCII characters with parity (EVEN) are valid

CHR	HEX	Description
,	2C	Comma as separator
=	3D	Separator between identification and value
0...9	30...39	Values for numbers and codes
B	41	Additional for codes
:...?	3A...3F	Values for floating point format (FP)
@...;	40...7F	Values for status and control bytes
...;	20...7F	Characters for text string (CHAR16)
.	2E	Decimal point

1) Possible after disturbance of transmission or after sending inadmissible data.
2) Possible after KS 800 failure, bus failure or faulty address specification.

3 Message structure

3.1 Message elements

In the following section, some expressions which shall be explained as follows are used:

Element	Description	Rem.
<addr>	Address of a participating unit, always 2 bytes long, adjustable on the instruments	A
<daten>	Data field composed of a) fields <identification> a. <value>, separated by character '=' b) a series of successive <value> with several block accesses	B
<identification>	identification field composed of a) field <code> and b) different selection criteria <selection>	C
<value>	Value of a datum, which is addressed with a key.	
<code>	Addressing key of a datum, 2-digit, decimal number range, first digit also 'B'.	D
<selection>	further addressing field for selection of <function block no.> a. <function no.>	E
<BCC>	Block Check Count. All characters between STX (exclusive) and ETX (inclusive) are connected bitwisely by an EXOR function and output as 1 byte, always after ETX.	F

Bem. A *Address field*

The address field can be transmitted only after 'EOT' and must be generated only by the master. It is two bytes long. The address number range is 00 ... 99. If the transmitted address corresponds with the one adjusted in the unit, the message is intended for this unit. Different address settings are possible for COM1 and COM2.

Bem. B *Data field*

The data field contains the parameters and data to be transmitted. The equality sign is followed by the value of a datum (<valuex>). Several data are separated by a comma. The data type depends on the access. The last value before 'ETX' ends without ','.
With block read access with additional selection criteria, these criteria are specified only once. The data follow without further identifications. Thus, the message structure becomes more compact.

Bem. C *Identification field*

The identification field addresses a defined datum or a data area in the instrument. It consists of a code and of an additional selection identification with some accesses.
With a data enquiry, the identification field contains information for KS92/94 which data the unit is expected to send. This is always followed by the address field. In the reply, it is also specified for clear determination of the datum, followed by the data field with separator „=“.
With data entry, STX is followed by the identification field for addressing the values to be specified. Connection of the data field is by means of character „=“.

Bem. D *Code*

The code identification is two bytes long and the value range is ASCII '00'...'99' and 'B2'...'B3'.

Bem. E Addition selection criteria

In order to form a purposeful sub-set from the variety of data, additional selection criteria are defined:

- Function block number**
A function block is addressed by a function block number. It is within '0' and '250' and is appended to the code field by means of a comma. '*<code>,<function block no.>*'

Function block number ranges:

- 0 general data for the overall instrument
- 1 - 99 fixed function blocks
- 100 - 250 variable function blocks

- Function number**
A function as a partial address of a function block is addressed with a function number. It is within '0' and '99' and is appended to the function block number by means of a comma. '*<code>,<function block no.>,<function no.>*'

Function number ranges:

- 0 - 99 Functions
- 0 Function General, Default, unless a no. is specified

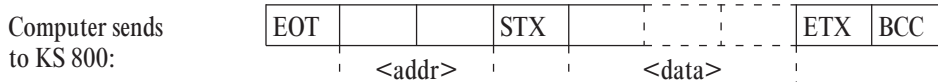
Bem. F Safety procedure

Correct transmission of a message is supported by two safety procedures:

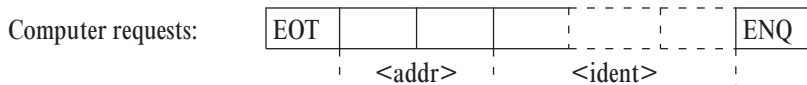
- check of each message byte by formation of parity (1 bit per 7 data bits)
- Check by Block Check Count : safety section, which connects all characters of a message between STX (excl.) and ETX (incl.) byte-wisely by a logic XOR function; length 1 byte, follows always after ETX.

3.2 Basic message structure

Message structure with data transmission:



Message structure with data enquiry:



3.3 Data types

Data values are classified according to data types for transmission. Only characters which can be represented in ASCII are permitted.

- BCD
Floating Point number in BCD-ASCII format,
Range: -9999 ... -0.001, 0, 0.001 ... 9999
optional: negative polarity sign and decimal point permitted; exponent representation not permitted.
KS800 controllers with an accuracy of max. 4 digits. With received data, number of digits and decimal point position are not fixed and depend on the FP resolution. The values are not rounded off.
Switch-off value for BCD data is : -32000
- INT
positive integer number in ASCII format
Range: 0 ... 32767
Range with configuration words: 0000 ... 9999 (→ page 14)
Exception: switch-off value '-32000'
- ST1
Status, bit-oriented, 1 byte length
Range: 00H ... 3FH, transmitted: 40H...7FH
Only 6 bits for transmission of information can be used, i.e. bit 0...5 (LSB = bit 0). Bit 6 must always be set to '1', in order to avoid confusion with the control characters. Bit 7 contains the parity bit.
- SYS16
System identification number, 16 bytes
Format: xx,yyyyyyyy,zzzz (→ page 11)

4 Standard protocol

The KS800 standard protocol version represents instrument-specific standard data.

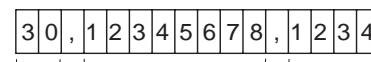
4.1 CODE table

Code	Description	R/W	Type	Range	Description	Rem.
18	System ident	R	SYS16		System identification	A
80	Block 81... 83	R	Block			
81	Write Error	R	INT	0, 100 ... 127	Error of last write access	B
82	Write Error Position	R	INT	0 ... 99	Position of last write access error	
83	Read Error	R	INT	0, 100 ... 127	Error of last read access	

Bem. A Instrument data

System identification number (code 18)

For instrument identification, instrument type and software code number can be read via code 18. The datum is composed of the following sections:



Instrument type: (30 =KS800)

SW code number: (the last 8 digits)

Instrument version: 7th to 10th digit of 12NC (4 digits)

Bem. B Diagnosis access: block 8x

For test purposes, an additional debug access which signals errors of the last write or read access is available. Presently, reading is possible for:

- error number of last write access; 0 = no error
- position of the faulty datum during the last write access;
 - 0 = no error or error in address
 - 1 = first datum is faulty (also with single accesses)
 - n = nth datum is faulty (with block accesses)
- error number of last read access; 0 = no error

An independent memory for error messages is available for each interface COM1 and COM2. Presently, the following error messages are defined:

Err. no.	Description	Error name
101	unspecified error	ERR_UNSPECIFIED
102	read not permitted	ERR_RD_NOTALLOWED
103	write not defined	ERR_WR_NOTALLOWED
104	local operation / no write access	ERR_LOCOPERAT
105	non-defined key code	ERR_KEYIDENT
106	range overflow function block no.	ERR_FB_OVERFL
107	range overflow function no.	ERR_FCT_OVERFL
108	write or range overflow	ERR_WR_RANGE_OV
109	char is not a digit	ERR_NODIGIT
110	no '\0' found in the correct position	ERR_ENDDELIMITER
111	no '=' in the correct position	ERR_NO_EQUALSIGN
112	faulty ST1 format (status)	ERR_NO_ST1FORMAT
113	no ',' in the correct position	ERR_NO_COMMA
114	byte range overflow	ERR_BYTE_OVERFL
115	number of digits exceeded	ERR_DIGIT_OVERFL
116	range 9999 exceeded	ERR_RG9999_OVERFL
117	undefined protocol type	ERR_UNDEF_PRTCTYPE
118	undefined parameter reference	ERR_UNDEF_PARAMREF
119	undefined decimal point	ERR_UNDEF_DECPNT

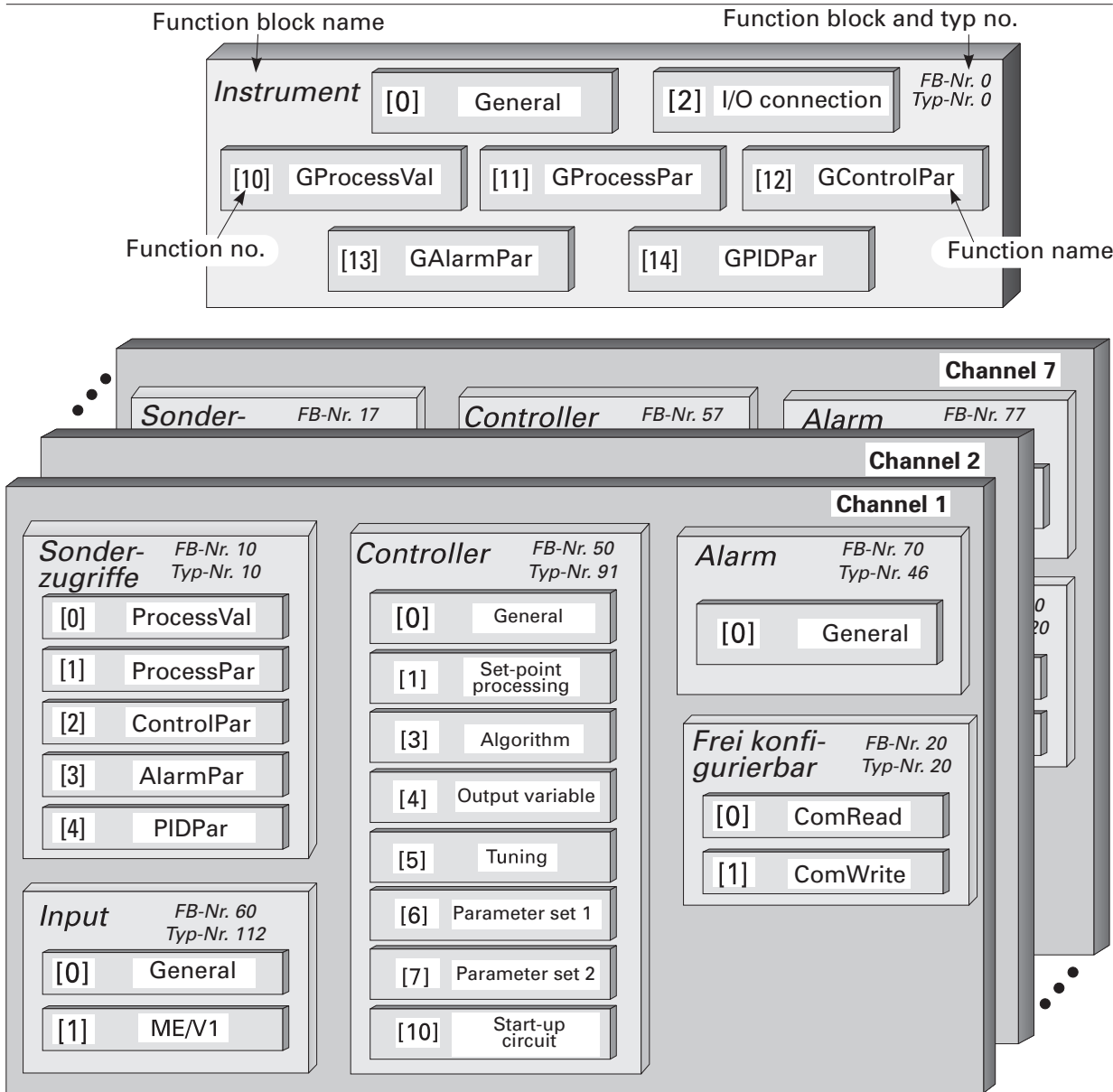
Err. no.	Description	Error name
120	no STX in the write message	ERR_NO_STX
121	number INT faulty	ERR_INT_ANZ
122	number REAL faulty	ERR_REAL_ANZ
123	faulty access type	ERR_ZUGRIFF
124	no config level	ERR_WR_NO_CONF
125	local operation	ERR_WR_LOCAL
126	error FI switch-over	ERR_WR_FU_UM

5 Function block protocol

5.1 Data structure

Due to the variety of information to be processed in KS800, logically related data and actions are grouped in function blocks. A function block has input, output data, parameters and configuration data. For KS800, 25 function blocks are defined. They are all addressed via fixed block addresses (FB no.). Each block is also divided into several functions. Functions are addressed via function numbers (Fct-no.). Function number 0 addresses function block-specific data.

Fig.: 3 Survey of KS 800 function blocks and functions



5.2 CODE tables

5.2.1 Structure of configuration words (C.xxxx)

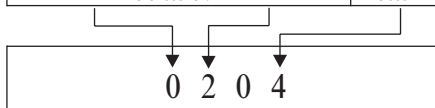
The configuration words mentioned in the following code tables are composed of several sections, which can be transmitted only in common.
The data in the table must be interpreted as follows:

Example (C100):

Code	Descr.	R/W	Type	Description	Range
71	C100	R/W	INT	CFunc: Controller function (T,H) Wfunc: set-point function (E)	0..xyz

Description	CFunc		WFunc
	Thousands	Hundreds	Ones
Range	x	x	z
	00 ... 07		0...1

Example: continuous controller; standard controller;
set-point-cascade with offset



- i** - For transmission of configuration words, see chapter page 29.
- Die Einstellmöglichkeiten der Konfigurationswörter entnehmen Sie der KS 800 Funktionsbeschreibung (Best. Nr. 9499 040 49218)

5.2.2 INSTRUMENT (FB no.: 0 type no.: 0)

Function block 'INSTRUMENT' is used for classification of all data which are valid for the overall instrument.

Process data

General						(function no.: 0)	
Code	Descr.	R/W	Type	Description	Range	Rem.	
01	Unit_State 1	R	ST1	Status 1		A	
10	Block 13..18	R	Block				
13	Write Error	R	INT	Error during last write access	0, 100...127	→ p. 11	
14	Write Error Position	R	INT	Position of last write access error	0...99		
15	Read Error	R	INT	Error of last read access	0, 100...127		
16	DPErr	R	INT	Error message from DP-module		B	
17	DPAAdr_eff	R	INT	Effectiv PROFIBUS adress	0 ... 126		
18	Type	R	INT	Type no. of function block	0		
20	Block 21...27	R	Block				
21	HWbas	R	INT	Basic HW options: module A, P		C	
23	SWopt	R	INT	SW options 1		D	
24	SWcod	R	INT	SW code no. 7th to 10th digit of 12NC	wxyz	E	
25	SWvers	R	INT	SW code no. 11th to 12th digit of 12NC	00xy	F	
26	OPVers ¹⁾	R	INT	Operating version			
27	EEPVers ¹⁾	R	INT	EEPROM version			
31	OpMod	R/W	INT	Switch over unit to configuration mode (only after 1)	0		
				Switch over unit to on-line mode (only after 0)	1		
				Cancelation of configuration mode (only after 0)	2		
32	Ostartg	R/W	INT	Self-tuning stop/start of all group controllers	0..1		
33	UPD	R/W	INT	Acknowledgement of local data change	0..1	G	

1) Data are specified for future use (distinction of internal version).

Bem. A Unit_State1

MSB				LSB			
D7	D6	D5	D4	D3	D2	D1	D0
Bit no.	Name	Allocation	Status '0'	Status '1'			
D0	'0'	Always '0'					
D1	CNF	Instrument status	on-line	configuration			
D2...D4	'0'	Always '0'					
D5	UPD	Parameter update	no	yes			
D6	'1'	Always '1'					
D7		Parity					

Bem. B DPErr

MSB														LSB	
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Bit no.	Name	Allocation	Status '0'	Status '1'											
D0		Buszugriff nicht erfolgreich	Kein Fehler	Fehler											
D1		Fehlerhaftes Prametrierelegramm	Kein Fehler	Fehler											
D2		Fehlerhafte Konfiguration	Kein Fehler	Fehler											
D3		Kein Nutzdatenverkehr	Kein Fehler	Fehler											
D4...D15	'0'	Always '0'													

Bem. C HWbas

COM2		0	0
T	H	Z	E

Basic version without COM2	0	0	0	0
COM2 with CANopen	0	1	0	0
COM2 with PROFIBUS-DP	1	0	0	0
COM2 with ISO1745	1	1	0	0

Example: Value 'HWbas = 0100' means that the addressed instrument is a COM2 interface with CANopen connection.

Bem. D SWopt

Version		0	0
T	H	Z	E

Basic version	0	0	0	0
Water cooling (so far not available)	0	1	0	0

Bem. E SWCod

T	H	Z	E
7th digit	8th digit	9th digit	10th digit

Example: Value 'SWCod= 7239' means that the software for the addressed instrument contains code number 4012 157 239xx.

Bem. F SWvers

T	H	Z	E
0	0	11th digit	12th digit

Example: Value 'SWVers= 11' means that the software of the addressed unit contains code number 4012 15x xxx11.

Bem. G UPD

Changing a parameter value or a configuration value via an interface is indicated in the UPD flag. This bit is also set after mains recovery. The flag, which can be read also via code UPD, can be reset (value =0).

I/O connection						(Function no.: 2)	
Code	Des.	R/W	Type	Description	Range	Rem.	
0	Block 1...2	R	Block				
1	State_alarm_out	R	ST1	Status alarm outputs		G	
2	State_dio	R	ST1	Status digital inputs/outputs		H	
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			
30	Block 31...33	L	Block				
31	Fdo1	L/S	ICMP	Forced digitale Ausgänge: OUT1 ... OUT8		J	
32	Fdo2	L/S	ICMP	Forced digitale Ausgänge: OUT9 ... OUT16		K	
33	Fdo3	L/S	ICMP	Forced digitale Ausgänge: OUT17 ... OUT19		L	

Bem. H State_alarm_out

MSB				LSB			
D7	D6	D5	D4	D3	D2	D1	D0
Bit no.	Name	Allocation	Status '0'	Status '1'			
D0	R1	Relay 1	off	on			
D1	R2	Relay 2	off	on			
D2	R3	Relay 3	off	on			
D3	do1_12 AL	Alarm output short circuit do1 ... do12	off	on			
D4	HCscAL	Alarm message heating current short circuit	off	on			
D5	'0'	Always '0'					
D6	'1'	Always '1'					
D7		Parity					

Bem. I State_dio

MSB				LSB			
D7	D6	D5	D4	D3	D2	D1	D0
Bit no.	Name	Allocation	Status '0'	Status '1'			
D0	Par_Nr	Parameter set number	set 0	set 1			
D1	w/w2	w/w2 switch-over	w	w2			
D2	Coff	Controller off	off	on			
D3	Leck	Leakage current	off	on			
D4	'0'	Always '0'					
D5	do13_16f	do13 ... do16 Fail	no	yes			
D6	'1'	Always '1'					
D7		Parity					

Bem. J Aufbau der Datenstruktur

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bedeutung	0	0	0	0	0	0	0	0	OUT8	OUT7	OUT6	OUT5	OUT4	OUT3	OUT2	OUT1

Bem. K Aufbau der Datenstruktur

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bedeutung	0	0	0	0	0	0	0	0	OUT16	OUT15	OUT14	OUT13	OUT12	OUT11	OUT10	OUT9

Bem. L Aufbau der Datenstruktur

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Bedeutung	0	0	0	0	0	0	0	0	0	0	0	0	0	0	OUT19	OUT18	OUT17

GProcessVal						(Funktions-Nr: 10)	
Code	Bez.	Kanal	L/S	Typ	Beschreibung	Bereich	Bem.
B2	Xeff	1	L	INT	Effektiver Istwert von Kanal 1		
	Yeff	1	L	INT	Effektive Stellgröße vom Kanal 1		
	HC	1	L	INT	Heizstrom vom Kanal 2		
	Xeff	2	L	INT	Effektiver Istwert von Kanal 2		
	Yeff	2	L	INT	Effektive Stellgröße vom Kanal 2		
	HC	2	L	INT	Heizstrom vom Kanal 2		
	...						
	Xeff	8	L	INT	Effektiver Istwert von Kanal 8		
	Yeff	8	L	INT	Effektive Stellgröße vom Kanal 8		
	HC	8	L	INT	Heizstrom vom Kanal 8		
	State_alarm_out	1	L	ST1	Status der Alarmausgänge von Kanal 1		H
	State_alarm_out	2	L	ST1	Status der Alarmausgänge von Kanal 2		H
	...						
	State_alarm_out	8	L	ST1	Status der Alarmausgänge von Kanal 8		H

GProcessPar						(Funktions-Nr: 11)	
Code	Bez.	Kanal	L/S	Typ	Beschreibung	Bereich	Bem.
B2	Wvol	1	L/S	INT	Flüchtiger Sollwert von Kanal 1		
	W2	1	L/S	INT	Zusatzsollwert von Kanal 1		
	Yman	1	L/S	INT	absolute Stellgrößenvorgabe von Kanal 1		
	Wboost	1	L/S	INT			
	Tboost	1	L/S	INT			
	Wvol	2	L/S	INT	Flüchtiger Sollwert von Kanal 2		
	W2	2	L/S	INT	Zusatzsollwert von Kanal 2		
	Yman	2	L/S	INT	absolute Stellgrößenvorgabe von Kanal 2		
	Wboost	2	L/S	INT			
	Tboost	2	L/S	INT			
	...						
	Wvol	8	L/S	INT	Flüchtiger Sollwert von Kanal 8		
	W2	8	L/S	INT	Zusatzsollwert von Kanal 8		
	Yman	8	L/S	INT	absolute Stellgrößenvorgabe von Kanal 8		
	Wboost	8	L/S	INT			
	Tboost	8	L/S	INT			

GControlPar						(Funktions-Nr: 12)	
Code	Bez.	Kanal	L/S	Typ	Beschreibung	Bereich	Bem.
B2	A/M	1	L/S	INT	Automatik/Hand- Umschaltung von Kanal 1		
	Coff	1	L/S	INT	Regler ein/aus von Kanal 1		
	w/W2	1	L/S	INT	Umschaltung w/W2 von Kanal 1		
	Ostart	1	L/S	INT	Start der Selbstoptimierung von Kanal 1		
	SoftStartEnable	1	L/S	INT			
	BoostStartEnable	1	L/S	INT			
	A/M	2	L/S	INT	Automatik/Hand- Umschaltung von Kanal 1		
	Coff	2	L/S	INT	Regler ein/aus von Kanal 1		
	w/W2	2	L/S	INT	Umschaltung w/W2 von Kanal 1		
	Ostart	2	L/S	INT	Start der Selbstoptimierung von Kanal 2		
	SoftStartEnable	2	L/S	INT			
	BoostStartEnable	2	L/S	INT			
	...						
	A/M	8	L/S	INT	Automatik/Hand- Umschaltung von Kanal 1		
	Coff	8	L/S	INT	Regler ein/aus von Kanal 1		
	w/W2	8	L/S	INT	Umschaltung w/W2 von Kanal 1		
	Ostart	8	L/S	INT	Start der Selbstoptimierung von Kanal 1		
	SoftStartEnable	8	L/S	INT			
BoostStartEnable	8	L/S	INT				

GAlarmPar						(Funktions-Nr: 13)	
Code	Bez.	Kanal	L/S	Typ	Beschreibung	Bereich	Bem.
B2	LimL	1	L/S	INT	Unterer Voralarm von Kanal 1		
	LimH	1	L/S	INT	Oberer Voralarm von Kanal 1		
	LimLL	1	L/S	INT	Unterer Hauptalarm von Kanal 1		
	LimHH	1	L/S	INT	Oberer Hauptalarm von Kanal 1		
	LimL	2	L/S	INT	Unterer Voralarm von Kanal 2		
	LimH	2	L/S	INT	Oberer Voralarm von Kanal 2		
	LimLL	2	L/S	INT	Unterer Hauptalarm von Kanal 2		
	LimHH	2	L/S	INT	Oberer Hauptalarm von Kanal 2		
	...						
	LimL	8	L/S	INT	Unterer Voralarm von Kanal 8		
	LimH	8	L/S	INT	Oberer Voralarm von Kanal 8		
	LimLL	8	L/S	INT	Unterer Hauptalarm von Kanal 8		
	LimHH	8	L/S	INT	Oberer Hauptalarm von Kanal 8		

GPIDPar					(Funktions-Nr: 14)		
Code	Bez.	Kanal	L/S	Typ	Beschreibung	Bereich	Bem.
B2	Xp1	1	L/S	INT	Proportionalbereich für Kanal 1		
	Tn1	1	L/S	INT	Nachstellzeit für kanal 1		
	Tv1	1	L/S	INT	Vorhaltezeit für Kanal 1		
	T1	1	L/S	INT	minimale Periodendauer für Kanal 1		
	Xp1	2	L/S	INT	Proportionalbereich für Kanal 2		
	Tn1	2	L/S	INT	Nachstellzeit für kanal 2		
	Tv1	2	L/S	INT	Vorhaltezeit für Kanal 2		
	T1	2	L/S	INT	minimale Periodendauer für Kanal 2		
	...						
	Xp1	8	L/S	INT	Proportionalbereich für Kanal 8		
	Tn1	8	L/S	INT	Nachstellzeit für kanal 8		
	Tv1	8	L/S	INT	Vorhaltezeit für Kanal 8		
T1	8	L/S	INT	minimale Periodendauer für Kanal 8			

Parameter a. configuration data

General					(function no.: 0)		
Code	Des.	R/W	Type	Description	Range	Rem.	
B2	41	lim_wk_enable	R/W	INT	Freigabe der Kühlenfunktion für alle Kanäle	-999,9 ... 999,9	
B3	71	C900 ¹⁾ COM1	R/W	INT	Prot: protocol type (T) Baud: Baud rate (H,Z)	0..xyy0	
	72	Adr1 ¹⁾	L/S	INT	COM1: instrument address:	0..99	
	73	C904	R/W	INT	Freq: mains frequency 50/60 (T)	0..x000	
	74	C902 ¹⁾ COM2	R/W	INT	Prot: protocol type (T) Baud: Baud rate (H,Z)	0..wxyz	
	75	Adr2 ¹⁾	R/W	INT	COM2: Instrument address: ISO1745 CAN-BUS	0..99 0..255	

I/O connection					(function no.: 2)		
Code	Des.	R/W	Type	Description	Range	Rem.	
B3	71	HC100	R/W	BCD	Span end for HC	1...9999	
	72	C500	R/W	ICNF	Main configuration di1/do13 ... di4/do16 Fkt_dio1: di1 / do13 (T) Fkt_dio2: di2 / do14 (H) Fkt_dio3: di3 / do15 (Z) Fkt_dio4: di4 / do16 (E)	0..wxyz	
	73	C530	R/W	ICNF	Main configuration do17 ... do19 mode_do17 (T) mode_do18 (H) mode_do19 (Z)	0...xyz0	
	74	C151	R/W	ICNF	Allocation HC/leakage current alarm DestHC (T) DestLeak (H) DestOutError (Z)	0...xyz0	
	75	HCycl	R/W	INT	Heating current cycle time	0...999	

1) Baud rate and adress setting are only effective after initialization, e.g. protocol switch-over.

5.2.3 Sonderzugriffe (FB-Nr.: 10 ... 17 Typ-Nr.: 10)

Der Funktionsblock 'Sonderzugriffe' ermöglicht, auf die Daten des KS800 in anderer Form zuzugreifen. Der Zugriff kann nur über den Code B2 ausgeführt werden.

ProcessVal						(Funktions-Nr: 0)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
B2	Xeff	L	INT	Effektiver Istwert		
	Yeff	L	INT	Effektive Stellgröße		
	HC	L	INT	Heizstrom		
	State_alarm_out	L	ST1	Status der Alarmausgänge		

ProcessPar						(Funktions-Nr: 1)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
B2	Wvol	L/S	INT	Flüchtiger Sollwert		
	W2	L/S	INT	Zusatzsollwert		
	Yman	L/S	INT	absolute Stellgrößenvorgabe		
	Wboost	L/S	INT	momentan keine Funktion		
	Tboost	L/S	INT	momentan keine Funktion		

ControlPar						(Funktions-Nr: 2)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
B2	A/M	L/S	INT	Automatik/Hand-Umschaltung		
	Coff	L/S	INT	Regler ein/aus		
	w/W2	L/S	INT	Umschaltung w/W2		
	Osart	L/S	INT	Start der Selbstoptimierung		
	SoftStartEnable	L/S	INT	momentan keine Funktion		
	BoostStartEnable	L/S	INT	momentan keine Funktion		

AlarmPar						(Funktions-Nr: 3)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
B2	LimL	L/S	INT	unterer Voralarm		
	LimH	L/S	INT	oberer Voralarm		
	LimLL	L/S	INT	unterer Hauptalarm		
	LimHH	L/S	INT	oberer Hauptalarm		

PIDPar						(Funktions-Nr: 4)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
B2	Xp1	L/S	INT	Proportionalbereich 1		
	Tn1	L/S	INT	Nachstellzeit 1		
	Tv1	L/S	INT	Vorhaltezeit 1		
	T1	L/S	INT	minimale Periodendauer		

5.2.4 Frei konfigurierbar (FB-Nr.: 20 ... 27 Typ-Nr.: 20)

Der Funktionsblock "Frei konfigurierbar" definiert Daten die dann per Blockzugriff 20 bzw. 30 gelesen werden können. Die Daten des ComWrite können mit den Schlüsseln 31 - 38 auch geändert werden. Zusätzlich stellt diese Einstellung der Profibus-Schnittstelle für die entsprechenden Datenmodulen die Werte zur Verfügung.

ComRead						(Funktions-Nr: 0)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
20	Block	L	Block		21... 28	
21	Val 1	L	Datenspezifisch	Wert 1		A
22	Val 2	L	Datenspezifisch	Wert 2		
23	Val 3	L	Datenspezifisch	Wert 3		
24	Val 4	L	Datenspezifisch	Wert 4		
25	Val 5	L	Datenspezifisch	Wert 5		
26	Val 6	L	Datenspezifisch	Wert 6		
27	Val 7	L	Datenspezifisch	Wert 7		
28	Val 8	L	Datenspezifisch	Wert 8		

ComWrite						(Funktions-Nr: 1)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.
30	Block	L	Block		31... 38	
31	Val 1	L/S	Datenspezifisch	Wert 1		A
32	Val 2	L/S	Datenspezifisch	Wert 2		
33	Val 3	L/S	Datenspezifisch	Wert 3		
34	Val 4	L/S	Datenspezifisch	Wert 4		
35	Val 5	L/S	Datenspezifisch	Wert 5		
36	Val 6	L/S	Datenspezifisch	Wert 6		
37	Val 7	L/S	Datenspezifisch	Wert 7		
38	Val 8	L/S	Datenspezifisch	Wert 8		

Bem. A *Datenspezifisch*

In Abhängigkeit von dem eingestellten Parameter wird der Wert als INT oder Status ausgegeben. Nicht definierte Werte werden im INT-Format mit -31000 ausgegeben.

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 ... 77	B
	42	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 1	0 ... 2999	
	43	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 2	0 ... 77	
	44	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 2	0 ... 2999	
	45	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 3	0 ... 77	
	46	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 3	0 ... 2999	
	47	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 4	0 ... 77	
	48	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 4	0 ... 2999	
	49	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 5	0 ... 77	
	51	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 5	0 ... 2999	
	52	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 6	0 ... 77	
	53	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 6	0 ... 2999	
	54	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 7	0 ... 77	
	55	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 7	0 ... 2999	
	56	ComReadBlock1	L/S	INT	Funktionsblocknummer für Wert 8	0 ... 77	
	57	ComReadFctKey1	L/S	INT	Funktionsnummer und Code für Wert 8	0 ... 2999	

ComRead							(Funktionsnr: 0)
Code	Bez.	L/S	Typ	Beschreibung	Bereich	Bem.	
B2	41	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 1	0 ... 77	B
	42	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 1	0 ... 2999	
	43	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 2	0 ... 77	
	44	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 2	0 ... 2999	
	45	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 3	0 ... 77	
	46	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 3	0 ... 2999	
	47	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 4	0 ... 77	
	48	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 4	0 ... 2999	
	49	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 5	0 ... 77	
	51	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 5	0 ... 2999	
	52	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 6	0 ... 77	
	53	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 6	0 ... 2999	
	54	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 7	0 ... 77	
	55	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 7	0 ... 2999	
	56	ComWriteBlock1	L/S	INT	Funktionsblocknummer für Wert 8	0 ... 77	
	57	ComWriteFctKey1	L/S	INT	Funktionsnummer und Code für Wert 8	0 ... 2999	

Bem. B Datenstruktur

Für die Definition, auf welche Date zugegriffen wird, müssen folgende Einträge Vorgenommen werden:

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

Beispiel:

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

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

5.2.5 INPUT (FB no.: 60 ... 67 Type no.: 112)

All data which concern acquisition and processing of all input values (analog/digital) are grouped in function block 'INPUT'. The data are provided once per controller channel.

Process data

General		Input processing of analog signals (Function no.: 0)				
Code	Des.	R/W	Type	Description	Range	Rem.
00	Block	R	Block	Block access (1, 3)		
1	Input_x_Fail	R	ST1	Signal Input x Fail		A
3	x1	R	BCD	Main variable		
10	Block	R	Block	Block access (13, 18)		
13	INP1	R	BCD	Raw meas. value before meas. value correction		
18	Function Type	R	INT	Type no. of function block	112	

Bem. A Statusbyte Input_X_Fail:

MSB				LSB			
D7	D6	D5	D4	D3	D2	D1	D0
Bit no.	Name	Allocation	Status '0'	Status '1'			
D0	INP1F	Input 1 Fail	no	yes			
D1...D5	'0'	Always '0'					
D6	'1'	Always '1'					
D7		Parity					

Parameter a. configuration data

ME/V1		Measurement value INP1 : acquisition and processing (Function no.: 1)					
Code	Des.	R/W	Type	Description	Range	Rem.	
B2	41	X1 _{in}	R/W	BCD	Measurement value correction X1 Input	-999..9999	
	42	X1 _{out}	R/W	BCD	Measurement value correction X1 Output	-999..9999	
	43	X2 _{in}	R/W	BCD	Measurement value correction X2 Input	-999..9999	
	44	X2 _{out}	R/W	BCD	Measurement value correction X2 Output	-999..9999	
B3	71	X0	R/W	BCD	Phys. value at 0%	-999..9999	
	72	X100	R/W	BCD	Phys. value at 100%	-999..9999	
	73	X _{Fail}	R/W	BCD	Substitute value at sensor fail	-999..9999	
	74	T _{fm}	R/W	BCD	Filter time const. meas. Value processing.	0.0 .. 999.9	
	75	T _{kref}	R/W	BCD	Customer-specified TC	0...60 °C /32...140°F	
	76	C200	R/W	INT	Type: sensor type Unit: unit	(T,H) (Z) 0..xxy0	
	77	C205	R/W	INT	Fail: sensor fail behaviour STk: TC source XKorr: process value correction enable	(T) (H) (Z) 1..wxy0	
	78	C190	R/W	INT	Allocation of digital signals: controller off w/w2	(Z) (E) 0...00xy	

5.2.6 CONTR (FB no.: 50 ... 57 Type no.: 91)

All data concerning the controller are grouped in function block 'CONTR'. They are provided once for each controller channel.

Process data

General							(Function no.: 0)
Code	Des.	R/W	Type	Description	Range	Rem	
00	Block	R	Block	Block access (1...9)			
1	Status 1	R	ST1	Status 1		A	
3	W	R	BCD	Eff. set-point			
4	X	R	BCD	Eff. process value			
5	Y	R	BCD	Effective output variable			
6	xw	R	BCD	Control deviation			
18	Type	R	INT	Type no. of function block	90		
20	Block	L	Block	Blockzugriff (21...26)			
21	Xeff	L	FP	eff. Istwert			
22	Yeff	L	FP	wirksame Stellgröße			
23	HC	L	FP	Heizstrommeßwert			
24	Unit_State	L	ICMP	Eingangswerte (di)	→ Seite		
25	Alarm_x	L	ICMP	Alarmwerte	→ Seite		
26	Status_x	L	ICMP	Statusinformationen	→ Seite		
30	Block	R	Block	Block access (31...38)			
33	A/M	R/W	INT	Automatic/manual switch-over	0..1		
34	OStart	R/W	INT	Self-tuning start	0..1		
35	We/i	R/W	INT	Wext/Wint switch-over	0..1		
36	w/W2	R/W	INT	w/W2 switch-over	0..1		
38	Coff	R/W	INT	Controller off/on	0..1		

Bem. A Status1: (code 01)

MSB				LSB			
D7	D6	D5	D4	D3	D2	D1	D0
Bit no.	Name	Allocation	Status '0'	Status '1'			
D0	Y1	Switching output	off	on			
D1	Y2	Switching output	off	on			
D2	A/M	Autom/manual	Auto	Manual			
D3	CFail	Controller status	ok	not ok			
D4	Coff	Controller switched off	no	yes			
D5	XFail	Sensor Fail	no	yes			
D6	'1'	Always '1'					
D7		Parity					

Set-point				Set-point processing			(Function no.:1)
Code	Des.	R/W	Type	Description	Range	Rem.	
00	Block	R	Block	Block access (1, 3)			
01	WState	R	ST1	Set-point status		B	
03	Wint	R	BCD	Effective internal set-point			
30	Block	R	Block	Block access (31...32)			
31	Wnvol	R/W	BCD	Int. set-point, non-volatile	-999..9999		
32	Wvol	R/W	BCD	Int. set-point, volatile	-999..9999		

Bem. B WState: (code 01)

MSB				LSB			
D7	D6	D5	D4	D3	D2	D1	D0
Bit no.	Name	Allocation	Status '0'	Status '1'			
D0	w/W2	w/W2 switch-over	w	W2			
D1	We/Wi	Wext/Wint	Wext	Wint			
D2	w/Wanf	w/Wanfähr	w	Wanf			
D3	GRW	Gradient function active	no	yes			
D4	Weff_fail	Error effective set-point	no	yes			
D5	'0'	Always '0'					
D6	'1'	Always '1'					
D7		Parity					

Output variable				Output variable processing (function no.:4)		
Code	Des.	R/W	Type	Description	Range	Rem.
30	Block	R	Block	Block access (31, 35)		
31	dYman	R/W	BCD	Difference output variable	-210..210	
32	Yman	R/W	BCD	Absolute output variable	-105..105	
33	Yinc	R/W	INT	Increment. output variable	0, 1	
34	Ydec	R/W	INT	Decrement. output variable	0, 1	
35	Ygrw_ls	R/W	INT	Speed for incr./decr. output variable offset	0, 1	

Tuning				Self-tuning(Function no.:5)		
Code	Des.	R/W	Type	Description	Range	Rem.
00	Block	R	Block	Block access (1, 3)		
1	State_Tune1	R	ST1	Status Tuning		C
3	ParNeff	R	INT	Eff. parameter set number	0...1	
30	Block	R	Block	Block access (31...39)		
31	ParNr	R/W	INT	Parameter set number effective	0 .. 1	
32	Tu1	R	BCD	Delay time heating	0...9999 s	
33	Vmax1	R	BCD	Rate of increase heating	0,000...9,999 %/s	
34	Kp1	R	BCD	Process gain heating	0,000...9,999	
35	MSG1	R	INT	Error code of self-tuning heating	0...8	
36	Tu2	R	BCD	Delay time cooling	0...9999 s	
37	Vmax2	R	BCD	Rate of increase cooling	0,000...9,999 %/s	
38	Kp2	R	BCD	Process gain cooling	0,000...9,999	
39	MSG2	R	INT	Error code of self-tuning cooling	0...8	

Bem. C Status 1 Tuning 'State_Tune1'

Bit no.	Name	Allocation	MSB				LSB			
			D7	D6	D5	D4	D3	D2	D1	D0
D0	OStab	Process at rest					no			
D1	Orun	Self-tuning mode					off			
D2	Oerr	Self-tuning result					Ok			
D3...D5	'0'	Always '0'								
D6	'1'	Always '1'								
D7		Parity								

Parameter a. configuration data

General				(Function no.: 0)			
Code	Des.	R/W	Type	Description	Range	Rem.	
B3	71	C100	R/W	INT	CFunc: controller function (T,H) CType: controller type (Z) WFunc:set-point function (E)	0..xxyz	
	72	C101	R/W	INT	CMode:controller output action (T) CDiff: x/x-w differentiation (H) CFail: behaviour with sensor fail (Z) CANf: start-up circuit (E)	0..wxyz	
	73	C700	R/W	INT	OMode: self-tuning mode (T) OCond: process at rest (H) OGrp: allocation group self-tuning (Z) OCntr: controlled adapt. mode (E)	0..wxyz	
	74	C180	R/W	INT	SWext: source for Wext (T)	0..x000	

Set-point				Set-point processing(Function no.: 1)			
Code	Des.	R/W	Type	Description	Range	Rem.	
B2	41	W0	R/W	BCD	Lower set-point limit f. Weff	-999..9999	
	42	W100	R/W	BCD	Upper set-point limit f. Weff	-999..9999	
	43	W2	R/W	BCD	Additional set-point	-999..9999	
	44	Grw+	R/W	BCD	Set-point gradient	>0..9.999	1)
	45	Grw-	R/W	BCD	Set-point gradient minus	>0..9.999	
	46	Grw2	R/W	BCD	Set-point gradient W2	>0..9.999	

Algo				Control algorithm (Function no.: 3)			
Code	Des.	R/W	Type	Description	Range	Rem.	
B2	41	Xsh	R/W	BCD	Neutral zone	0.2 .. 20,0 %	
	42	Tpuls	R/W	BCD	Min. pulse length	0.1..2,0 s	1)
	43	Tm	R/W	BCD	Actuator response time	10..300 s	
	44	Xsd1	R/W	BCD	Switching difference signaller	0,1..9999 %	
	45	LW	R/W	BCD	Trigger point separation addit. contact	-999..9999	
	46	Xsd2	R/W	BCD	Switching difference addit. contact	0,1..9999 %	
	47	Xsh1	R/W	BCD	Neutral zone	0.0 .. 999.9%	
	48	Xsh2	R/W	BCD	Neutral zone	0.0 .. 999.9 %	

Output variable				Output variable processing(Function no.: 4)			
Code	Des.	R/W	Type	Description	Range	Rem.	
B2	41	Y _{min}	R/W	BCD	Min. output limiting	-105..105 %	
	42	Y _{max}	R/W	BCD	Max. output limiting	-105..105 %	
	43	Y0	R/W	BCD	Working point f. output variable	-105..105 %	
	44	Yh	R/W	BCD	Maximum mean output value	5..100%	
	45	LYh	R/W	BCD	Limit for mean value formation	0,1 .. 10,0	

Tuning				Self-tuning(Function no.: 5)			
Code	Des.	R/W	Type	Description	Range	Rem.	
B2	41	YO _{ptm}	R/W	BCD	Output variable during process at rest	-105..105	
	42	dYO _{pt}	R/W	BCD	Step height with identification	5..100	
	43	PO _{pt}	R/W	INT	Parameter set to be optimized	0..1	
	44	OXS _d	R/W	BCD	Hysteresis with parameter selection	0.0..9999	
	45	Trig1	R/W	BCD	Trigger point 1	0.0..9999	

Paramset x				Control parameter set 1 / 2(Function no.: 6,7)			
Code	Des.	R/W	Type	Description	Range	Rem.	
B2	41	Xp1	R/W	BCD	Proportional band 1	0.1..999.9	
	42	Tn1	R/W	BCD	Integral time 1	0..9999	
	43	Tv1	R/W	BCD	Derivative time 1	0..9999	
	44	T1	R/W	BCD	Min. cycle time 1	0.4..999.9	
	45	Xp2	R/W	BCD	Proportional band 2	0.1..999.9	
	46	Tn2	R/W	BCD	Integral time 2	0..9999	
	47	Tv2	R/W	BCD	Derivative time 2	0..9999	
	48	T2	R/W	BCD	Min. cycle time 2	0.4..999.9	

Start-up circuit				(Function no.: 10)			
Code	Des.	R/W	Type	Description	Range	Rem.	
B2	41	Ya	R/W	BCD	Max. output value	5 .. 100 %	
	42	Wa	R/W	BCD	Start-up set-point	-999 .. 9999	
	43	TPa	R/W	BCD	Start-up holding time	0 .. 9999 min	

1) Datum has switch-off function; additional data value '-32000'

5.2.7 ALARM (FB-Nr.: 70 ... 77 Typ-Nr.: 46)

Function block 'ALARM' defines the overall alarm processing of the relevant controller. The data are provided once per controller channel.

Process data

General							(Function no.: 0)
Code	Des.	R/W	Type	Description	Range	Rem.	
00	Block	R	Block	Block access (1 .. 3)			
1	Status_AI1	R	ST1	Alarm status 1		A	
2	Status_AI2	R	ST1	Heating current alarm		B	
3	HC	R	BCD	Heating current meas. value			
18	Type	R	INT	Type no. of function block	46		

Bem. A Status_AI1

MSB				LSB			
D7	D6	D5	D4	D3	D2	D1	D0
Bit no.	Name	Allocation	Status '0'	Status '1'			
D0	Lim HH	Alarm HH	off	on			
D1	Lim H	Alarm H	off	on			
D2	Lim L	Alarm L	off	on			
D3	Lim LL	Alarm LL	off	on			
D4	Fail	Fail	no	yes			
D5	'0'	Always '0'					
D6	'1'	Always '1'					
D7		Parity					

Bem. B Status_AI2

MSB				LSB			
D7	D6	D5	D4	D3	D2	D1	D0
Bit-Nr.	Name	Allocation	Status '0'	Status '1'			
D0	HCA1	Heating current alarm channel	off	on			
D1	LeckA1	Leakage current alarm channel	off	on			
D2	do1_8A1	Output alarm do1..do8	off	on			
D3.. D5	'0'	Always '0'					
D6	'1'	Always '1'					
D7		Parity					

Parameter a. configuration data

General							(Function no.: 0)
Code	Des.	R/W	Type	Description	Range	Rem.	
B2	41	LimL	R/W	BCD	Low limit alarm	-999..9999	1)
	42	LimH	R/W	BCD	High limit alarm	-999..9999	
	43	xsd1	R/W	BCD	Switching difference low and high alarms	0..9999	
	44	LimLL	R/W	BCD	Low low limit alarm	-999..9999	1)
	45	LimHH	R/W	BCD	High high limit alarm	-999..9999	
	46	LimHC	R/W	BCD	Heating current limit value	0..HC100	
B3	71	C600	R/W	INT	Src: signal source Fnc: function DestFail: fail Destination	(T,H) (Z) (E)	0..xyz
	72	C601	R/W	INT	DestLL : DestL : DestH : DestHH :	(T) (H) (Z) (E)	0..wxyz

1) Datum has switch-off function; additional data value '-32000'

6 Examples

6.1 Message examples in standard protocol

Example:

The computer requests system identification (code 18) from KS800 with address 01.

Computer requests:

EOT	0	1	1	8	ENQ
-----	---	---	---	---	-----

| <addr> | <code> |

KS 800 replies:

STX	1	8	=	3	0	,	1	5	7	2	7	5	1	0	,	0	0	0	0	ETX	BCC
-----	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	-----	-----

| <code> | | <val> |

<val> = 30 means a KS800 is concerned
 15727510 \triangleq Software code no. 4012-157-27510
 0000 \triangleq Order no. 9407-480-00001

6.2 Principles of the function block protocol

A function block comprises input and output data (process data) as well as parameter and configuration data. It can be addressed via a block number. An allocated block type defines the relevant function.

The various access mechanisms are:

6.2.1 Single access

This access (code xx) can be used for reading or writing a single process value of a function. Single accesses to parameter and configuration data are not possible.

Example: (message structure with specified data)

Transmission of the absolute output value (Y_{man}) to controller channel 1.

Computer transmits data to KS800:

EOT	0	2	STX	3	2	,	5	0	,	4	=	5	0	ETX	BCC
-----	---	---	-----	---	---	---	---	---	---	---	---	---	---	-----	-----

| Addr | | code | | FB no. | | Fct-no. | | Value |

KS 800 replies:

ACK

 or

NAK

 in case of failure

6.2.2 Block access (tens block)

This access (code x0) can be used for reading max. nine process values of a function.

Example: (message structure with data request)

Reading of set-points (W_{nvol} and W_{vol}) from controller channel 3.

Computer requests:

EOT	0	2	3	0	,	5	3	,	1	ENQ
-----	---	---	---	---	---	---	---	---	---	-----

| Addr | Code | | FB-no. | | Fct-no. |

KS 800 replies:

STX	3	1	=	5	0	,	3	2	=	7	9	ETX	BCC
-----	---	---	---	---	---	---	---	---	---	---	---	-----	-----

| Code | | Value1 | | Code | | Value2 |

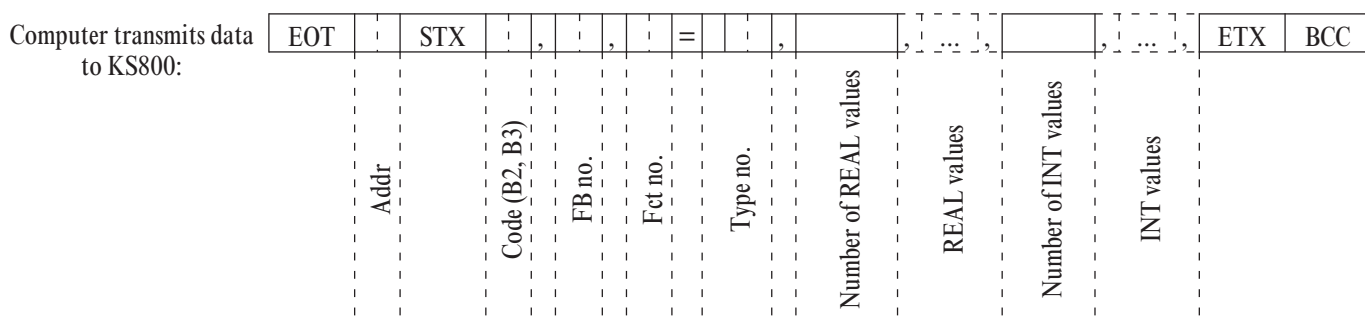
6.2.3 Block access (overall block)

This access can be used for reading and writing all parameter (code B2) and configuration data (code B3) of a function. The following conditions are applicable for this access:

- For writing data with 'Code B3', the unit must be switched to configuration mode (→ see page 14 'OpMod'). All new configuration data and parameters entered are only effective, when the unit was switched back to on-line.
- All data of a message must be defined, omissions are not permissible.
- If all parts of a message in the instrument are unused (HW and SW options), the complete message must be transmitted nevertheless. Checking the non-available data is omitted.
- With faulty block write accesses, the following rule is applicable: A message is answered with NAK, if min. one datum is faulty. Already valid values are stored.
- If the function number is omitted, function 0 (general) is addressed.

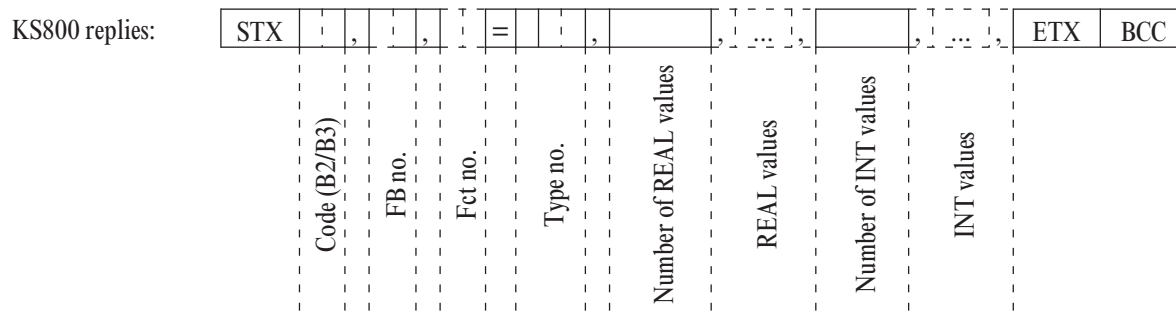
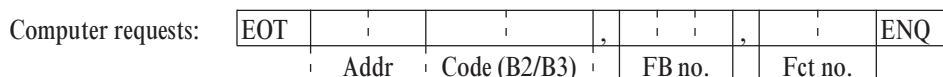
The general structure of a message with block accesses with code B2/B3 is shown below. The exact message structure (between *STX* and *ETX*) for the individual functions can be found below the relevant code table.

Message structure with data specification:



KS 800 replies: ACK or NAK in case of failure

Message structure with data request:



6.3 Message structure in function block protocol

6.3.1 INSTRUMENT

Message structure for function 'General'

Block access to configuration data											max. eff. length: 41 bytes												
STX	B3	,	0	,	0	=	0	,	0	,	5	,	C900	,	Adr1	,	C904	,	C902	,	Adr2	ETX	BCC

Message structure for function 'I/O connection'

Block access to configuration data											max. eff. length: 43 bytes												
STX	B3	,	0	,	2	=	0	,	1	,	HC100	,	4	,	C500	,	C530	,	C551	,	HCcycl	ETX	BCC

6.3.2 INPUT

Message structure for function 'ME/V1'

Block access to parameter data											max. eff. length: 44 bytes										
STX	B2	,	6x	,	1	=	112	,	4	,	X1in	,	X1out	,	X2in	,	X2out	,	0	ETX	BCC

Block access to configuration data											max. eff. length: 66 bytes																		
STX	B3	,	6x	,	1	=	112	,	5	,	X0	,	X100	,	XFail	,	Tfm	,	Tkref	,	3	,	C200	,	C205	,	C190	ETX	BCC

6.3.3 CONTR

Message structure for function 'General'

Block access to configuration data											max. eff. length: 36 bytes										
STX	B3	,	5x	,	0	=	91	,	0	,	4	,	C100	,	C101	,	C700	,	C180	ETX	BCC

Message structure for function 'Set-point'

Block access to parameter data											max. eff. length: 56 bytes														
STX	B2	,	5x	,	1	=	91	,	6	,	W0	,	W100	,	W2	,	Grw+	,	Grw-	,	Grw2	,	0	ETX	BCC

Message structure for function 'Algo'

Block access to parameter data											max. eff. length: 72 bytes																		
STX	B2	,	5x	,	3	=	91	,	8	,	Xsh	,	Tpuls	,	Tm	,	Xsd ₁	,	LW	,	Xsd ₂	,	Xsh ₁	,	Xsh ₂	,	0	ETX	BCC

Message structure for function 'Output variable'

Block access to parameter data											max. eff. length: 51 bytes												
STX	B2	,	5x	,	4	=	91	,	5	,	Ymin	,	Ymax	,	Y0	,	Yh	,	LYh	,	0	ETX	BCC

Message structure for function 'Tuning'

Block access to parameter data											max. eff. length: 49 bytes												
STX	B2	,	5x	,	5	=	91	,	4	,	YOptm	,	dYopt	,	OXsd	,	Trig1	,	1	,	POpt	ETX	BCC

Message structure for function 'Paramset x'

Block access to parameter data											max. eff. length: 72 bytes																		
STX	B2	,	5x	,	<6,7>	=	91	,	8	,	Xp1	,	Tn1	,	Tv1	,	T1	,	Xp2	,	Tn2	,	Tv2	,	T2	,	0	ETX	BCC

Message structure for function 'Start-up circuit'

Block access to parameter data											max. eff. length: 37 bytes								
STX	B2	,	5x	,	10	=	91	,	3	,	Ya	,	Wa	,	Tpa	,	0	ETX	BCC

6.3.4 ALARM

Message structure for function 'General'

Block access to parameter data											max. eff. lange: 58 bytes												
STX	B2	,	7x	,	0	=	46	,	6	,	LimL	,	LimH	,	xsd_1	,	LimLL	,	LimHH	,	LimHC	ETX	BCC

Block access to configuration data											max. eff. length: 36 bytes						
STX	B3	,	7x	,	0	=	46	,	0	,	2	,	C600	,	C601	ETX	BCC

7 Annex

7.1 Terms

FB	Abbr. of function block
Fkt	Abbr. of function
ET	Abbr. of Engineering Tool
Function	A self-contained partial function of the function block seen from the interface
Function block	Self-contained processing unit
HW	Abbr. f. hardware
ISO1745	Standard communication protocol ISO 1745, ASCII-based
PC interface	Front-panel interface on KSX controller for connecting an engineering tool
PCI	Process Control Instrument
PCI protocol	Protocol based ISO 1745, implemented for Philips controllers
RS422	Standard 4-wire interface, Full duplex, (EIA RS 422); in this case: separate send/receive channels with up to 32 units
RS485	Standard 2-wire connection, Half duplex, (EIA RS 485)
SW	Abbr. f. software
TTL	Signal level at module level

8 Index

	Neuer Index	
	A	
Address field		8
Addressing		6
	B	
Baud rate		6
	C	
Character format		7
Code.		8
	D	
Data format		6
Data request		7
Data sending		6
	F	
Function block number		9
Function block protocol		13
Block access (overall block)		29
Block access(tens block)		28
Single access		28
Function number		9
	P	
Parity		6
	R	
RS485/422 interface		5
	S	
Safety procedure		9
Standard protocol		11
System identification number		11
	T	
Terms		31
Transmission control characters		7

Subject to alterations without notice.
Bei Änderungen erfolgt keine Mitteilung.
Modifications sans avertissement réservés.

© PMA Prozeß- und Maschinen-Automation GmbH
Postfach 310 229, D - 34058 Kassel
Printed in Germany 9499 040 49418 (0106)



9499-040-49411

A4