

# HNC100

## Profibus-DP interface: Connection to Siemens S7 and S5

RE 30 131-P-03/02.03

User Manual





© 2003  
by Bosch Rexroth AG, Industrial Hydraulics, D-97813 Lohr am Main

All rights reserved. No part of this document may be reproduced or stored, processed, duplicated or circulated using electronic systems, in any form or by any means, without the prior written authorisation of Bosch Rexroth AG.  
In the event of contravention of the above provisions, the contravening party is obliged to pay compensation.

---

<b>1.</b>	<b>Introduction</b>	<b>4</b>
1.1	General	4
1.2	Profibus-DP address of the HNC 100	6
<b>2.</b>	<b>Transfer protocol</b>	<b>7</b>
2.1	Communication sequence	7
2.2	Format of transferred data	7
<b>3.</b>	<b>Master assemblies</b>	<b>11</b>
3.1	Overview	11
3.2	Master assembly IM 308-C (S5-115U - CPU: 941 - 944)	11
3.3	Master assembly IM 308-C (S5-115U - CPU: 945)	12
3.4	Master assembly IM 308-C (S5-115U - CPU: 928 and 948)	12
3.5	S5-95U with DP master interface	13
3.6	S7-300 and S7-400 with integrated DP master	13
3.7	Example of integrating an HNC100 into a Profibus-DP system under the COM PROFIBUS V3.0 software	14
<b>4.</b>	<b>Function and data blocks (S5-115U- CPU: 941 - 944)</b>	<b>17</b>
4.1	Data block DB 200	17
4.2	Function block FB 21 (initialization)	18
4.2.1	General	18
4.2.2	Parameterization of FB 21	19
4.2.3	Technical data	19
4.2.4	Application example	20
4.3	Function block FB 10 (communication block)	20
4.3.1	General	20
4.3.2	Parameterization of FB 10	20
4.3.3	Technical data	26
4.3.4	Application examples	26
	Fetching the value of an R-parameter from the HNC	26
	Further examples	27
<b>5.</b>	<b>Function and data blocks (S5-115U - CPU: 945)</b>	<b>28</b>
5.1	Data block DB 200	28
5.2	Function block FB 21 (initialization)	30
5.2.1	General	30
5.2.2	Technical data	30
5.3	Function block FB 10 (communication block)	31
5.3.1	General	31
5.3.2	Technical data	31
<b>6.</b>	<b>Function and data blocks (S5-135U- CPU: 928 and 948)</b>	<b>32</b>
6.1	Data block DB 200	32

6.2	Function block FB 21 (initialization) .....	34
6.2.1	General .....	34
6.2.2	Technical data .....	34
6.3	Function block FB 10 (communication block) .....	35
6.3.1	General .....	35
6.3.2	Technical data .....	35
<b>7.</b>	<b>Function and data blocks (for S5-95U with DP master interface) .....</b>	<b>36</b>
7.1	Data block DB 200 .....	36
7.2	Function block FB 21 (initialization) .....	37
7.2.1	General .....	37
7.2.2	Technical data .....	38
7.3	Function block FB 10 (communication block) .....	38
7.3.1	General .....	38
7.3.2	Technical data .....	38
<b>8.</b>	<b>Functions and data blocks (S7-300 and S7-400 with DP master) .....</b>	<b>39</b>
8.1	Data block DB 10 or DB xy .....	39
8.2	Function FC 21 (initialization).....	41
8.2.1	General .....	41
8.2.2	Parameterization of FC 21 .....	42
8.2.3	Technical data .....	42
8.2.4	Application example .....	43
8.3	Function FC 22 (initialization).....	43
8.3.1	General .....	43
8.3.2	Parameterization of FC 22 .....	43
8.3.3	Technical data .....	44
8.3.4	Application example .....	44
8.4	Function FC 10 (communication function) .....	44
8.4.1	General .....	44
8.4.2	Parameterization of FC 10 .....	45
8.4.3	Technical data .....	51
8.4.4	Application examples .....	51
	Fetching a process datum from the HNC .....	51
	Further examples .....	52
8.5	Function FC 11 (communication function) .....	53
8.5.1	General .....	53
8.5.2	Parameterization of FC 11 .....	53
8.5.3	Technical data .....	60
8.5.4	Application examples .....	61
	Fetching a process datum from the HNC .....	61
	Fetching the value of an R-parameter from the HNC .....	62
	Sending the value of an R-parameter to the HNC.....	62

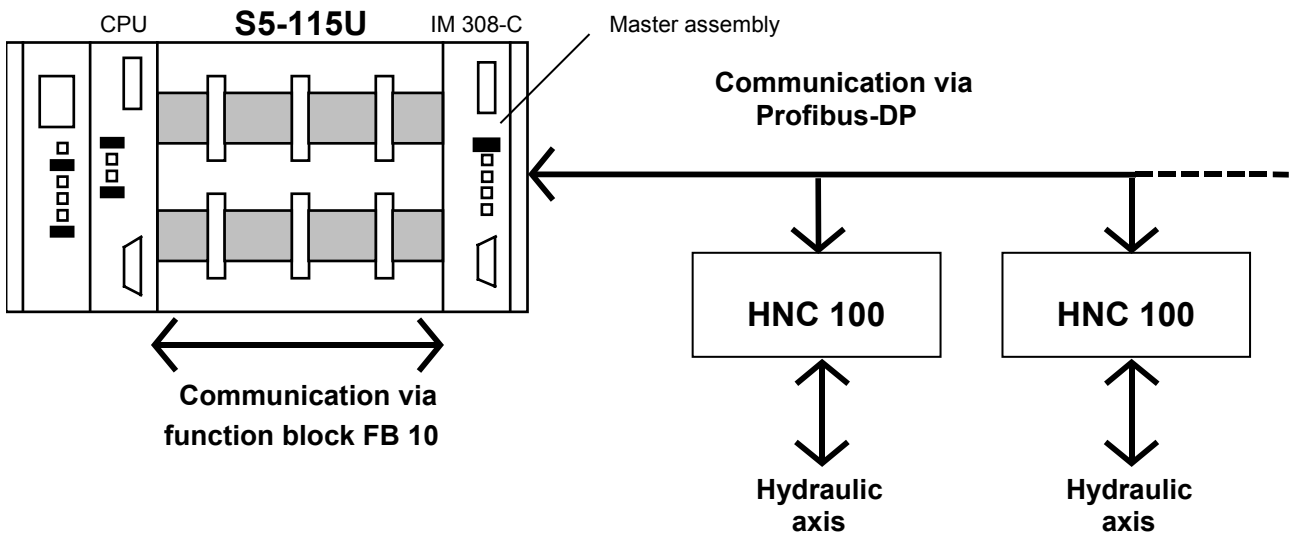
---

Fetching the value of a machine datum from the HNC .....	62
Fetching the value of a C-variable from the HNC.....	63
Sending the value of a C-variable to the HNC .....	63
Fetching the value of a B-variable from the HNC.....	63
Sending the value of a B-variable to the HNC.....	64
Fetching digital inputs from the HNC.....	64
Setting digital inputs on the HNC .....	64
Resetting digital inputs on the HNC .....	65
Fetching digital outputs from the HNC .....	65
Fetching the status of a flag from the HNC .....	66
Setting a flag on the HNC.....	66
Resetting a flag on the HNC.....	66
Fetching the x- or y-value of a curve point from the HNC .....	67
Setting the x- or y-value of a curve point on the HNC .....	67
Sending a complete curve to the HNC .....	67
<b>9.    Diagnosis data from the HNC 100.....</b>	<b>69</b>
9.1    Structure of diagnosis data .....	69
9.2    Fetching diagnosis data (IM 308-C) .....	70
9.3    Fetching diagnosis data (S5-95U).....	71
9.4    Fetching diagnosis data (S7-300 and S7-400).....	71
<b>10.   Time.....</b>	<b>72</b>
10.1   Bus cycle time (Profibus-DP) .....	72
10.2   Processing time of the function blocks.....	73
10.3   Time required for processing the data on the HNC 100.....	73
<b>11.   GSE file (component master file).....</b>	<b>74</b>
<b>12.   Annex .....</b>	<b>78</b>
12.1   Software .....	78
12.1.1 Software for S5-115U (CPU: 941 - 944).....	78
12.1.2 Software for S5-115U (CPU: 945).....	78
12.1.3 Software for S5-115U (CPU: 928 and 948).....	79
12.1.4 Software for S5-95U with DP master interface.....	79
12.1.5 Software for S7-300 and S7-400 with integrated DP master .....	79

# 1. Introduction

## 1.1 General

An optional extra of the HNC 100 is a Profibus-DP interface. This offers the possibility of exchanging data between the individual HNCs and a higher-level control via a Profibus-DP system. For this, the higher-level control must be fitted with a Profibus-DP master. An example of such a Profibus-DP system is illustrated in the figure below.



Example of Profibus-DP system structure

### Example

In the example shown, the HNC 100 is coupled via the Profibus-DP to the S5-115U, which is provided with an IM 308-C master assembly. The IM 308-C and the CPU of the S5 communicate via the peripheral bus, which electrically interconnects all assemblies that exist on the S5.

On the HNC side, the connection to the Profibus-DP is realized by means of an integrated DP slave node. Function and data blocks are used for the data exchange between the CPU of the S5 and the individual HNC 100s.

### Hardware requirements

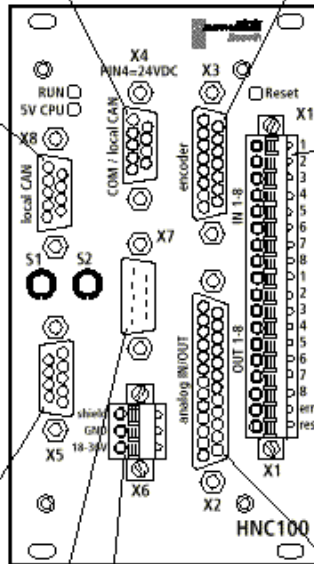
To enable the connection of the HNC 100 to a Profibus-DP system at all, the HNC 100 must be fitted with a Profibus-DP interface. The HNC 100 is connected to the Profibus-DP using a screened, twisted two-wire cable. The Profibus-DP interface of the HNC 100 is made available on connector **X5** ( see the following connector assignment of the HNC 100 ). For further information on the structure of a Profibus-DP system, please refer to the **master assembly manual** and **DIN standard 19245 part 3**.

X8: local CAN	
Pin 1	CAN_GND
2	res
3	res
4	res
5	res
6	res
7	res
8	CAN_H
9	CAN_L

X4: COM / local CAN	
Pin 1	CAN_GND
2	TxD
3	CTS
4	24 VN
5	0 VN
6	RxD
7	RTS
8	CAN_H
9	CAN_L

X3: encoder		
Pin	incremental	SSI
1	/Ua 2	
2		Pulse
3	Ua 0	
4	/Ua 0	
5	Ua 1	Data
6	/Ua 1	/Data
7		/Pulse
8	Ua 2	
9	res	
10	0 VN	
11	res	
12	5 VTTL (max. 200 mA)	
13	res	
14	24 VN (max. 450 mA)	
15	res	

S1, S2 ... Address, Baud rate CAN



X1: digital I/O	
Pin 1	IN1
2	IN2
3	IN3
4	IN4
5	IN5
6	IN6
7	IN7
8	IN8
9	OUT1
10	OUT2
11	OUT3
12	OUT4
13	OUT5
14	OUT6
15	OUT7
16	OUT8
17	/error
18	res

X5:		
Pin	Profibus DP	INTERBUS-S (OUT)
1	n.c.	+ DO 2
2	n.c.	+ DI 2
3	RxD / TxD-P	GND 2
4	CNTR-P	n.c.
5	DGND	U <sub>dd</sub>
6	VP	- DO 2
7	n.c.	- DI 2
8	RxD / TxD-N	n.c.
9	n.c.	BCI

X6: voltage supply	
Pin 1	shield
2	GND
3	18 - 36 VDC

X2: analog IN / OUT		
Pin 1	U <sub>in</sub> 1 +	I <sub>in</sub> 1 -
2	U <sub>in</sub> 1 -	
3	U <sub>in</sub> 2 +	I <sub>in</sub> 2 -
4	U <sub>in</sub> 2 -	
5	U <sub>in</sub> 3 +	I <sub>in</sub> 3 -
6	U <sub>in</sub> 3 -	
7	U <sub>in</sub> 4 +	I <sub>in</sub> 4 -
8	U <sub>in</sub> 4 -	
9	I <sub>out</sub> 2	
10	U <sub>out</sub> 2	
11	analog_GND	
12	U <sub>ref</sub> = + 10 V	
13	U <sub>ref</sub> = - 10 V	
14	I <sub>out</sub> 1	
15	U <sub>out</sub> 1	
16	U <sub>out</sub> 3	
17	U <sub>out</sub> 4	
18		I <sub>in</sub> 1 +
19		I <sub>in</sub> 2 +
20		I <sub>in</sub> 3 +
21		I <sub>in</sub> 4 +
22	U <sub>imp</sub> 1	
23	U <sub>imp</sub> 2	
24	U <sub>imp</sub> 3	
25	U <sub>imp</sub> 4	

X7:			
Pin	CANopen	inductive	INTERBUS-S (IN)
1	n.c.	Supply 1 +	+ DO1
2	CAN_L	Supply 1 -	+ DI1
3	CAN_GND	Signal 1 +	GND1
4	n.c.	Signal 1 -	n.c.
5	n.c.	Supply 2 +	n.c.
6	n.c.	Supply 2 -	- DO1
7	CAN_H	Signal 2 +	- DI1
8	n.c.	Signal 2 -	n.c.
9	n.c.	Sync IN/OUT	n.c.

Connector assignment of the HNC100

## Software requirements

The relevant software drivers for the Profibus-DP interface are integrated in the HNC 100 only in software version <b>4_02m00n00d00x00k00v00</b> or higher.
---

## 1.2 Profibus-DP address of the HNC 100

The address of the HNC 100 is set using the **address switches ( S1 and S2 )** and **machine datum M410 ( offset HNC address )**. The address of the HNC 100 results from the sum of addresses, which were set using the address switches, and the value in machine datum M410 ( offset HNC address ). If, for example, address 10 was set on the address switches and the machine datum contains the value "2", the resulting address of the HNC 100 will be "12".

Address switches S1 and S2 allow the setting of the addresses 1 to 32. If an address higher than 32 is to be assigned to the HNC 100, a corresponding value must be entered in machine datum M410 ( offset HNC address ).

The following assignment is valid for address switches S1 and S2:

S1	S2	Address of HNC100:
0	0	1
1	0	2
2	0	3
..	..	..
9	0	10
A	0	11
B	0	12
C	0	13
D	0	14
E	0	15
F	0	16
0	1	17
1	1	18
2	1	19
..	..	..
9	1	26
A	1	27
B	1	28
C	1	29
D	1	30
E	1	31
F	1	32

The value of machine datum M410 ( offset of HNC address ) is set in the WIN-PED program, program part "Machine data". Select menu command **Edit > Change** in the machine data program part. Then open the group "**General**". This group includes machine datum M410.

## Remark

If the address of the HNC 100 is changed via address switches S1 and S2 or via machine datum M410, the HNC 100 takes the new address over only after a reset was carried out on the HNC 100.



## 2. Transfer protocol

### 2.1 Communication sequence

#### General

The data connection between the master assembly and the HNC 100 is established automatically when the operating voltage is switched on. The master assembly addresses all slaves ( HNC 100 ) connected to the Profibus-DP system and checks them for readiness for operation ( operating voltage applied ). When the individual DP slaves signal their readiness for operation, they are parameterized and configured by the DP master. After this the master assembly starts to exchange user data with the DP slaves on a cyclical basis.

The Profibus-DP does not provide any determinations with regard to contents and interpretation of data. This means that each supplier of a Profibus-DP slave is responsible for determining the format of the data to be transferred.

For the HNC 100 it was specified that always 8 bytes of data are transferred. These 8 bytes include the coding of the function which is to be executed by the HNC 100.

### 2.2 Format of transferred data

The HNC 100 and the DP master exchange data blocks having a length of 8 bytes. With the help of these 8 bytes it can be determined which function is to be executed by the HNC 100. The table below shows an overview of the functions which can currently be executed by the HNC 100.

As can be seen from the overview table, the first byte in the data block, the so-called identification byte, defines, which function is to be executed by the HNC 100. In the subsequent byte, the complementary identification byte, 3 bits are of importance: the error indication bit ( f ), the synchronization bit ( y ) and the complementary identification bit ( z )

Contents of the data block	Direction of data	ID byte (1 <sup>st</sup> byte)	CID byte (2 <sup>nd</sup> byte)	3 <sup>rd</sup> byte	4 <sup>th</sup> byte	5 <sup>th</sup> byte	6 <sup>h</sup> byte	7 <sup>th</sup> byte	8 <sup>th</sup> byte
R-parameter	Read	1 0 0 0 0 a a a	f 0 0 0 0 0 y z	Number		Value			
	Write	0 0 0 0 0 a a a							
Machine data	Read	1 0 0 0 1 a a a	f 0 0 0 0 0 y z	Number		Value			
	Write	0 0 0 0 1 a a a							
C-variables	Read	1 0 0 1 0 a a a	f 0 0 0 0 0 y z	Num-ber	Value				Free
	Write	0 0 0 1 0 a a a							
B-variables	Read	1 0 0 1 1 x x x	f 0 0 0 0 0 y z	Num-ber	Value				Free
	Write	0 0 0 1 1 x x x							
Process data	Read	1 0 1 0 0 x x x	f 0 0 0 0 0 y z	Num-ber	Value				Free
	Write	Not permitted							
Digital inputs	Read	1 0 1 0 1 x k k	f 0 0 0 0 0 y z	Inputs ( bit-coded )				Free	
	Write *)	0 0 1 0 1 s k k		8 - 1	16 - 9	24 - 17	32 - 25		
Digital outputs	Read	1 0 1 1 0 x k k	f 0 0 0 0 0 y z	Outputs ( bit-coded )				Free	
	Write	0 0 1 1 0 s k k		8 - 1	16 - 9	24 - 17	32 - 25		
Flags	Read	1 0 1 1 1 x x x	f 0 0 0 0 0 y z	Number		Value	Free		
	Write	0 0 1 1 1 s x x							
Curve points	Read	1 1 0 1 1 x x w	f 0 0 0 0 0 y z	Curve number ( 0 - 15 )	Number of the curve point ( 0 - 4095 )		x- or y-value of the curve point		
	Write	0 1 0 1 1 x x w							

aaa: 001 = Axis 1      kk: 00 = I/O card 1      s: 0 = Delete bit on the HNC      xxx: Bits are ignored  
       010 = Axis 2         01 = I/O card 2         1 = Set bit on the HNC         f: Error indication bit  
       100 = Auxiliary axis    10 = I/O card 3         w: 0 = x-value of a curve point    y: Synchronization bit  
                                   11 = I/O card 4            1 = y-value of a curve point      z: Complementary identification bit

\*) Setting the digital inputs of I/O card 1 is not permitted.

**Error indication bit ( f )**

The error indication bit is set on the HNC 100 during the evaluation of the data block, if an error is present on the HNC 100 ( e.g. encoder error or traversing range limit ). The exact cause of the error can be determined with the help of the diagnosis data provided. How to request and interpret these diagnosis data is explained in more detail in the chapter "Diagnosis data of the HNC 100".

**Complementary identification bit ( z )**

The complementary identification bit signals the HNC 100 whether the user data block contains new data. These have to be changed from 0 to 1 or vice versa, if new user data are to be transferred. If the HNC 100 recognizes that data are new, these will be evaluated accordingly. Otherwise, the data will not be evaluated by the HNC 100. The complementary identification bit is only required because the user data transfer via the Profibus-DP is cyclical.

**Synchronization bit ( y )**

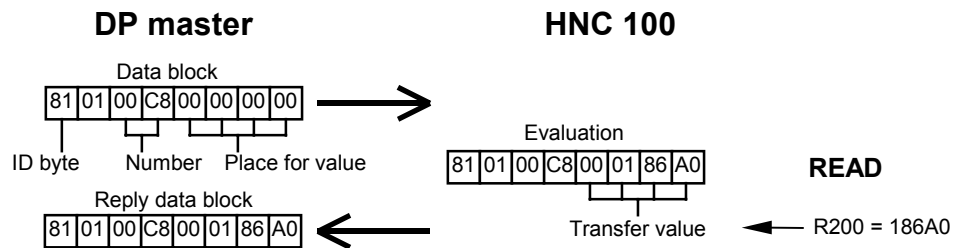
The "value" of the complementary identification bit is monitored by the HNC 100. This means that the synchronization bit is set in the HNC 100, if the HNC 100 receives new user data via the Profibus-DP, but the complementary identification bit has a wrong value. In this case, the HNC 100 does not evaluate the data.

Direction of data

In the "read" direction of data, values, inputs or outputs are fetched from the HNC 100. In the "write" direction, values, inputs or outputs are changed on the HNC 100. Some examples are to illustrate the format of transferred data.

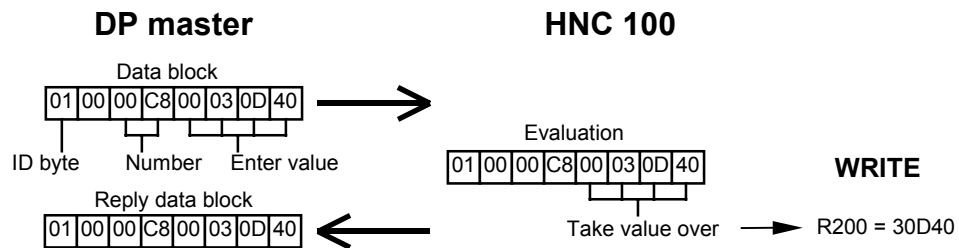
Example: Read

If the value of the R-parameter 200 ( axis 1 ) is to be read, a data block with the corresponding identification byte ( = 81H ) and the number of the R-parameter ( = 00C8H ) must be passed on to the DP master. The bytes, which are used for the value, initially contain the value zero. This user data block is transferred via the Profibus-DP to the HNC 100. There, the user data block is evaluated and "filled" with the value of the R-parameter. The user data block with the R-parameter is resent to the DP master via the Profibus-DP, where the value of the R-parameter will be available.



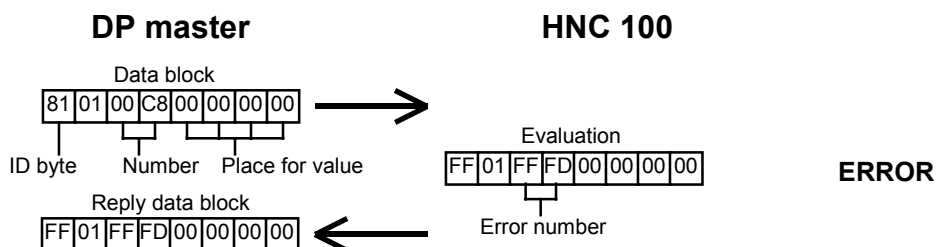
Example: Write

If the value of R-parameter 200 is to be changed, the DP master must transfer a data block. In this situation, the bytes contain a number which relates to the value. The HNC evaluates the data block and transfers the data block unchanged as an acknowledgement.



Example: Error

If the HNC 100 detects an error while evaluating the data block ( e.g. R-parameter is not defined ), the HNC 100 overwrites the identification byte with "FF" and assigns a corresponding error number to bytes 3 and 4. Then, it transfers this modified data block.



The following error numbers are fed back by the HNC 100 in the case of an error.

FFAC	The curve point could not be sent, because the curve is being updated (-> nc command: KURVE ... UPDATE) or new curve points are received over the serial interface.
FFB1	Sending of a complete curve has not been started
FFB2	x-value for the curve point is not permitted
FFB3	Curve point is not defined
FFCC	Process datum is not defined
FFD1	Function is not defined ( transfer protocol )
FFD2	Invalid flag number
FFD3	Writing of a process datum is impossible
FFD4	Writing of digital inputs is impossible
FFD5	Invalid B-variable number
FFD6	Invalid axis number specified
FFD7	Invalid C-variable number
FFD8	Value of the machine datum too small
FFD9	Value of the machine datum too great
FFDA	Value of the R-parameter too small
FFDB	Value of the R-parameter too great
FFE4	Curve is not defined
FFFC	Machine datum not defined
FFFD	R-parameter not defined

#### Remark

To facilitate the user's accessing the HNC 100 via the Profibus-DP, function blocks ( FB 10 and FB 21 ) are available for master assembly IM 308-C ( S5 assembly ), or ( FC 10 and FC 21 ) for the S7 functions, which use the described data format.

### 3. Master assemblies

#### 3.1 Overview

At present, Bosch Rexroth offers drivers ( function blocks ) for the following master assemblies to make the access to the HNC 100 via the Profibus-DP easier for the user. The drivers are provided on a floppy disk ( see also annex - chapter: Software ).

Master assembly	Name of the function blocks	File name
IM 308-C for S5-115U ( CPU 941 - 944 )	FB 10 ( KODP115A ) FB 21 ( KODP115A ) DB 200	F010PAST.S5D F021PAST.S5D D200PAST.S5D
IM 308-C for S5-115U ( CPU 945 )	FB 10 ( KODP945 ) FB 21 ( INDP945 ) DB 200	F010P4ST.S5D F021P4ST.S5D D200P4ST.S5D
IM 308-C for S5-135U ( CPU 928 and CPU 948 )	FB 10 ( KODP135A ) FB 21 ( INDP135A ) DB 200	F010P3ST.S5D F021P3ST.S5D D200P3ST.S5D
S5-95U with DP master interface	FB 10 ( KODP095U ) FB 21 ( INDP095U ) DB 200	F010P9ST.S5D F021P9ST.S5D D200P9ST.S5D
S7-300 and S7-400 with integrated DP master	FC 10 ( KODPS7 ) FC 21 ( INDPS7 ) FC 11 ( KODPS711 ) FC 22 ( INDPS722 ) DB 10	Library: HNC_S7

#### 3.2 Master assembly IM 308-C (S5-115U - CPU: 941 - 944)

The IM 308-C master assembly can be parameterized and configured with the help of the COM PROFIBUS software by Siemens. It must be noted here that the CPU of the S5 and the master assemblies communicate via the **I/O area** with **linear addressing**.

When parameterizing the master and configuring the individual DP slaves ( HNC 100 ), care must be taken that **linear addressing** is set.

The FB 10 function block was developed for this type of addressing. If another type of addressing is to be employed, you **cannot** use the FB 10 function block.

In addition, in conjunction with the configuration, make sure that the input and output data for the HNC 100 on the IM 308-C are read or written via the **same addresses**. If this requirement is not met, the FB 10 function block **cannot** be used.

With the IM 308-C, the component master file **MMRR0476.GSD** must be used for the configuration of the HNC 100 ( see annex - chapter: Software ). This file must be copied into the relevant subdirectory for component master files of the COM PROFIBUS program. Only then can the HNC 100 be configured as Profibus-DP slave in the COM PROFIBUS program. The HNC 100 is listed in the group "NC".

When configuring the HNC 100, the following configurations have to be selected.

**S5 <- HNC: 8-Byte-Standard**  
**S5 -> HNC: 8-Byte-Standard**

### 3.3 Master assembly IM 308-C (S5-115U - CPU: 945)

Parameterize and configure the master assembly IM 308-C using the COM PROFIBUS software by Siemens. It must be noted here that the CPU of the S5 and the master assembly communicate via the **P area** or **Q area**.

“**Linear**” addressing must be firmly set in the COM PROFIBUS software. The FB 10 function block was developed for this type of addressing. If another type of addressing is to be employed, you **cannot** use the FB 10 function block.

In addition, in conjunction with the configuration, make sure that the input and output data for the HNC 100 on the IM 308-C are read or written via the **same addresses**. If this requirement is not met, the FB 10 function block **cannot** be used.

With the IM 308-C, the component master file **MMRR0476.GSD** must be used for the configuration of the HNC 100 ( see annex - chapter: Software ). This file must be copied into the relevant subdirectory for component master files of the COM PROFIBUS program. Only then can the HNC 100 be configured as Profibus-DP slave in the COM PROFIBUS program. The HNC 100 is listed in the group “NC”.

When configuring the HNC 100, the following configurations have to be selected.

**S5 <- HNC: 8-Byte-Standard**  
**S5 -> HNC: 8-Byte-Standard**

### 3.4 Master assembly IM 308-C (S5-115U - CPU: 928 and 948)

Parameterize and configure the master assembly IM 308-C using the COM PROFIBUS software by Siemens. It must be noted here that the CPU of the S5 and the master assembly communicate via the **P area** or **Q area**.

“**Linear**” addressing must be firmly set in the COM PROFIBUS software. The FB 10 function block was developed for this type of addressing. If another type of addressing is to be employed, you **cannot** use the FB 10 function block.

In addition, in conjunction with the configuration, make sure that the input and output data for the HNC 100 on the IM 308-C are read or written via the **same addresses**. If this requirement is not met, the function block FB 10 **cannot** be used.

With the IM 308-C, the component master file **MMRR0476.GSD** must be used for the configuration of the HNC 100 ( see annex - chapter: Software ). This file must be copied into the relevant subdirectory for component master files of the COM PROFIBUS program. Only then can the HNC 100 be configured as Profibus-DP slave in the COM PROFIBUS program. The HNC 100 is listed in the group “NC”.

When configuring the HNC 100, the following configurations have to be selected.

**S5 <- HNC: 8-Byte-Standard**  
**S5 -> HNC: 8-Byte-Standard**

### 3.5 S5-95U with DP master interface

Parameterize and configure the DP master interface of the S5-95U using the COM PROFIBUS software by Siemens. Make sure that “**linear**” addressing is firmly set in the COM PROFIBUS software.

In addition, in conjunction with the configuration of the individual HNC100s, make sure that the input and output data for the HNC 100 are read or written via the **same I/O addresses**. If this requirement is not met, the FB 10 function block **cannot** be used.

Moreover, when configuring the HNC 100, take into account that the input and output data of the HNC 100 must be **completely** within the **I/O address area 64 to 127** or in the **I/O address area 128 to 191**. Overlapping of these areas within the input and output data of the HNC 100 **must** be avoided. If overlapping is required, the FB 10 function block **cannot** be used.

With the S5-95U with DP master interface, the component master file **MMRR0476.GSD** must be used for the configuration of the HNC 100 ( see annex - chapter: Software ). This file must be copied into the relevant subdirectory for component master files of the COM PROFIBUS program. Only then can the HNC 100 be configured as Profibus-DP slave in the COM PROFIBUS program. The HNC 100 is listed in the group “NC”.

When configuring the HNC 100, the following configurations have to be selected.

**S5 <- HNC: 8-Byte-Standard**  
**S5 -> HNC: 8-Byte-Standard**

### 3.6 S7-300 and S7-400 with integrated DP master

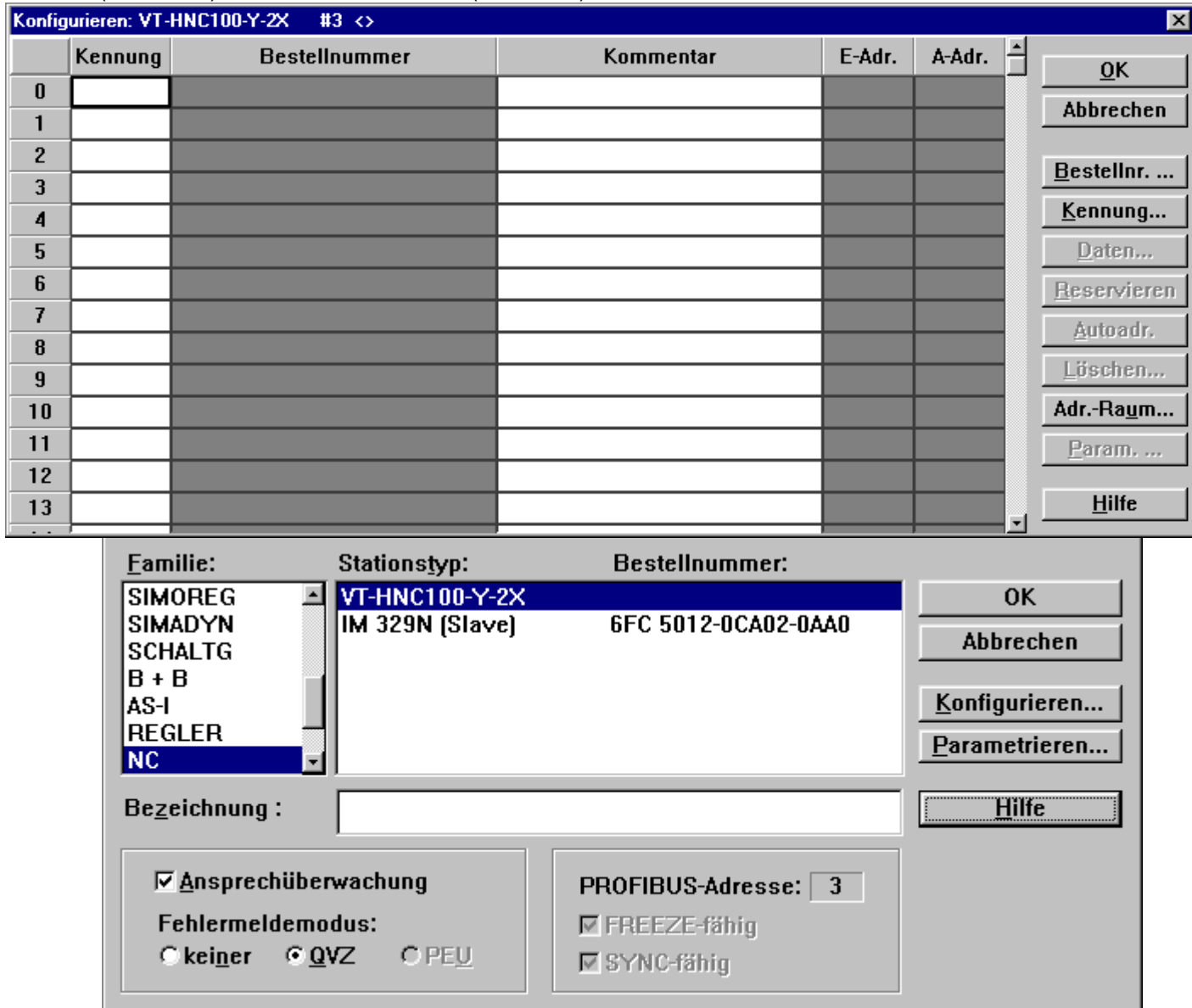
Parameterize and configure the DP master on the S7-300 and S7-400 using the STEP7 software by Siemens.

In addition, in conjunction with the configuration of the individual HNC 100s, make sure that the input and output data for the HNC 100 are read or written via the **same I/O addresses**. If this requirement is not met, the FB 10 function block **cannot** be used.

With the S7-300 and S7-400 with integrated DP master, the component master file **MMRR0476.GSD** must be used for the configuration of the HNC 100 ( see annex - chapter: Software ). This file must be copied into the relevant subdirectory for component master files of the STEP7 program. Only then can the HNC 100 be configured as Profibus-DP slave in the STEP7 program. The HNC 100 is listed in the group “NC”.

When configuring the HNC 100, the following configurations have to be selected.

**S7 <- HNC: 8-Byte-Standard**  
**S7 -> HNC: 8-Byte-Standard**



### 3.7 Example of integrating an HNC100 into a Profibus-DP system under the COM PROFIBUS V3.0 software

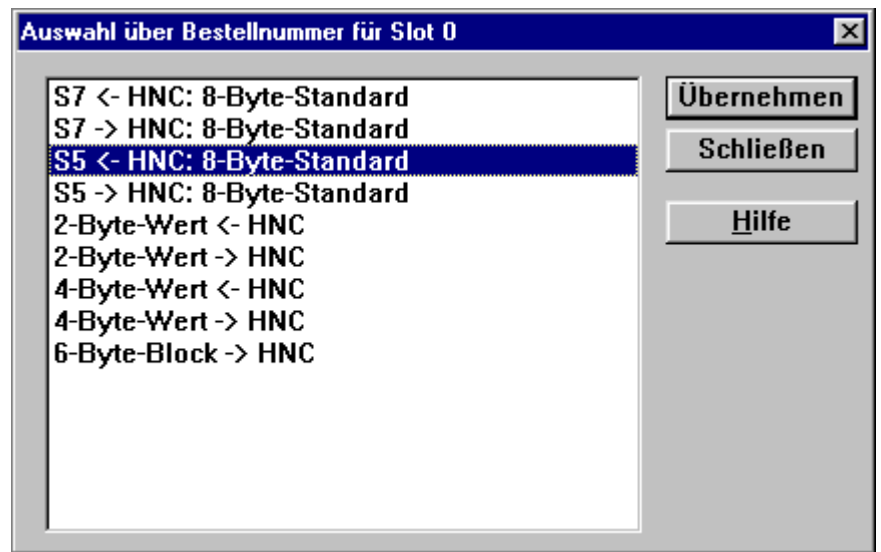
In order that the HNC 100 appears in the group "NC", you must copy the component master file of the HNC 100 (mmrr0476.gsd) into the subdirectory **IGSD** of the COM PROFIBUS program. You can read in the component master file of the HNC 100 and include the HNC 100 as Profibus-DP slave in the group "NC" by selecting menu item **Datei > GSD-Dateien einlesen**. Now you can open a new or existing master system, into which the HNC 100 can be included as DP slave. For adding the HNC 100, follow the steps below:

Select the menu item **Projektieren > Neuer Slave...**. A dialog will appear, in which you have to enter the Profibus address of the HNC 100. After having entered the Profibus address, click the OK button. The dialog **Slaveeigenschaften** will open.

In this dialog, you have to select the **NC** family and **VT-HNC100-Y-2X** (= HNC 100) as station type. Then click the **Konfigurieren** button to open the dialog **Konfigurieren**.

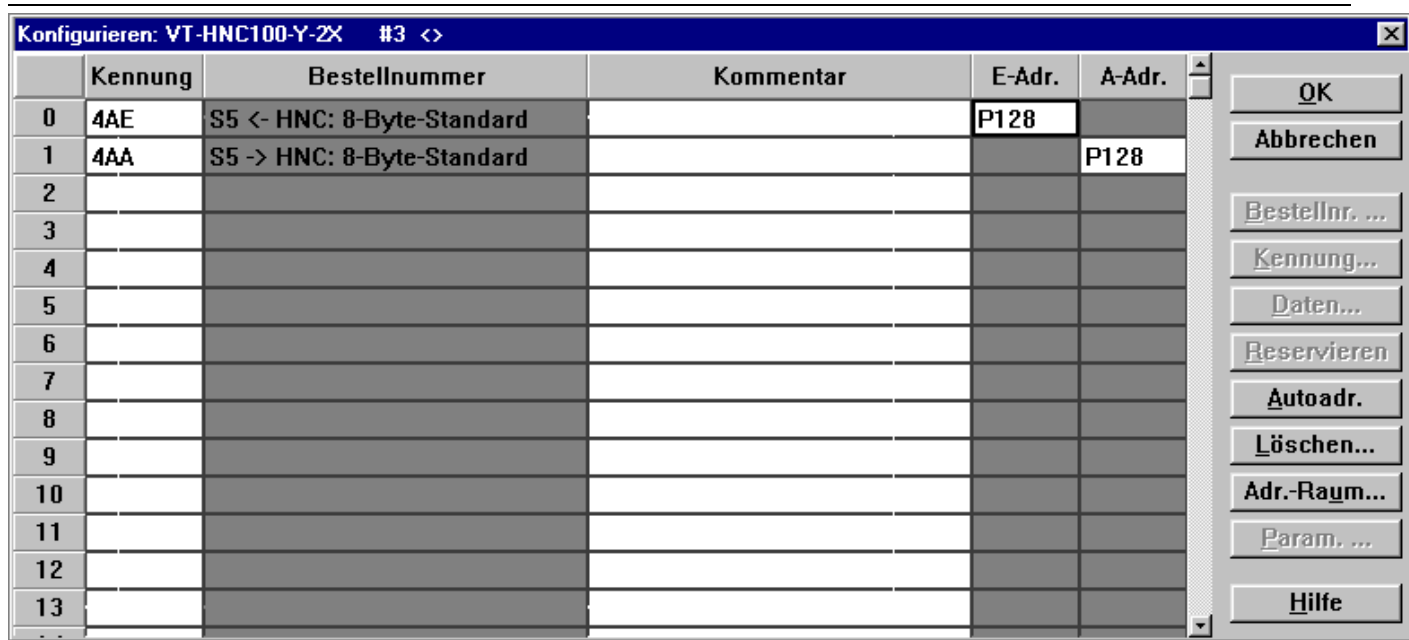


Click the button **BestellNr** in this dialog to open the list with the available configurations.



Insert the configuration **S5 <- HNC: 8-Byte-Standard** in the slot 0 and the configuration **S5 -> HNC: 8-Byte-Standard** in the slot 1.

After selecting the configurations click the button **Schliessen** to open the dialog **Konfigurieren** again.

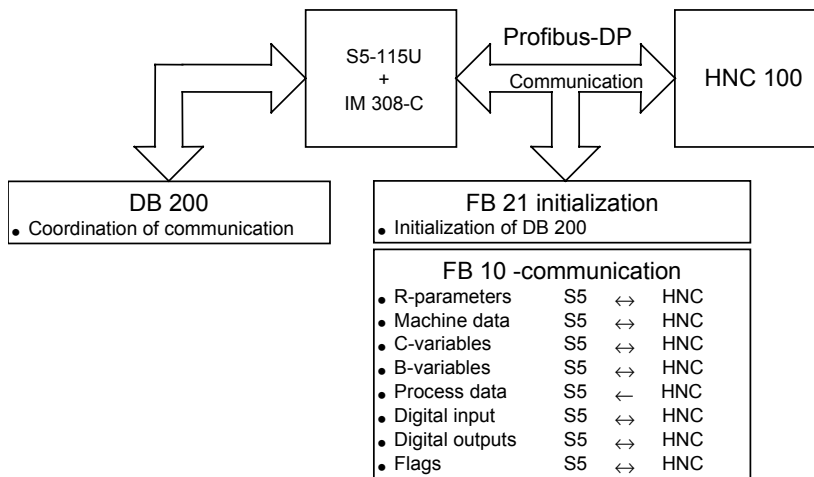


In this dialog, you have to enter the starting address of the address range, via which data are sent to the HNC 100 and also received from the HNC 100. Please note that the input and output addresses must be identical, since this is assumed by the function blocks.

After having entered the addresses, click the OK button. Thus, the inclusion of an HNC 100 as DP slave in a master system is completed.

## 4. Function and data blocks (S5-115U- CPU: 941 - 944)

To facilitate the user's access to the individual HNCs via the Profibus-DP with the IM 308-C Bosch Rexroth created function blocks ( FB 10 and FB 21 ). The user simply must integrate these blocks into his S5 program and call them with suitable transfer parameters.



### 4.1 Data block DB 200

The user must save in data block DB 200 the addresses on the IM 308-C, via which the user data can be transferred to the individual HNC 100s ( determination of the addresses with the help of the COM PROFIBUS program ). In addition, certain data and states can be saved in DB 200 in order to coordinate the communication via the Profibus-DP.

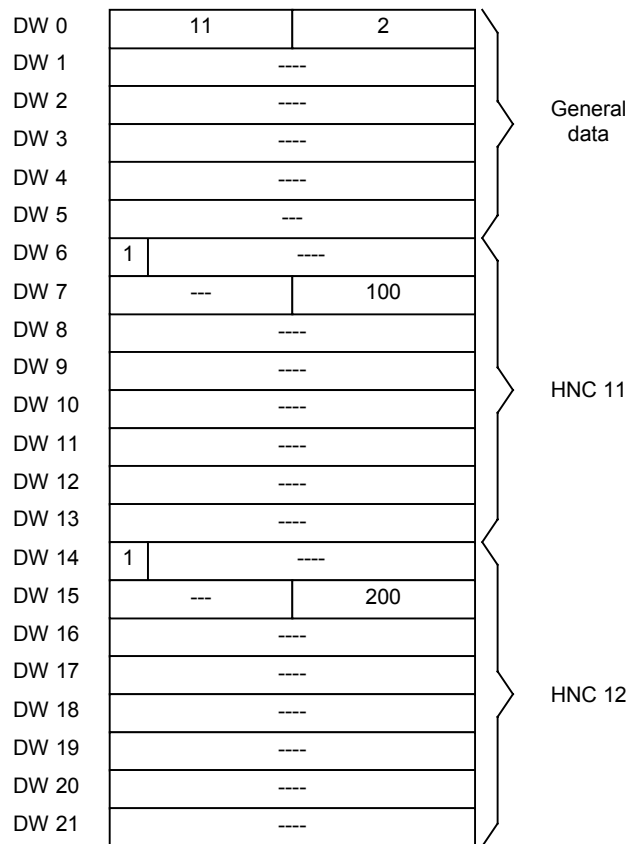
The length of data block DB 200 depends on the number of HNCs, which are connected to the IM 308-C via the Profibus-DP. 8 data words are required for each HNC 100. The first 6 data words in the DB 200 are required for saving general data. The first data word is used for saving the address of the 1<sup>st</sup> HNC and the total number of HNCs. For an HNC 100 that is coupled to the Profibus-DP, **bit L2** must be set additionally **equal to one**.

DW 0	Address of 1st HNC	Number of HNCs	General data
DW 1	Reserved		
DW 2	Reserved		
DW 3	Reserved		
DW 4	Reserved		
DW 5	Reserved		
DW 6	L2	Reserved	1 <sup>st</sup> HNC
DW 7	Reserved	Address	
DW 8	Reserved		
DW 9	Reserved		
DW 10	Reserved		
DW 11	Reserved		
DW 12	Reserved		
DW 13	Reserved		
DW 14	Further HNCs		
DW 15			

Example

The HNCs with the **addresses 11** and **12** are connected to the Profibus-DP. This results in a length of 22 data words for data block DB200 ( 6 data words for general data, 16 data words for the two HNCs ). Consequently, DB200 must include data word DW 0 through data word DW 21. In the first data byte of the DB 200, **11** must be entered, since the address of the first HNC is "11". In the second data byte, **2** must be entered, since two HNCs are available.

With the COM PROFIBUS software, it was determined that data are exchanged with the individual HNC 100s via **addresses 100 to 107 ( P-area )** and **200 to 207 ( P-area )**. This address range is used for both transferring data to the HNC 100 and reading data that are received from the HNC 100. The user must enter the addresses for the individual HNC 100s that were determined using the COM PROFIBUS software as illustrated in the example below.



Example of DB 200 assignment by the user

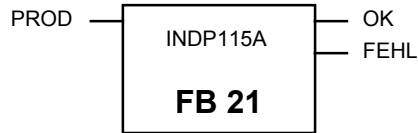
## 4.2 Function block FB 21 (initialization)

### 4.2.1 General

Function block FB 21 should be called **once** in the start-up blocks OB 21 and OB 22. The use of FB 21 should be avoided in cyclical program processing ( OB 1 ).

The main task of function block FB 21 is to initialize all data words in data block DB 200, whose values do not need to be entered by the user.

### 4.2.2 Parameterization of FB 21



Parameter	Description	Nature	Type
PROD	Product data management	E	W
OK	Indication bit "OK"	A	BI
FEHL	Error output	A	W

Transfer parameter of FB 21

**PROD** (product data management) At present, the transfer parameter PROD is irrelevant. It can have any value.

**OK** ( error-free execution ) This bit provides information on whether an error has occurred during the processing of function block FB 21. If the value of this parameter is 1, FB 21 was processed without any error. However, if the content of this bit is zero, an error has occurred during processing of this function block. In this case, the associated error number is saved in the transfer parameter "FEHL".

**FEHL** ( error ) This transfer parameter includes an error number, if the value of the OK bit is zero. The table below shows an overview of possible errors with the associated numbers.

Error number	Meaning
F00B	The values of data bytes DL 0 and DR 0 are not permitted in DB 200.

Overview of possible errors with FB 21

**Caution:**

**No data word** must be used for the transfer parameter FEHL, because data block DB 200 is called in FB 21.

### 4.2.3 Technical data

Block number	FB 21
Block name	INDP115A
Nesting depth	1
Assigned data blocks	DB 200
Assigned flags	MB 200 - 205
Assigned timers <sup>1)</sup>	T ( HNC addresses )
Average operating time (S5-115U, length of DB200: 22 DWs)	CPU 943: 3 ms CPU 944: 0.5 ms

1) For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S5-115U. Thus, timers T5 and T12 are assigned.

#### 4.2.4 Application example

```

0003      :
0004      :L   KH 0000           Initialize error flag word
0006      :T   MW 12
0007      :
0008      :SPA FB 21           Call FB 21
0009 NAME :INDP115A
000A PROD :   MW 10
000B OK   :   M 110.0
000C FEHL :   MW 12
000D      :BE
    
```

### 4.3 Function block FB 10 (communication block)

#### 4.3.1 General

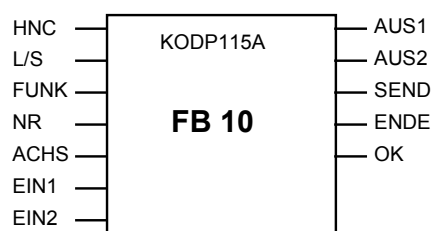
By integrating function block FB 10 in an S5 program the user can read or change values of parameters on the HNC 100 ( R-parameters, machine data, C- and B-variables, process data, digital inputs and outputs as well as flags ). The user can define in the transfer parameters of FB 10, which parameter is to be processed.

To change or read the value of a parameter, FB 10 must be called at least **twice**. In the first call, a user data block is compiled with the help of the transfer parameter and transferred to the HNC 100. In the subsequent call of FB 10 a check is carried out to verify whether the related reply was received. If the reply was not received, the check is repeated in the next calls until the reply has been received or the timeout counter ( output of an error message ! ) has elapsed. After that, the entire process can start over again.

**Caution:**

FB 10 may only be used in cyclical program execution ( OB 1 ). The integration in organization blocks OB 21 and OB 22 is **not** possible.

#### 4.3.2 Parameterization of FB 10



Parameter	Description	Nature	Type
HNC	Address of the HNC	E	BY
L/S	Distinction of direction of data	E	BY
FUNK	Type of the parameter	E	W
NR	Number of the parameter	E	W
ACHS	Number of the axis	E	BY
EIN1	Input value 1	E	W
EIN2	Input value 2	E	W
AUS1	Output value 1	A	W
AUS2	Output value 2	A	W
SEND	Output bit "SEND"	A	BI
ENDE	Output bit "ENDE"	A	BI
OK	Output bit "OK"	A	BI

Transfer parameter of FB 10

HNC

With the HNC transfer parameter, the user determines, which HNC 100 is to process a parameter. It may have any value between 0 and 126. With the help of this parameter, data can be loaded, which are saved for each individual HNC in DB 200 ( e.g.: via which I/O address the HNC 100 can be addressed ).

L/S

The L/S transfer parameter signals whether the value of the relevant parameter is to be read or written on the HNC. If L/S is equal to zero, a write access is executed. However, if L/S is not equal to zero, the corresponding parameter value is read.

L/S	Meaning
0	Parameter value is written
not equal to 0	Parameter value is read

FUNK

The FUNK transfer parameter determines which type of parameter is to be processed by the HNC. With regard to digital inputs and outputs and flags, this parameter is also used to define, whether these are to be set or reset. However, this is only valid, if these are to be written ( L/S = 0 ). However, if L/S is not equal to zero when the digital inputs and outputs or flags are accessed, these are simply read on the HNC and transferred.

FUNK	Meaning
RP	R-parameter
MD	Machine datum
CV	C-variable
BV	B-variable
PG	Process datum
ES	Set digital inputs
ER	Reset digital inputs
AS	Set digital outputs
AR	Reset digital outputs
MS	Set flags
MR	Reset flags

**Caution:**

When changing values of machine data, please note that in the case of some machine data, the new value is only taken over, if a **reset** has been triggered on the HNC 100 ( see manual of the HNC / chapter Machine data ).

**NR**

The NR transfer parameter is used to determine for functions RP, MD, CV, BV, PG, MS and MR, which number of the parameter is to be processed. ( e.g. R-parameter 200: NR = 200, machine datum 13: NR = 13, C-variable 20: NR = 20, B-variable 10: NR = 10, flag 100: NR = 100. ) In conjunction with functions ES, ER, AS and AR, "NR" is used to declare, which I/O card is to be accessed. At present, two I/O cards are available on the HNC100. Each I/O card consists of 32 I/Os. The first I/O card comprises the hardware I/Os. These hardware I/Os are addressed in the NC program via E1.1 - E1.32 and A1.1 - A1.32, respectively. The hardware inputs ( E1.1 - E1.32 ) can only be queried using function block FB 10; the hardware outputs ( A1.1 - A1.32 ) can be queried and set. The second I/O card comprises the so-called virtual I/Os. These I/Os are addressed in the NC program via E2.1 - E2.32 and A1.1 - A1.32, respectively. The virtual inputs ( E2.1 - E2.32 ) and outputs ( A2.1 - A2.32 ) can be set and queried using function block FB 10. For functions ES, ER, AS and AR the NR parameter must contain the following values in order to allow the access to the hardware I/Os or virtual I/Os:

NR	Meaning ( FUNK = ES, ER, AS or AR )
1	Hardware I/Os ( E1.1 - E1.32 and A1.1 - A1.32 )
2	Virtual I/Os ( E2.1 - E2.32 and A2.1 - A2.32 )

In the case of function PG, the NR parameter must contain the number of the process datum whose value is to be read. If, for example, the current value of the actual position value is to be determined, the NR parameter must contain "3" as a value ( L/S = 1, FUNK = PG ). The assignment of the individual process data numbers is listed in the following table.

Number of the process datum	Meaning
3	Actual position value Axis 1
4	Actual position value Axis 2 ( 2-axis version )
5	Actual pressure value Axis 1
6	Actual pressure value Axis 2 ( 2-axis version )
7	Position command value Axis 1
8	Position command value Axis 2 ( 2-axis version )
9	Pressure command value Axis 1
10	Pressure command value Axis 2 ( 2-axis version )
11	Peak value Minus Axis 1
12	Peak value Minus Axis 2 ( 2-axis version )
13	Peak value Plus Axis 1
14	Peak value Plus Axis 2 ( 2-axis version )
15	Actual value Encoder 1
16	Actual value Encoder 2 ( 2-axis version )
17	Actual value Encoder 3 ( 2-axis version )
18	Actual value Encoder 4
19	Actual value Encoder 5
20	Actual value Encoder 6
21	Actual value Encoder 7
22	Actual value Encoder 8
23	Actual value Encoder 9
24	Actual value Encoder 10
25	Actual value Encoder 11
26	Following error Axis 1
27	Following error Axis 2 ( 2-axis version )
28	Synchronization error ( 2-axis version )
29	Target position value Axis 1
30	Target position value Axis 2 ( 2-axis version )



31	Actual velocity value Axis 1
32	Actual velocity value Axis 2 ( 2-axis version )
33	DAC 1
34	DAC 2
35	DAC 3
36	DAC 4
37	Actual value Encoder 4 ( Volt )
38	Actual value Encoder 5 ( Volt )
39	Actual value Encoder 6 ( Volt )
40	Actual value Encoder 7 ( Volt )
41	Actual value Encoder 8 ( Volt )
42	Actual value Encoder 9 ( Volt )
43	Actual value Encoder 10 ( Volt )
44	Actual value Encoder 11 ( Volt )

**ACHS**

The NR transfer parameter is only relevant for the functions RP, MD, CV. It is used to distinguish, on which axis (= NC-program) the relevant parameter (R-parameter, machine datum or C-variable) is to be processed. If, for example, the value of R-parameter R200, which is used in the NC-program of axis 1, is to be read, the ACHS parameter must have the value "1" ( L/S =1, FUNK = RP, NR = 200 ).

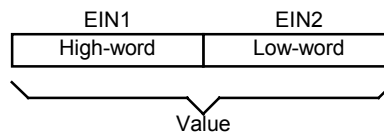
ACHS	Meaning
1	Axis 1 (= NC-program 1)
2	Axis 2 (= NC-program 2)
3	Auxiliary axis (= auxiliary NC-program)

**EIN1 and EIN2**

The transfer parameters EIN1 and EIN2 are used to transfer the value or the bit pattern, which the parameter should have on the HNC. Two transfer parameters are required, as the values of the R-parameters, machine data, C/B-variables and process data and also the bit pattern of the digital inputs and outputs each take 4 bytes. These transfer parameters are only relevant, if L/S is equal to zero ( write access ).

For the functions RP, MD, CV, BV and PG, EIN1 must contain the high-word of the value, EIN2 the low-word.

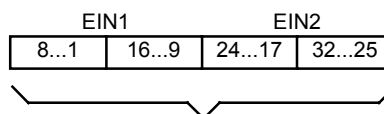
**FUNK = RP, MD, CV, BV, PG**



Example: Value = 100000 ( decimal ) = 186A0 ( hexadecimal )  
 EIN1 = 0001H  
 EIN2 = 86A0H

For the functions ES, ER, AS ad AR, EIN1 must contain the digital inputs or outputs 1 to 16 of the HNC in a bit-coded form, and EIN2 the digital inputs or outputs 17 to 32.

**FUNK = ES, ER, AS, AR**



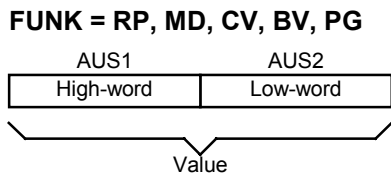
Bit-coded digital inputs or outputs of the HNC

Example: Set input 2, 15, 18 and 32  
 Bit pattern = 00000010 01000000 00000010 10000000  
 Bit pattern ( hexadecimal ) = 02 40 02 80

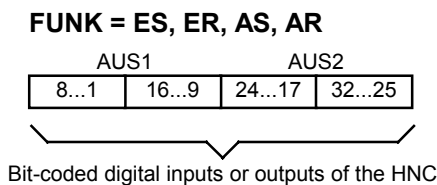
EIN1 = 0240H  
 EIN2 = 0280H

**AUS1 and AUS2**

In the case of a read access ( L/S not equal to zero ) the value or the bit pattern of a parameter is saved in the transfer parameters AUS1 and AUS2. Should an error occur while FB 10 is being executed, AUS1 will contain a corresponding error number.



Example: AUS1 = 0000H  
 AUS2 = 772EH  
 Value = 772E ( hexadecimal ) = 30510 ( decimal )



Example: Inputs on the HNC were read  
 AUS1 = 3210H  
 AUS2 = 5401H  
 Bit pattern ( hexadecimal ) = 32 10 54 01  
 Bit pattern = 00110010 00010000 01010100 00000001  
 The inputs 2, 5, 6, 13, 19, 21, 23 and 25 are set

**Format of transferred values**

All the values of parameters, which are to be transmitted, are transferred in a fixed-point format with three digits after the decimal point. This means that when reading the value of an R-parameter, which has a value of 125.35 on the HNC, the resulting value for the R-parameter in the transfer parameters AUS1 and AUS2 is 125350. The actual value of the parameter can be determined by dividing the value by 1000.

The same is valid for changing the value of a parameter. If, for example, the value of an R-parameter is to be changed to 180, the value which must be saved for the R-parameter in EIN1 and EIN2 is 180000 ( = 180 x 1000 ).

SEND

The SEND output bit signals that FB 10 transferred a user data block to the HNC 100 and that the associated reply is awaited.

SEND	Meaning
0	Reply received
1	Reply awaited

ENDE

The ENDE output bit signals whether a data exchange with the HNC 100 has been completed. This means that a user data block was sent, the associated reply received and data included in AUS1 and AUS2 ( L/S not equal to zero ).

ENDE	Meaning
0	Data exchange is going on
1	Data exchange completed

OK

The OK indication bit provides information on whether an error has occurred during the execution of FB 10 or the data exchange. If OK is equal to 1, no error occurred during the processing of FB 10 or the data exchange. However, in case an error occurred, OK is equal to zero and AUS1 contains a corresponding error number. The user must react to an error as required.

Error number	Meaning
F000	Specified HNC is invalid
F001	HNC does not respond (timeout elapsed)
F002	Error on HNC ( e.g. encoder error )
F003	Invalid axis number specified
F004	Invalid I/O card specified
F005	Invalid function specified
F006	HNC not coupled to Profibus ( L2 = 0 )
F007	Invalid C-variable number
F008	Invalid B-variable number or process datum number specified
F009	Incorrect area specified in DB200 ( P- or Q-area )
FFCC	Process datum is not defined
FFB1	Function is not defined ( transfer protocol )
FFD2	Invalid flag number
FFD3	Writing of process datum is impossible
FFD4	Writing of digital inputs is impossible
FFD5	Invalid B-variable number
FFD6	Invalid axis number specified
FFD7	Invalid C-variable number
FFD8	Value of the machine datum too small
FFD9	Value of the machine datum too great
FFDA	Value of the R-parameter too small
FFDB	Value of the R-parameter too great
FFFC	Machine datum not defined
FFFD	R-parameter not defined

Overview of possible errors with FB 10

**Caution:**

**No data word** must be used for the transfer parameters AUS1 and AUS2, because data block DB 200 is called in FB 10.

**Summary**

The following table shows the possible combinations of the individual transfer parameters in a summarized form. The parameters marked with an "X" in this table are relevant for the execution of the individual functions ( FUNK ).

FUNK	L/S	NR	ACHS	EIN1/EIN2 <sup>1</sup>	AUS1/AUS2 <sup>2</sup>
RP	X	X	X	X	X
MD	X	X	X	X	X
CV	X	X	X	X	X
BV	X	X		X	X
PG	X	X		X	X
ES	X	X		X	X
ER	X	X		X	X
AS	X	X		X	X
AR	X	X		X	X
MS	X	X			X
MR	X	X			X

<sup>1</sup>) The value in EIN1/EIN2 is only considered, if L/S is equal to zero

<sup>2</sup>) A value is only output via AUS1/AUS2 if L/S is not equal to zero

Possible combinations of transfer parameters

**4.3.3 Technical data**

Block number	FB 10
Block name	KODP115A
Nesting depth	1
Assigned data blocks	DB 200
Assigned flags	MB 200 -253
Assigned timers <sup>1)</sup>	T ( HNC addresses )
Average operating time ( S5-115U )	CPU 943: 3.3 ms CPU 944: 0.2 ms

<sup>1)</sup> For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S5-115U. Thus, timers T5 and T12 are assigned.

**4.3.4 Application examples****Fetching the value of an R-parameter from the HNC**

The value of R-parameter R201, which is assigned to axis 1, is to be fetched from the HNC 100 ( address = 11 ), when a rising edge has been detected on input E0.0 and FB 10 does not send ( SEND = 0 ).

The value of the R-parameter is included in flag words MW20 and MW22, when FB 10 has been completed without error ( ENDE = 1, OK = 1 ). The related S5-program is shown in the following.

```

001A      :
001B      :U   E   0.0
001C      :UN  M  60.0
001D      :UN  A   4.0
001E      :S   M  60.1
001F      :
0020      :U   E   0.0
0021      :S   M  60.0
0022      :UN  E   0.0
0023      :R   M  60.0
0024      :
0025      :A   DB  10
0026      :L   KF +11
0028      :T   DL   0
0029      :
002A      :L   KH 0001
002C      :T   DR   0
002D      :
002E      :L   KC RP
0030      :T   DW   1
0031      :
0032      :L   KF +201
0034      :T   DW   2
0035      :
0036      :L   KH 0001
0038      :T   DL   3
0039      :
003A      :L   KH 0000
003C      :T   DW   4
003D      :T   DW   5
003E      :
003F      :U   M  60.1
0040      :SPB FB  10
0041 NAME :KODP115A
0042 HNC   :    DL   0
0043 L/S   :    DR   0
0044 FUNK  :    DW   1
0045 NR    :    DW   2
0046 ACHS  :    DL   3
0047 EIN1  :    DW   4
0048 EIN2  :    DW   5
0049 AUS1  :    MW  20
004A AUS2  :    MW  22
004B SEND  :    A    4.0
004C ENDE  :    A    4.1
004D OK    :    A    4.2
004E      :

```

Flag M60.1 is set, when a rising edge is detected on input E0.0 and the FB 10 does not send (SEND = 0)

Save status of input E0.0 in flag M60.0

Load HNC address

Load (read) direction of data

Load function (R-parameter)

Load number of the R-parameter

Load number of the axis

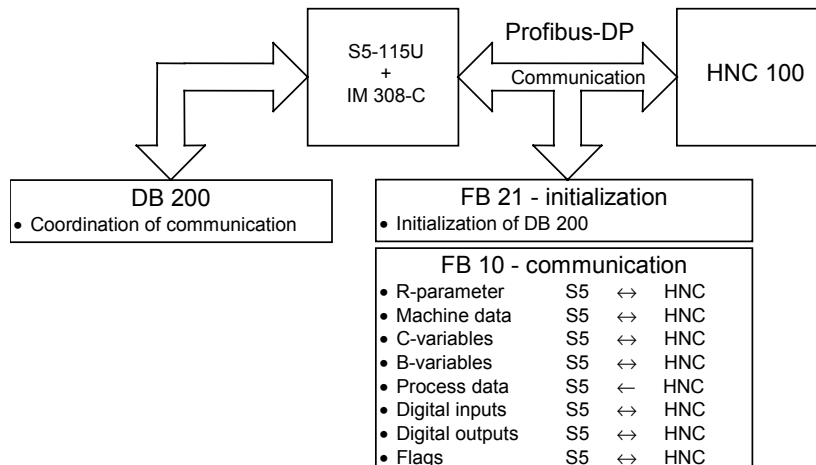
Initialize transfer parameters EIN1 and EIN2 with zero

### Further examples

For further application examples, please refer to the description of function FC 11 in chapter 8.5.4.

## 5. Function and data blocks (S5-115U - CPU: 945)

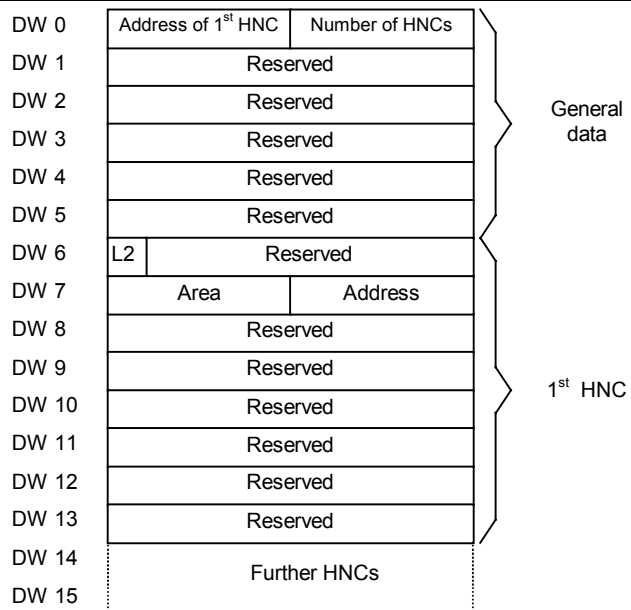
To facilitate the user's access to the individual HNCs via the Profibus-DP with the IM 308-C, Bosch Rexroth created function blocks ( FB 10 and FB 21 ). The user simply must integrate these blocks into his S5 program and call them with suitable transfer parameters.



### 5.1 Data block DB 200

The user must save in data block DB 200 in defined data words the addresses on the IM 308-C, via which the user data can be transferred to the individual HNC 100s ( determination of the addresses with the help of the COM PROFIBUS program ). In addition, you must save in DB 200, via which memory area ( P- or Q-area ) the data are to be transferred. For the P-area, 'P' ( = 50 hex ) is to be saved at the relevant place in DB 200, and for the Q-area 'Q' ( = 51 hex ). In addition, certain data and states can be saved in DB 200 in order to coordinate the communication via the Profibus-DP.

The length of data block DB 200 depends on the number of HNCs, which are connected to the IM 308-C via the Profibus-DP. 8 data words are required for each HNC 100. The first 6 data words in the DB 200 are required for saving general data. The first data word is used for saving the address of the 1<sup>st</sup> HNC and the total number of HNCs. For an HNC 100 that is coupled to the Profibus-DP, **bit L2** must be set additionally **equal to one**.

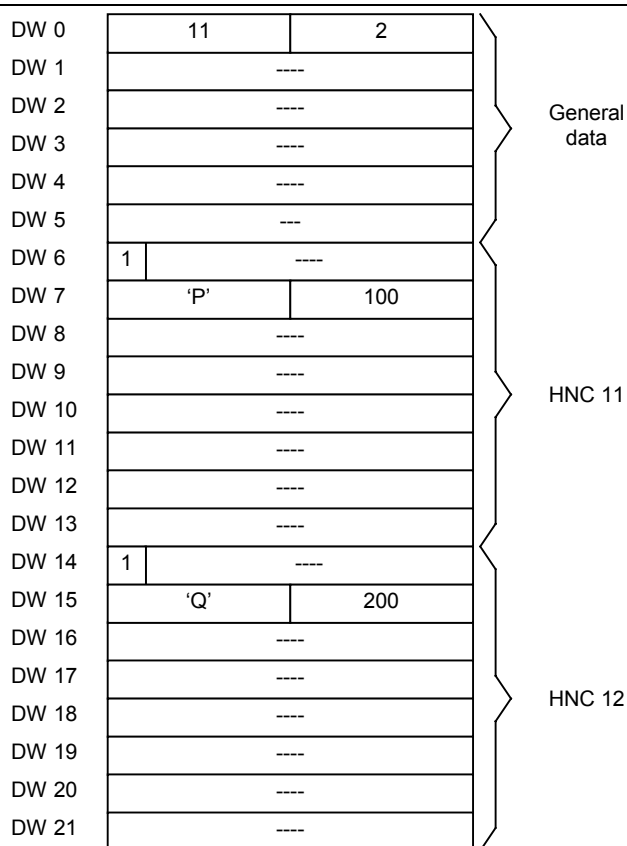


Structure of data block DB 200

Example

The HNCs with **address 11** and **address 12** are connected to the Profibus-DP. This results in a length of 22 data words for data block DB 200 ( 6 data words for general data, 16 data words for the two HNCs ). Consequently, DB200 must include data word DW 0 through data word DW 21. In the first data byte of the DB 200, **11** must be entered, since the address of the first HNC is "11". In the second data byte, **2** must be entered, since two HNCs are available. The L2-bit is to be set for each HNC.

With the COM PROFIBUS software, it was determined that data are exchanged with the individual HNC 100s via **addresses 100 to 107 ( P-area )** and **200 to 207 ( Q-area )**. This address range is used for both transferring data to the HNC 100 and reading data that are received from the HNC 100. The user must enter the addresses, which were determined for the individual HNC 100s using the COM PROFIBUS software, as illustrated in the example below .



Example of DB 200 assignment by the user

## 5.2 Function block FB 21 (initialization)

### 5.2.1 General

The application and parameterization of function block FB 21 are identical with those of *function and data blocks for S5-115U ( CPU: 941 - 944 )*. A more detailed description of FB 21 is included in chapters 4.2.1, 4.2.2 and 4.2.4.

### 5.2.2 Technical data

Block number	FB 21
Block name	INDP945
Nesting depth	1
Assigned data blocks	DB 200
Assigned flags	MB 200 - 205
Assigned timers <sup>1)</sup>	T ( HNC addresses )
Average operating time (S5-115U, length of DB 200: 22 DWs)	CPU 945: 30 µs

1) For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S5-115U. Thus, timers T5 and T12 are assigned.



## 5.3 Function block FB 10 (communication block)

### 5.3.1 General

The application and parameterization of function block FB 10 are identical with those of *function and data blocks for S5-115U ( CPU: 941 - 944 )*. A more detailed description of FB 10 is included in chapters 4.3.1, 4.3.2 and 4.3.4.

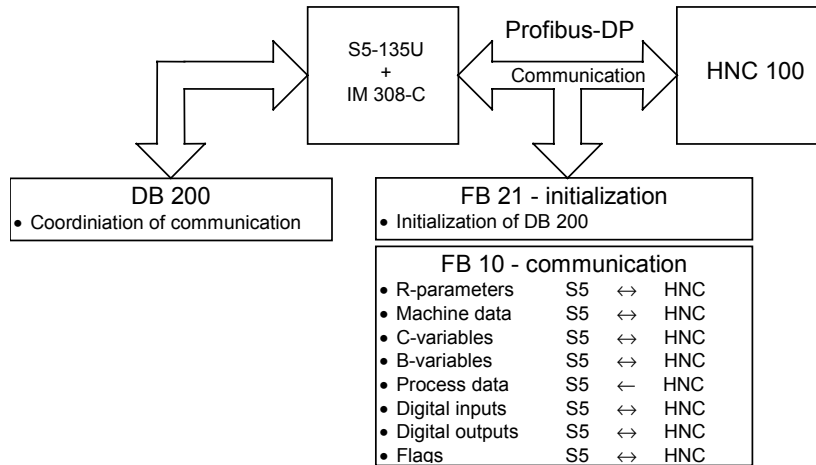
### 5.3.2 Technical data

Block number	FB 10
Block name	KODP945
Nesting depth	1
Assigned data blocks	DB 200
Assigned flags	MB 200 -253
Assigned timers <sup>1)</sup>	T ( HNC addresses )
Average operating time ( S5-115U )	CPU 945: 40 µs

1) For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S5-115U. Thus, timers T5 and T12 are assigned.

## 6. Function and data blocks (S5-135U- CPU: 928 and 948)

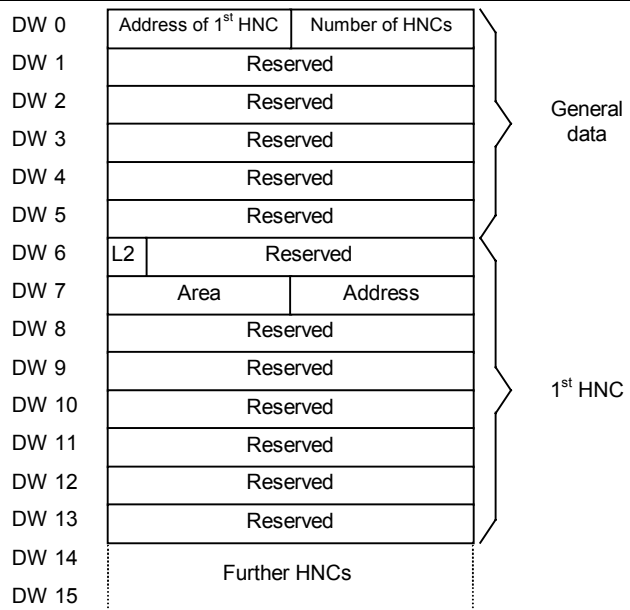
To facilitate the user's access to the individual HNCs via the Profibus-DP with the IM 308-C, Bosch Rexroth created function blocks ( FB 10 and FB 21 ). The user simply must integrate these blocks into his S5 program and call them with suitable transfer parameters.



### 6.1 Data block DB 200

The user must save in data block DB 200 in defined data words the addresses on the IM 308-C, via which the user data can be transferred to the individual HNC 100s ( determination of the addresses with the help of the COM PROFIBUS program ). In addition, you must save in DB 200, via which memory area ( P- or Q-area ) the data are to be transferred. For the P-area, 'P' ( = 50 hex ) is to be saved at the relevant place in DB 200, and for the Q-area 'Q' ( = 51 hex ). In addition, certain data and states can be saved in DB 200 in order to coordinate the communication via the Profibus-DP.

The length of data block DB 200 depends on the number of HNCs, which are connected to the IM 308-C via the Profibus-DP. 8 data words are required for each HNC 100. The first 6 data words in the DB 200 are required for saving general data. The first data word is used for saving the address of the 1<sup>st</sup> HNC and the total number of HNCs. For an HNC 100 that is coupled to the Profibus-DP, **bit L2** must be set additionally **equal to one**.

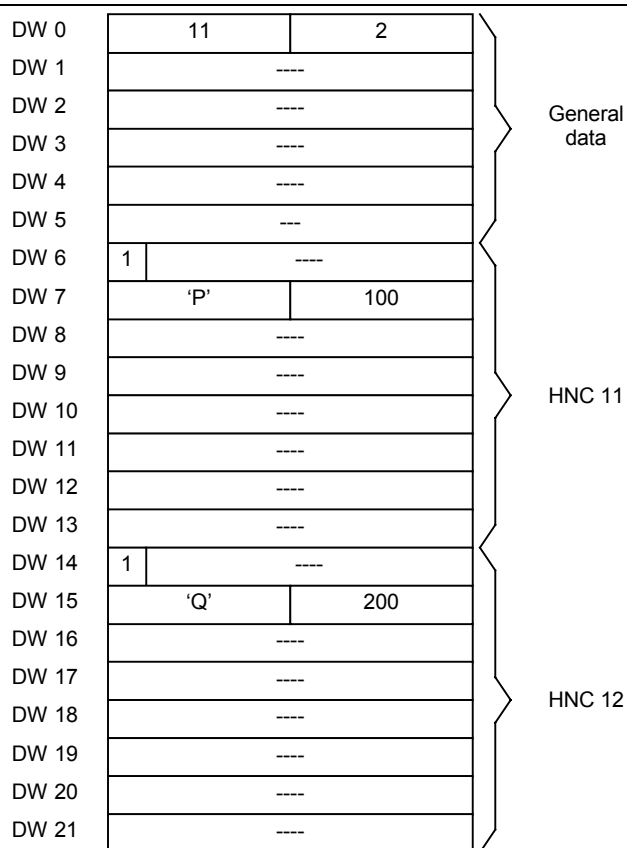


Structure of data block DB 200

Example

The HNCs with **address 11** and **address 12** are connected to the Profibus-DP. This results in a length of 22 data words for data block DB 200 ( 6 data words for general data, 16 data words for the two HNCs ). Consequently, DB 200 must include data word DW 0 through data word DW 21. In the first data byte of the DB 200, **11** must be entered, since the address of the first HNC is "11". In the second data byte, **2** must be entered, since two HNCs are available. The L2-bit is to be set for each HNC.

With the COM PROFIBUS software, it was determined that data are exchanged with the individual HNC 100s via **addresses 100 to 107 ( P-area )** and **200 to 207 ( Q-area )**. This address range is used for both transferring data to the HNC 100 and reading data that are received from the HNC 100. The user must enter the addresses, which were determined for the individual HNC 100s using the COM PROFIBUS software, as illustrated in the example below .



Example of DB 200 assignment by the user

## 6.2 Function block FB 21 (initialization)

### 6.2.1 General

The application and parameterization of function block FB 21 are identical with those of *function and data blocks for S5-115U ( CPU: 941 - 944 )*. A more detailed description of FB 21 is included in chapters 4.2.1, 4.2.2 and 4.2.4.

### 6.2.2 Technical data

Block number	FB 21
Block name	INDP135U
Nesting depth	1
Assigned data blocks	DB 200
Assigned flags	MB 200 - 205
Assigned timers <sup>1)</sup>	T ( HNC addresses )
Average operating time (S5-135U, length of DB200: 22 DWs)	CPU 928A: 2 ms

1) For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S5-135U. Thus, timers T5 and T12 are assigned.

## 6.3 Function block FB 10 (communication block)

### 6.3.1 General

The application and parameterization of function block FB 10 are identical with those of *function and data blocks for S5-115U ( CPU: 941 - 944 )*. A more detailed description of FB 10 is included in chapters 4.3.1, 4.3.2 and 4.3.4.

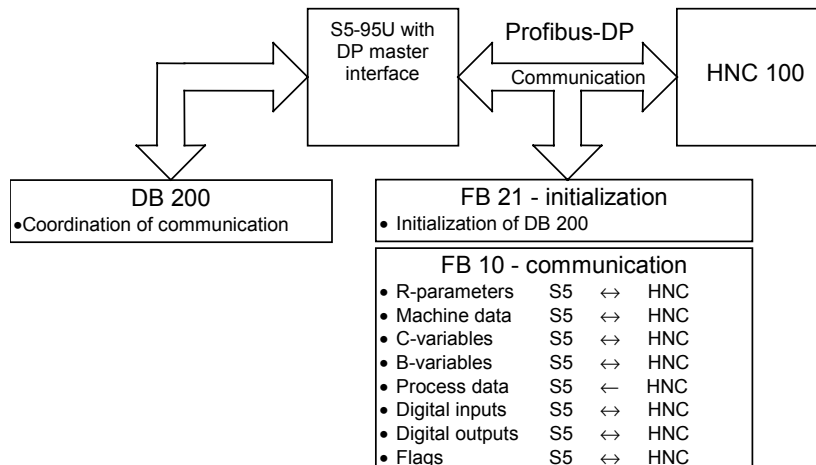
### 6.3.2 Technical data

Block number	FB 10
Block name	KODP135U
Nesting depth	1
Assigned data blocks	DB 200
Assigned flags	MB 200 -253
Assigned timers <sup>1)</sup>	T ( HNC addresses )
Average operating time ( S5-135U )	CPU 928A: 2.3 ms

1) For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S5-135U. Thus, timers T5 and T12 are assigned.

## 7. Function and data blocks (for S5-95U with DP master interface)

To facilitate the user's access to the individual HNCs via the Profibus-DP with the S5-95U, Bosch Rexroth created function blocks ( FB 10 and FB 21 ). The user simply must integrate these blocks into his S5 program and call them with suitable transfer parameters. An overview of the individual function and data blocks is given in the following diagram.



### 7.1 Data block DB 200

The user must save in data block DB 200 the I/O addresses, via which the user data can be transferred to the individual HNC 100s ( determination of the addresses with the help of the COM PROFIBUS program ). In addition, certain data and states can be saved in DB 200 in order to coordinate the communication via the Profibus-DP.

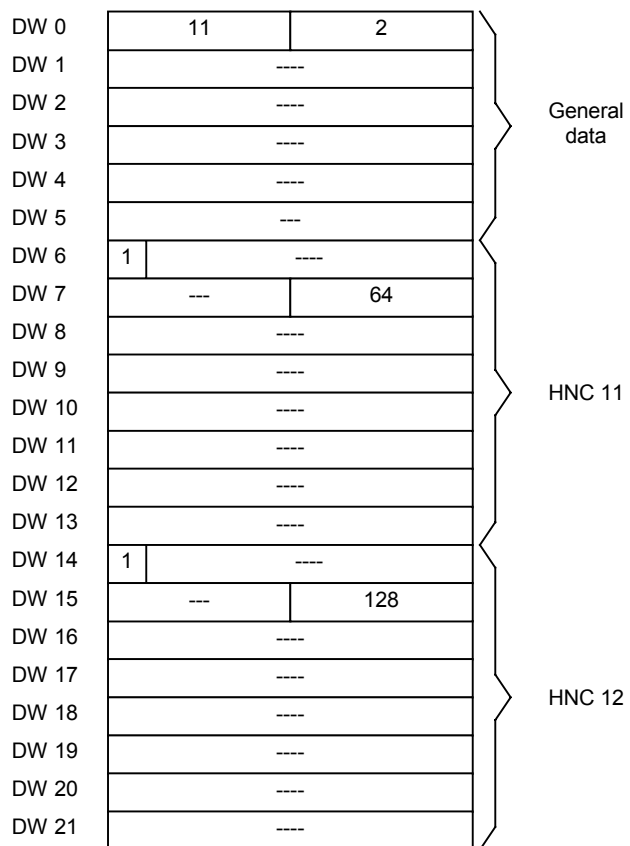
The length of data block DB 200 depends on the number of HNCs, which are connected to the S5-95U via the Profibus-DP. 8 data words are required for each HNC 100. The first 6 data words in the DB 200 are required for saving general data. The first data word is used for saving the address of the 1<sup>st</sup> HNC and the total number of HNCs. For an HNC 100 that is coupled to the Profibus-DP, **bit L2** must be set additionally **equal to one**.

DW 0	Address of 1 <sup>st</sup> HNC	Number of HNCs	} General data
DW 1	Reserved		
DW 2	Reserved		
DW 3	Reserved		
DW 4	Reserved		
DW 5	Reserved		
DW 6	L2	Reserved	} 1 <sup>st</sup> HNC
DW 7	Reserved	Address	
DW 8	Reserved		
DW 9	Reserved		
DW 10	Reserved		
DW 11	Reserved		
DW 12	Reserved		} Further HNCs
DW 13	Reserved		
DW 14	Reserved		
DW 15	Reserved		

Example

The HNCs with **address 11** and **address 12** are connected to the Profibus-DP. This results in a length of 22 data words for data block DB200 ( 6 data words for general data, 16 data words for the two HNCs ). Consequently, DB 200 must include data word DW 0 through data word DW 21. In the first data byte of the DB 200, **11** must be entered, since the address of the first HNC is "11". In the second data byte, **2** must be entered, since two HNCs are available. The L2-bit is to be set for each HNC.

With the COM PROFIBUS software, it was determined that data are exchanged with the individual HNC 100s via **I/O addresses 64 to 71** and **128 to 135**. This address range is used for both transferring data to the HNC 100 and reading data that are received from the HNC 100. The user must enter the addresses, which were determined for the individual HNC 100s using the COM PROFIBUS software, as illustrated in the example below .



Example of DB 200 assignment by the user

## 7.2 Function block FB 21 (initialization)

### 7.2.1 General

The application and parameterization of function block FB 21 are identical with those of *function and data blocks for S5-115U ( CPU: 941 - 944 )*. A more detailed description of FB 21 is included in chapters 4.2.1, 4.2.2 and 4.2.4.

## 7.2.2 Technical data

Block number	FB 21
Block name	INDP095U
Nesting depth	1
Assigned data blocks	DB 200
Assigned flags	MB 200 - 205
Assigned timers <sup>1)</sup>	T ( HNC addresses )
Average operating time (S5-95U, length of DB200: 22 DWs)	2 ms

1) For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S5-95U. Thus, timers T5 and T12 are assigned.

## 7.3 Function block FB 10 (communication block)

### 7.3.1 General

The application and parameterization of function block FB 10 are identical with those of *function and data blocks for S5-115U ( CPU: 941 - 944 )*. A more detailed description of FB 10 is included in chapters 4.3.1, 4.3.2 and 4.3.4.

### 7.3.2 Technical data

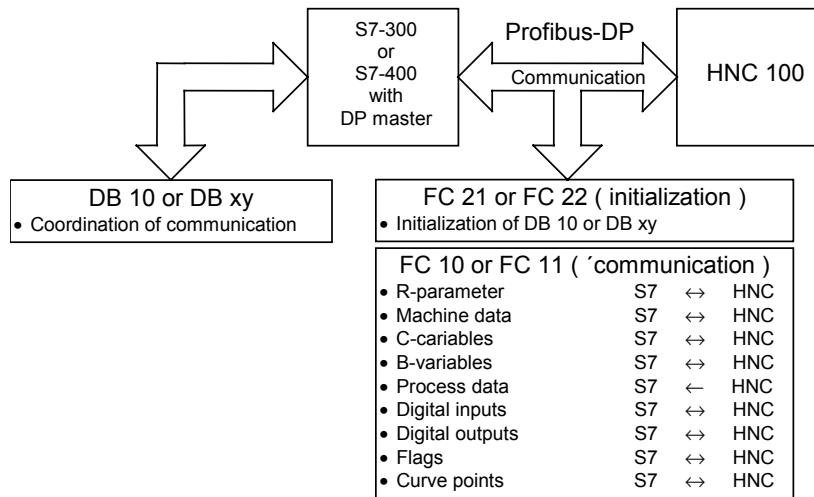
Block number	FB 10
Block name	KODP095U
Nesting depth	1
Assigned data blocks	DB 200
Assigned flags	MB 200 -253
Assigned timers <sup>1)</sup>	T ( HNC addresses )
Average operating time (S5-95U)	3.1 ms

1) For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S5-95U. Thus, timers T5 and T12 are assigned.



## 8. Functions and data blocks (S7-300 and S7-400 with DP master)

To facilitate the user's access to the individual HNCs via the Profibus-DP with the S7-300 and S7-400 with integrated DP master, Bosch Rexroth wrote functions ( FC 10 and FC 21 as well as FC 11 and FC 22 ). The user simply must integrate these functions into his S7 program and call them with suitable transfer parameters.



Functions FC 10 and FC 11, respectively, can be used to read or change parameters on the HNC 100 ( R-parameters, machine data, C- and B-variables, process data, digital inputs and outputs as well as flags ). The transfer parameters of functions FC 10 and FC 11 define, which parameter is to be read or changed.

Both functions, FC 10 and FC 11, require a data block, which contains in defined data words the I/O addresses, via which the data are to be exchanged with the individual HNCs. ( Remark: The I/O addresses are determined using the STEP7 program in the program part "Hardware configuration". ). Function FC 10 expects that the I/O addresses are included in data block DB 10. With function FC 11, the I/O addresses can be included in any data block. The number of the data block is signaled to function FC 11 in transfer parameter DBNR.

The data block that contains the I/O addresses is initialized using function FC 21 or FC 22. Function FC 21 can only initialize data block FC 10. Function FC 22 can initialize any data block. The number of the data block is signaled to function FC 22 in the transfer parameter DBNR.

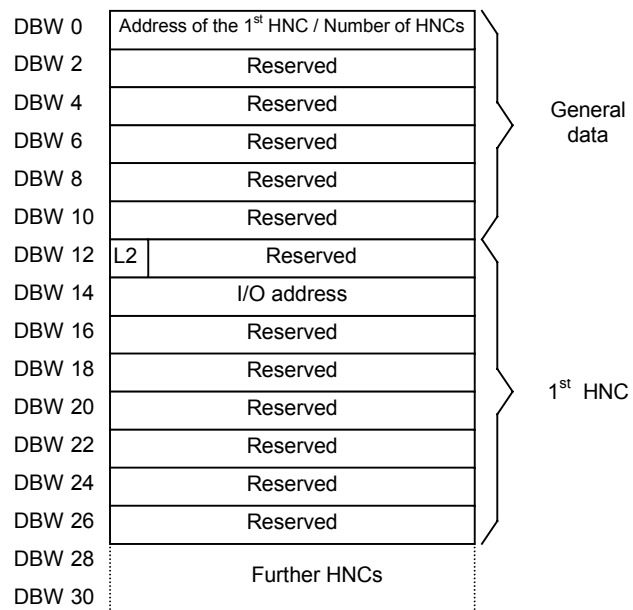
### 8.1 Data block DB 10 or DB xy

Function FC 10 and function FC 11 require the I/O addresses, via which user data can be transferred to the individual HNC 100s ( remark: The I/O addresses are determined using the STEP7 program in the program part "Hardware configuration". ). These I/O addresses must be saved in a data block with firmly defined data words.

If function FC 10 is to be used for reading or changing parameters on the HNC 100, the I/O addresses must be included in data block DB 10. However, with function FC 11, the I/O addresses can be included in any DB xy data block. The number of the data block will then be saved in transfer parameter DBNR.

Apart from the I/O addresses, certain data and states are saved in data block DB 10 or DB xy, respectively, to coordinate the communication via the Profibus-DP. The length of data block DB 10 or DB xy, respectively, depends on the number of HNCs, which are connected to the S7-300 or

S7-400 via the Profibus-DP. 8 data words are required for each HNC 100. The first 6 data words are required for saving general data. The first data word is used for saving the address of the 1<sup>st</sup> HNC and the total number of HNCs. For an HNC 100 that is coupled to the Profibus-DP, **bit L2** must be set additionally **equal to one**.

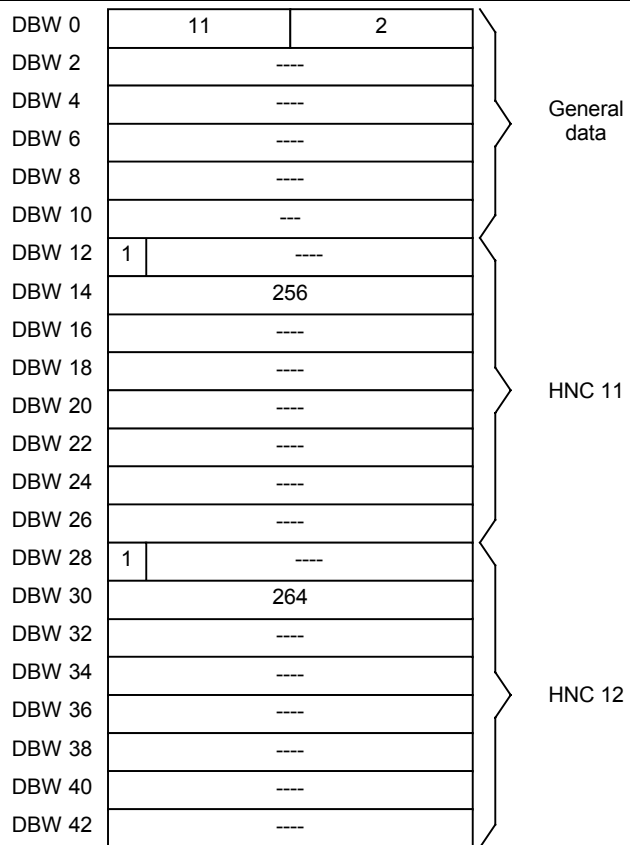


Structure of data block DB 10 or DB xy

Example

The HNCs with the **addresses 11 and 12** are connected to the Profibus-DP. This results in a length of 22 data words for data block DB 10 or DB xy, respectively ( 6 data words for general data, 16 data words for the two HNCs ). Consequently, DB 10 or DB xy must include data word DBW 0 through data word DBW 42. In the first data byte, **11** must be entered, since the address of the first HNC is “11”. In the second data byte, **2** must be entered, since two HNCs are available. The L2-bit is to be set for each HNC.

With the help of software STEP7 it was determined in the hardware configuration that data can be exchanged with the HNC 100 via **I/O addresses ( bytes ) 256 to 263** or **264 to 271**. This address range is used for both transferring data to the HNC 100 and reading data that are received from the HNC 100. The user must enter the I/O addresses, which were determined for the individual HNC 100s using the STEP7 software, as illustrated in the example below.



Example of DB 10 or DB xy assignment by the user

## 8.2 Function FC 21 (initialization)

### 8.2.1 General

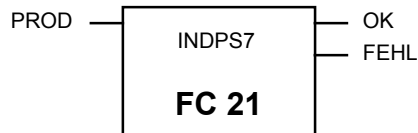
Function FC 21 should be called **once** in organization block OB 100 ( restart ). The use of function FC 21 should be avoided in cyclical program processing ( OB 1 ).

The main task of FC 21 is to initialize data block DB 10. Data block DB 10 contains in defined data words the I/O addresses, via which data are exchanged with the individual HNCs ( cf. chapter 8.1 ).

#### Remark

The task of function FC 22 is the same as that of function FC 21. Function FC 22 can not only initialize data block number 10, but any data block ( cf. chapter 8.3 ). The data block should be initialized using either function FC 21 or function FC 22 .

### 8.2.2 Parameterization of FC 21



Parameter	Description	Nature	Type
PROD	Product data management	in	WORD
OK	Indication bit "OK"	out	BOOL
FEHL	Error output	out	WORD

Transfer parameters of FC 21

**PROD (product data management)** At present, the transfer parameter PROD is irrelevant. It can have any value.

**OK ( error-free execution )** This bit provides information on whether an error has occurred during the processing of function FC 21. If the value of this parameter is 1, FC 21 was processed without any error. However, if the content of this bit is zero, an error has occurred during processing of this function. In this case, the associated error number is saved in the transfer parameter "FEHL".

**FEHL ( error )** This transfer parameter includes an error number, if the value of the OK bit is zero. The figure below shows an overview of possible errors with the associated numbers.

Error number	Meaning
F00A	The length of DB 10 is incorrect.
F00B	The values of data bytes DBB 0 and 1 are not permitted in DB 10.

Overview of possible errors with FC 21

**Caution:**

If a data word is used for transfer parameter FEHL, it must be transferred with the absolute address of FC 21 ( e.g. DB20.DBW10 ). This is required, because data block DB 10 is called in FC 21.

### 8.2.3 Technical data

Function number	FC 21
Function name	INDPS7
Nesting depth	1
Assigned data blocks	DB 10
Local data	8 bytes
Assigned timers <sup>1)</sup>	T ( HNC addresses )
Average running time (S7-300, length of DB 10: 22 DBWs)	CPU 315-2 DP: 0.3 ms

1) For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S7-300. Thus, timers T5 and T12 are assigned.

### 8.2.4 Application example

```
// Data block DB 10 is initialized by calling function FC 21.
L      W#16#0
T      MW      2
T      MW      4
CALL  FC      21
      PROD:=MW2
      OK   :=A0.0
      FEHL:=MW4
```

## 8.3 Function FC 22 (initialization)

### 8.3.1 General

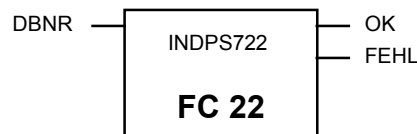
Function FC 22 should be called **once** in organization block OB 100 ( restart ). The use of function FC 22 should be avoided in cyclical program processing ( OB 1 ).

The main task of FC 22 is to correctly initialize all data words in the data block, whose number is saved in transfer parameter DBNR. This transferred data block contains in defined data words the I/O addresses, via which data are exchanged with the individual HNCs ( cf. chapter 8.1 ).

#### Remark

The task of function FC 21 is the same as that of function FC 22. Function FC 21 can **only** initialize data block number 10 ( cf. chapter 8.2 ).

### 8.3.2 Parameterization of FC 22



Parameter	Description	Nature	Type
DBNR	Number of the data block	in	INT
OK	Indication bit "OK"	out	BOOL
FEHL	Error output	out	WORD

#### Transfer parameters of FC 22

**DBNR (number of the data block)** This transfer parameter contains the number of the data block that is to be initialized. This data block contains in defined data words the I/O addresses via which data are exchanged with the individual HNCs ( cf. chapter 8.1 ).

**OK ( error-free execution )** This bit provides information of whether an error has occurred during the processing of function FC 22. If the value of this parameter is 1, FC 22 was processed without any error. However, if the content of this bit is zero, an error has occurred during processing of this function. In this case, the associated error number is saved in the transfer parameter "FEHL".

FEHL ( error )

This transfer parameter includes an error number, if the value of the OK bit is zero. The figure below shows an overview of possible errors with the associated numbers.

Error number	Meaning
F00A	The length of DB 10 is incorrect.
F00B	The values of data bytes DBB 0 and 1 are not permitted in DB 10.

Overview of possible errors with FC 22

**Caution:**

If a data word is used for transfer parameter FEHL, it must be transferred with the absolute address of FC 22 ( e.g. DB20.DBW10 ). This is required, because the data block, whose number is included in transfer parameter DBNR, is opened in FC 22.

### 8.3.3 Technical data

Function number	FC 22
Function name	INDPS722
Nesting depth	1
Assigned data blocks	Data block, whose number is written in transfer parameter DBNR
Local data	8 bytes
Assigned timers <sup>1)</sup>	T (HNC addresses )
Average running time (S7-300, length of DB 10: 22 DBWs)	CPU 315-2 DP: 0.3 ms

1) For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S7-300. Thus, timers T5 and T12 are assigned.

### 8.3.4 Application example

```
// Transfer parameter DBNR of function FC 22 contains the value "10". This results
// in the fact that data block DB 10 will be initialized.
L      W#16#0
T      MW      4
CALL  FC      22
      DBNR:=10
      OK   :=A0.0
      FEHL:=MW4
```

## 8.4 Function FC 10 (communication function)

### 8.4.1 General

By integrating function FC 10 in an S7 program the user can read or change values of parameters on the HNC 100 ( R-parameters, machine data, C- and B-variables, process data, digital inputs and outputs as well as flags ). The user can define in the transfer parameters of FC 10, which parameter is to be processed.

To change or read the value of a parameter, FC 10 must be called at least **twice**. In the first call, a user data block is compiled with the help of the transfer parameter and transferred to the HNC 100. In the subsequent call of FC 10 a check is carried out to verify whether the related reply was

received. If the reply was not received, the check is repeated in the next calls until the reply has been received or the timeout counter ( output of an error message ! ) has elapsed. After that, the entire process can start over again.

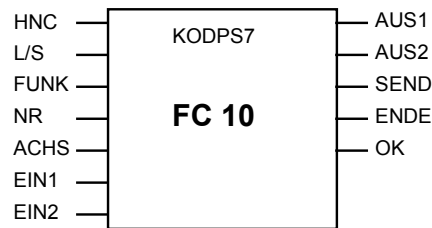
**Remark**

The task of function FC 11 is the same as that of function FC 10. Function FC 10 assumes that the I/O addresses, via which data are exchanged with the individual HNCs, are contained in data block DB 10. With function FC 11, the I/O addresses can be included in any data block. The number of the data block is signaled to function FC 11 in the transfer parameter DBNR ( cf. chapter 8.5 ). Reading or changing parameters on the HNC 100 should be carried out using **only** function FC 10 or **only** function FC 11.

**Caution:**

FC 10 may only be used in cyclical program processing ( OB 1 ). It **cannot** be integrated in the starting organization block ( e.g. OB 100 ).

**8.4.2 Parameterization of FC 10**



Parameter	Description	Nature	Type
HNC	Address of the HNC	In	BYTE
L/S	Distinction of direction of data	In	BYTE
FUNK	Type of the parameter	In	WORD
NR	Number of the parameter	In	WORD
ACHS	Number of the axis	In	BYTE
EIN1	Input value 1	In	WORD
EIN2	Input value 2	In	WORD
AUS1	Output value 1	out	WORD
AUS2	Output value 2	out	WORD
SEND	Output bit "SEND"	out	BOOL
ENDE	Output bit "ENDE"	out	BOOL
OK	Output bit "OK"	out	BOOL

Transfer parameters of FC 10

**HNC**

With the HNC transfer parameter, the user determines, which HNC 100 is to process a parameter. It may have values ranging from 0 to 126. With the help of this parameter, data can be loaded, which are saved for each individual HNC in DB 10 ( e.g.: via which I/O address the HNC 100 can be addressed. ).

**L/S**

The L/S transfer parameter signals whether the value of the relevant parameter is to be read or written on the HNC. If L/S is equal to zero, a write access is executed. However, if L/S is not equal to zero, the corresponding parameter value is read.

L/S	Meaning
0	Parameter value is written

not equal to 0	Parameter value is read
----------------	-------------------------

**FUNK**

The FUNK transfer parameter determines which type of parameter is to be processed by the HNC. With regard to digital inputs and outputs and flags, this parameter is also used to define, whether these are to be set or reset. However, this is only valid, if these are to be written ( L/S = 0 ). However, if L/S is not equal to zero when the digital inputs and outputs or flags are accessed, these are simply read on the HNC and transferred.

<b>FUNK</b>	<b>Meaning</b>
RP	R-parameter
MD	Machine datum
CV	C-variable
BV	B-variable
PG	Process datum
ES	Set digital inputs
ER	Reset digital inputs
AS	Set digital outputs
AR	Reset digital outputs
MS	Set flags
MR	Reset flags

**Caution:**

When changing values of machine data, please note that in the case of some machine data, the new value is only taken over, when a **reset** has been triggered on the HNC 100.

**NR**

The NR transfer parameter is used to determine for functions RP, MD, CV, BV, PG, MS and MR, which number of the parameter is to be processed. ( e.g. R-parameter 200: NR = 200, machine datum 13: NR = 13, C-variable 20: NR = 20, B-variable 10: NR = 10, flag 100: NR = 100 ). In conjunction with functions ES, ER, AS and AR, "NR" is used to declare, which I/O card is to be accessed. At present, two I/O cards are available on the HNC100. Each I/O card consists of 32 I/Os. The first I/O card comprises the hardware I/Os. These hardware I/Os are addressed in the NC program via E1.1 - E1.32 and A1.1 - A1.32, respectively. The hardware inputs ( E1.1 - E1.32 ) can only be queried using function FC 10; the hardware outputs ( A1.1 - A1.32 ) can be queried and set. The second I/O card comprises the so-called virtual I/Os. These I/Os are addressed in the NC program via E2.1 – E2.32 and A1.1 - A1.32, respectively. The virtual inputs ( E2.1 - E2.32 ) and outputs ( A2.1 - A2.32 ) can be set and queried using function FC 10. For functions ES, ER, AS and AR, the NR parameter must contain the following values in order to access the hardware I/Os or virtual I/Os:



NR	Meaning ( FUNK = ES, ER, AS or AR )
1	Hardware I/Os ( E1.1 - E1.32 and A1.1 - A1.32 )
2	Virtual I/Os ( E2.1 - E2.32 and A2.1 - A2.32 )

In the case of function PG, the NR parameter must contain the number of the process datum whose value is to be read. If, for example, the current value of the actual position value is to be determined, the NR parameter must contain "3" as a value ( L/S = 1, FUNK = PG ). The assignment of the individual process data numbers is listed in the following table.

Number of the process datum	Meaning
3	Actual position value Axis 1
4	Actual position value Axis 2 ( 2-axis version )
5	Actual pressure value Axis 1
6	Actual pressure value Axis 2 ( 2-axis version )
7	Position command value Axis 1
8	Position command value Axis 2 ( 2-axis version )
9	Pressure command value Axis 1
10	Pressure command value Axis 2 ( 2-axis version )
11	Peak value Minus Axis 1
12	Peak value Minus Axis 2 ( 2-axis version )
13	Peak value Plus Axis 1
14	Peak value Plus Axis 2 ( 2-axis version )
15	Actual value Encoder 1
16	Actual value Encoder 2 ( 2-axis version )
17	Actual value Encoder 3 ( 2-axis version )
18	Actual value Encoder 4
19	Actual value Encoder 5
20	Actual value Encoder 6
21	Actual value Encoder 7
22	Actual value Encoder 8
23	Actual value Encoder 9
24	Actual value Encoder 10
25	Actual value Encoder 11
26	Following error Axis 1
27	Following error Axis 2 ( 2-axis version )
28	Synchronization error ( 2-axis version )
29	Target position value Axis 1
30	Target position value Axis 2 ( 2-axis version )
31	Actual velocity value Axis 1
32	Actual velocity value Axis 2 ( 2-axis version )
33	DAC 1
34	DAC 2
35	DAC 3
36	DAC 4
37	Actual value Encoder 4 ( Volt )
38	Actual value Encoder 5 ( Volt )
39	Actual value Encoder 6 ( Volt )
40	Actual value Encoder 7 ( Volt )
41	Actual value Encoder 8 ( Volt )
42	Actual value Encoder 9 ( Volt )
43	Actual value Encoder 10 ( Volt )
44	Actual value Encoder 11 ( Volt )

**ACHS**

The ACHS transfer parameter is only relevant for the functions RP, MD, CV. It is used to distinguish, on which axis ( = NC-program ) the relevant parameter ( R-parameter, machine datum or C-variable ) is to be processed. If, for example, the value of R-parameter R200, which is used in the NC-program of axis 1, is to be read, the ACHS parameter must have the value "1" ( L/S =1, FUNK = RP, NR = 200 ).

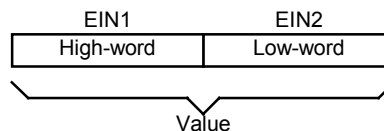
ACHS	Meaning
1	Axis 1 ( = NC-program 1 )
2	Axis 2 ( = NC-program 2 )
3	Auxiliary axis ( = auxiliary NC-program )

**EIN1 and EIN2**

The transfer parameters EIN1 and EIN2 are used to transfer the value or the bit pattern, which the parameter should have on the HNC. Two transfer parameters are required, as the values of the R-parameters, machine data, C/B-variables and process data and also the bit pattern of the digital inputs and outputs each take 4 bytes. These transfer parameters are only relevant, if L/S is equal to zero ( write access ).

For the functions RP, MD, CV, BV and PG, EIN1 must contain the high-word of the value, EIN2 the low-word.

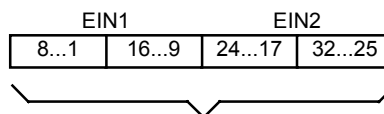
**FUNK = RP, MD, CV, BV, PG**



Example: Value = 100000 ( decimal ) = 186A0 ( hexadecimal )  
 EIN1 = 0001H  
 EIN2 = 86A0H

For the functions ES, ER, AS and AR, EIN1 must contain the digital inputs or outputs 1 to 16 of the HNC in a bit-coded form, and EIN2 the digital inputs or outputs 17 to 32.

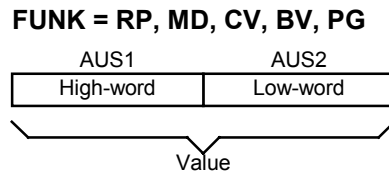
**FUNK = ES, ER, AS, AR**



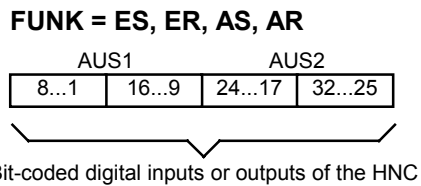
Example: Set input 2, 15, 18 and 32  
 Bit pattern = 00000010 01000000 00000010 10000000  
 Bit pattern ( hexadecimal ) = 02 40 02 80  
 EIN1 = 0240H  
 EIN2 = 0280H

**AUS1 and AUS2**

In the case of a read access ( L/S not equal to zero ), the value or the bit pattern of a parameter is saved in the transfer parameters AUS1 and AUS2. Should an error occur while FC 10 is being executed, AUS1 will contain a corresponding error number.



Example: AUS1 = 0000H  
 AUS2 = 772EH  
 Value = 772E ( hexadecimal ) = 30510 ( decimal )



Example: Inputs on the HNC were read  
 AUS1 = 3210H  
 AUS2 = 5401H  
 Bit pattern ( hexadecimal ) = 32 10 54 01  
 Bit pattern = 00110010 00010000 01010100 00000001  
 The inputs 2, 5, 6, 13, 19, 21, 23 and 25 are set

**Format of transferred values**

All the values of parameters, which are to be transmitted, are transferred in a fixed-point format with three digits after the decimal point. This means that when reading the value of an R-parameter, which has a value of 125.35 on the HNC, the resulting value for the R-parameter in the transfer parameters AUS1 and AUS2 is 125350. The actual value of the parameter can be determined by dividing the value by 1000.

The same is valid for changing the value of a parameter. If, for example, the value of an R-parameter is to be changed to 180, the value which must be saved for the R-parameter in EIN1 and EIN2 is 180000 ( = 180 x 1000 ).

**SEND**

The SEND output bit signals that FC 10 transferred a user data block to the HNC 100 and that the associated reply is awaited.

SEND	Meaning
0	Reply received
1	Reply awaited

ENDE

The ENDE output bit signals whether a data exchange with the HNC 100 has been completed. This means that a user data block was sent, the associated reply received and data included in AUS1 and AUS2 ( L/S not equal to zero ).

ENDE	Meaning
0	Data exchange is going on
1	Data exchange completed

OK

The OK indication bit provides information on whether an error has occurred during the execution of FC 10 or the data exchange. If OK is equal to 1, no error occurred during the processing of FC 10 or the data exchange. However, in case an error occurred, OK is equal to zero and AUS1 contains a corresponding error number. The user must react to an error as required.

Error number	Meaning
8090	I/O address entered in the DB is incorrect
8093	Data block ( 8 bytes ) for the HNC is not consistent
80A0	Error on the HNC
80B0	HNC cannot be addressed via the Profibus
80B1	I/O address entered in the DB is incorrect
F000	Specified HNC is invalid
F001	HNC does not respond (timeout elapsed)
F002	Error on HNC ( e.g. encoder error )
F003	Invalid axis number specified
F004	Invalid I/O card specified
F005	Invalid function specified
F006	HNC not coupled to Profibus ( L2 = 0 )
F007	Invalid C-variable number
F008	Invalid B-variable number or process datum number specified
FFCC	Process datum is not defined
FFD1	Function is not defined ( transfer protocol )
FFD2	Invalid flag number
FFD3	Writing of a process datum is impossible
FFD4	Writing of digital inputs is impossible
FFD5	Invalid B-variable number
FFD6	Invalid axis number specified
FFD7	Invalid C-variable number
FFD8	Value of the machine datum too small
FFD9	Value of the machine datum too great
FFDA	Value of the R-parameter too small
FFDB	Value of the R-parameter too great
FFFC	Machine datum not defined
FFFD	R-parameter not defined

Overview of possible errors with FC 10

**Caution:**

If a data word is used for each of the transfer parameters AUS1 and AUS2, it must be transferred with the absolute address of FC 10 ( e.g. DB20.DBW10 ). This is required, because data block DB 10 is called in FC 10.

Summary

The following table shows the possible combinations of the individual

transfer parameters in a summarized form. The parameters marked with an "X" in this table are relevant for the execution of the individual functions ( FUNK ).

FUNK	L/S	NR	ACHS	EIN1/EIN2 <sup>1</sup>	AUS1/AUS2 <sup>2</sup>
RP	X	X	X	X	X
MD	X	X	X	X	X
CV	X	X	X	X	X
BV	X	X		X	X
PG	X	X		X	X
ES	X	X		X	X
ER	X	X		X	X
AS	X	X		X	X
AR	X	X		X	X
MS	X	X			X
MR	X	X			X

1) The value in EIN1/EIN2 is only considered, if L/S is equal to zero

2) A value is only output via AUS1/AUS2, if L/S is not equal to zero

Possible combinations of transfer parameters

### 8.4.3 Technical data

Function number	FC 10
Function name	KODPS7
Nesting depth	2 ( function SFC 14 and SFC15 are called )
Assigned data blocks	DB 10
Local data	26 bytes
Assigned timers <sup>1)</sup>	T ( HNC addresses )
Average running time (S7-300)	CPU 315-2 DP: 0.4 ms

1) For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S7-300. Thus, timers T5 and T12 are assigned.

### 8.4.4 Application examples

#### Fetching a process datum from the HNC

The value of process datum 3 ( = actual position value Axis 1 ) is to be fetched from the HNC 100 ( address = 11 ), when a rising edge has been detected on input E0.7 and FC 10 does not send ( SEND = 0 ).

The value of the process datum is included in flag words MW21 and MW23, when FC 10 has been completed without error ( ENDE = 1, OK = 1 ). The related S7-program is shown in the following.

```
// Flag M0.1 is set when a rising edge has been detected at input E0.7 and
// function FC 10 does not send (SEND = 0).
    U    E    0.7
    UN   M    0.0
    UN   A    0.1
    S    M    0.1
// The status of input E0.7 is saved in flag M0.1.
    U    E    0.7
    =    M    0.0
// When flag M0.1 is set, function FC 10 is called.
```

```

    U      M      0.1
    SPB    FC10
    BEA
// The address of the HNC is loaded.
FC10: L      11
    T      MB     10
// The direction of data is loaded (read).
    L      1
    T      MB     11
// The function is loaded (process datum).
    L      'PG'
    T      MW     12
// The number of the process datum is loaded.
    L      3
    T      MW     14
// Function FC 10 is called.
CALL FC     10
    HNC :=MB10
    L_S :=MB11
    FUNK:=MW12
    NR  :=MW14
    ACHS:=MB16
    EIN1:=MW17
    EIN2:=MW19
    AUS1:=MW21
    AUS2:=MW23
    SEND:=A0.1
    ENDE:=A0.2
    OK  :=A0.3
// When the ENDE-bit of FC 10 is set, flag M0.1 is reset.
    U      A      0.2
    R      M      0.1

```

### Further examples

Further application examples are included in the description of function FC 11 in chapter 8.5.4.

## 8.5 Function FC 11 (communication function)

### 8.5.1 General

By integrating function FC 11 in an S7 program the user can read or change values of parameters on the HNC 100 ( R-parameters, machine data, C- and B-variables, process data, digital inputs and outputs as well as flags ). The user can define in the transfer parameters of FC 11, which parameter is to be processed.

To change or read the value of a parameter, FC 11 must be called at least **twice**. In the first call, a user data block is compiled with the help of the transfer parameters and transferred to the HNC 100. In the subsequent call of FC 11 a check is carried out to verify whether the related reply was received. If the reply was not received, the check is repeated in the next calls until the reply has been received or the timeout counter ( output of an error message ! ) has elapsed. After that, the entire process can start over again.

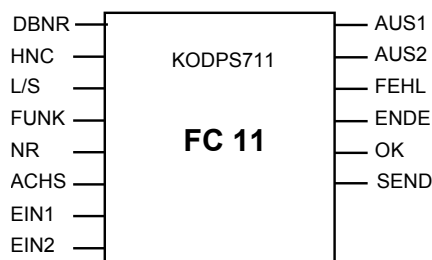
#### Remark

The task of function FC 10 is the same as that of function FC 11. Function FC 10 assumes that the I/O addresses, via which data are exchanged with the individual HNCs, are contained in data block DB 10. With function FC 11, the I/O addresses can be included in any data block. The number of the data block is signaled to function FC 11 with transfer parameter DBNR ( cf. chapter 8.4 ). Reading or changing parameters on the HNC 100 should be carried out using **only** function FC 11 or **only** function FC 10.

#### Caution:

FC 11 may only be used in the cyclical program execution ( OB 1 ). It **cannot** be integrated in the starting organization block ( e.g. OB 100 ).

### 8.5.2 Parameterization of FC 11



Parameter	Description	Nature	Type
DBNR	Number of the data block	in	INT
HNC	Address of the HNC	in	BYTE
L/S	Distinction of direction of data	in	BYTE
FUNK	Type of the parameter	in	WORD
NR	Number of the parameter	in	WORD
ACHS	Number of the axis	in	BYTE
EIN1	Input value 1	in	WORD
EIN2	Input value 2	in	WORD
AUS1	Output value 1	out	WORD
AUS2	Output value 2	out	WORD
FEHL	Error output	out	WORD
ENDE	Output bit "ENDE"	out	BOOL
OK	Output bit "OK"	out	BOOL
SEND	Output bit "SEND"	in_out	BOOL

## Transfer parameters of FC 11

- DBNR** (number of the data block) This transfer parameter contains the number of the data block, in which the I/O addresses are saved in defined data words, via which data are exchanged via the individual HNCs ( cf. chapter 8.1 ).
- HNC** With the HNC transfer parameter, the user determines, which HNC 100 is to process a parameter. It may have values ranging from 0 to 126. With the help of this parameter, data can be loaded, which are saved for each individual HNC in the transferred DB ( see **DBNR** parameter ) ( e.g.: via which I/O address the HNC 100 can be addressed. ).
- L/S** The L/S transfer parameter signals whether the value of the relevant parameter is to be read or written on the HNC. If L/S is equal to zero, a write access is executed. However, if L/S is not equal to zero, the corresponding parameter value is read.

L/S	Meaning
0	Parameter value is written
not equal to 0	Parameter value is read

- FUNK** The FUNK transfer parameter determines which type of parameter is to be processed by the HNC. With regard to digital inputs and outputs and flags, this parameter is also used to define, whether these are to be set or reset. However, this is only valid, if these are to be written ( L/S = 0 ). However, if L/S is not equal to zero when the digital inputs and outputs or flags are accessed, these are simply read on the HNC and transferred.



<b>FUNK</b>	<b>Meaning</b>
RP	R-parameter
MD	Machine datum
CV	C-variable
BV	B-variable
PG	Process datum
ES	Set digital inputs
ER	Reset digital inputs
AS	Set digital outputs
AR	Reset digital outputs
MS	Set flags
MR	Reset flags
KX	x-value of a curve point
KY	y-value of a curve point

**Caution:**

When changing values of machine data, please note that in the case of some machine data, the new value is only taken over, when a **reset** has been triggered on the HNC 100.

**NR**

The NR transfer parameter is used to determine for functions RP, MD, CV, BV, PG, MS and MR, which number of the parameter is to be processed. ( e.g. R-parameter 200: NR = 200, machine datum 13: NR = 13, C-variable 20: NR = 20, B-variable 10: NR = 10, flag 100: NR = 100 ). In conjunction with functions ES, ER, AS and AR, "NR" is used to declare, which I/O card is to be accessed. At present, two I/O cards are available on the HNC100. Each I/O card consists of 32 I/Os. The first I/O card comprises the hardware I/Os. These hardware I/Os are addressed in the NC program via E1.1 - E1.32 and A1.1 - A1.32, respectively. The hardware inputs ( E1.1 - E1.32 ) can only be queried using function FC 11; the hardware outputs ( A1.1 - A1.32 ) can be queried and set. The second I/O card comprises the so-called virtual I/Os. These I/Os are addressed in the NC program via E2.1 - E2.32 and A1.1 - A1.32, respectively. The virtual inputs ( E2.1 - E2.32 ) and outputs ( A2.1 - A2.32 ) can be set and queried using function FC 11. For functions ES, ER, AS and AR, the NR parameter must contain the following values for accessing the hardware I/Os or virtual I/Os:

<b>NR</b>	<b>Meaning ( FUNK = ES, ER, AS or AR )</b>
1	Hardware I/Os ( E1.1 - E1.32 and A1.1 - A1.32 )
2	Virtual I/Os ( E2.1 - E2.32 and A2.1 - A2.32 )

In the case of function PG, the NR parameter must contain the number of the process datum whose value is to be read. If, for example, the current value of the actual position value is to be determined, the NR parameter must contain "3" as a value ( L/S = 1, FUNK = PG ). The assignment of the individual process data numbers is listed in the following table.

Number of the process datum	Meaning
3	Actual position value Axis 1
4	Actual position value Axis 2 ( 2-axis version )
5	Actual pressure value Axis 1
6	Actual pressure value Axis 2 ( 2-axis version )
7	Position command value Axis 1
8	Position command value Axis 2 ( 2-axis version )
9	Pressure command value Axis 1
10	Pressure command value Axis 2 ( 2-axis version )
11	Peak value Minus Axis 1
12	Peak value Minus Axis 2 ( 2-axis version )
13	Peak value Plus Axis 1
14	Peak value Plus Axis 2 ( 2-axis version )
15	Actual value Encoder 1
16	Actual value Encoder 2 ( 2-axis version )
17	Actual value Encoder 3 ( 2-axis version )
18	Actual value Encoder 4
19	Actual value Encoder 5
20	Actual value Encoder 6
21	Actual value Encoder 7
22	Actual value Encoder 8
23	Actual value Encoder 9
24	Actual value Encoder 10
25	Actual value Encoder 11
26	Following error Axis 1
27	Following error Axis 2 ( 2-axis version )
28	Synchronization error ( 2-axis version )
29	Target position value Axis 1
30	Target position value Axis 2 ( 2-axis version )
31	Actual velocity value Axis 1
32	Actual velocity value Axis 2 ( 2-axis version )
33	DAC 1
34	DAC 2
35	DAC 3
36	DAC 4
37	Actual value Encoder 4 ( Volt )
38	Actual value Encoder 5 ( Volt )
39	Actual value Encoder 6 ( Volt )
40	Actual value Encoder 7 ( Volt )
41	Actual value Encoder 8 ( Volt )
42	Actual value Encoder 9 ( Volt )
43	Actual value Encoder 10 ( Volt )
44	Actual value Encoder 11 ( Volt )

In conjunction with functions KX and KY, parameter NR is used to transfer the number of the curve point that is to be read or changed. If, for example, the x-value of the 12<sup>th</sup> curve point of curve 4 is to be read, parameter NR must contain the value "12" ( L/S = 1, FUNK = KX and ACHS = 4 ).

#### ACHS

The ACHS transfer parameter is only relevant for the functions RP, MD, CV, KX and KY. In conjunction with functions RP, MD and CV, this parameter is used to distinguish, on which axis (= NC-program) the relevant parameter ( R-parameter, machine datum or C-variable ) is to be processed. If, for example, the value of R-parameter R200, which is used in the NC-program of axis 1, is to be read, the ACHS parameter must have the value "1" ( L/S = 1, FUNK = RP, NR = 200 ).

ACHS	Meaning ( FUNK = RP, MD or CV )
1	Axis 1 ( = NC-program 1 )
2	Axis 2 ( = NC-program 2 )
3	Auxiliary axis ( = auxiliary NC-program )

In conjunction with functions KX and KY, parameter ACHS is used to transfer the number of the curve, of which a curve point is to be read or changed. If, for example, the x-value of the 12<sup>th</sup> curve point of curve 4 is to be read, parameter NR must contain the value "4" ( L/S = 1, FUNK = KX and NR = 12 ).

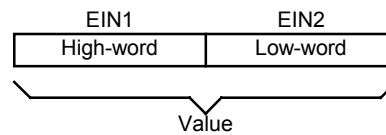
ACHS	Meaning ( FUNK = KX or KY )
1 - 6	Number of the curve

EIN1 and EIN2

The transfer parameters EIN1 and EIN2 are used to transfer the value or the bit pattern, which the parameter should have on the HNC. Two transfer parameters are required, as the values of the R-parameters, machine data, C/B-variables, process data and curve points and also the bit pattern of the digital inputs and outputs each take 4 bytes. These transfer parameters are only relevant, if L/S is equal to zero ( write access ).

For the functions RP, MD, CV, BV, PG, KX and KY, "EIN1" must contain the high-word of the value, EIN2 the low-word.

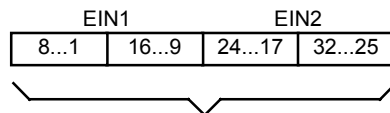
**FUNK = RP, MD, CV, BV, PG, KX, KY**



Example: Value = 100000 ( decimal ) = 186A0 ( hexadecimal )  
 EIN1 = 0001H  
 EIN2 = 86A0H

For the functions ES, ER, AS ad AR, "EIN1" must contain the digital inputs or outputs 1 to 16 of the HNC in a bit-coded form, and EIN2 the digital inputs or outputs 17 to 32.

**FUNK = ES, ER, AS, AR**

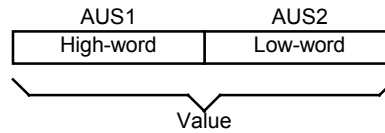


Example: Set input 2, 15, 18 and 32  
 Bit pattern = 00000010 01000000 00000010 10000000  
 Bit pattern ( hexadecimal ) = 02 40 02 80  
 EIN1 = 0240H  
 EIN2 = 0280H

AUS1 and AUS2

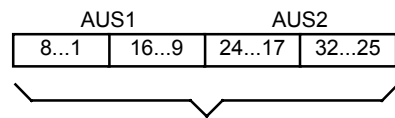
In the case of a read access ( L/S not equal to zero ) the value or the bit pattern of a parameter is saved in the transfer parameters AUS1 and AUS2.

**FUNK = RP, MD, CV, BV, PG, KX, KY**



Example: AUS1 = 0000H  
 AUS2 = 772EH  
 Value = 772E ( hexadecimal ) = 30510 ( decimal )

**FUNK = ES, ER, AS, AR**



Example: Inputs on the HNC were read  
 AUS1 = 3210H  
 AUS2 = 5401H  
 Bit pattern ( hexadecimal ) = 32 10 54 01  
 Bit pattern = 00110010 00010000 01010100 00000001  
 The inputs 2, 5, 6, 13, 19, 21, 23 and 25 are set

Format of transferred values

All the values of parameters, which are to be transmitted, are transferred in a fixed-point format with three digits after the decimal point. This means that when reading the value of an R-parameter, which has a value of 125.35 on the HNC, the resulting value for the R-parameter in the transfer parameters AUS1 and AUS2 is 125350. The actual value of the parameter can be determined by dividing the value by 1000.

The same is valid for changing the value of a parameter. If, for example, the value of an R-parameter is to be changed to 180, the value which must be saved for the R-parameter in EIN1 and EIN2 is 180000 ( = 180 x 1000 ).

FEHL ( error )

This transfer parameter includes an error number, if an error has occurred during the execution of function FC 11. An overview of possible errors is given in the description of the OK parameter.

ENDE

The ENDE output bit signals whether a data exchange with the HNC 100 has been completed. This means that a user data block was sent, the associated reply received and data included in AUS1 and AUS2 ( L/S not equal to zero ).

ENDE	Meaning
0	Data exchange is going on
1	Data exchange completed

OK

The OK indication bit provides information on whether an error has occurred during the execution of FC 11 or the data exchange. If OK is equal to 1, no error occurred during the processing of FC 11 or the data exchange. However, in case an error occurred, OK is equal to zero and FEHL contains a corresponding error number. The user must react to an error as required.

Error number	Meaning
8090	I/O address entered in the DB is incorrect
8093	Data block ( 8 bytes ) for the HNC is not consistent
80A0	Error on the HNC
80B0	HNC cannot be addressed via the Profibus
80B1	I/O address entered in the DB is incorrect
F000	Specified HNC is invalid
F001	HNC does not respond (timeout elapsed)
F002	Error on HNC ( e.g. encoder error )
F003	Invalid axis number specified
F004	Invalid I/O card specified
F005	Invalid function specified
F006	HNC not coupled to Profibus ( L2 = 0 )
F007	Invalid C-variable number
F008	Invalid B-variable number or process datum number specified
F009	Curve number not permitted
F00A	Curve point number not permitted
FFAC	The curve point could not be sent, because the curve is being updated (-> nc command: KURVE ... UPDATE) or new curve points are received over the serial interface.
FFB1	Sending of a complete curve has not been started
FFB2	x-value for the curve point is not permitted
FFB3	Curve point is not defined
FFCC	Process datum is not defined
FFD1	Function is not defined ( transfer protocol )
FFD2	Invalid flag number
FFD3	Writing of a process datum is impossible
FFD4	Writing of digital inputs is impossible
FFD5	Invalid B-variable number
FFD6	Invalid axis number specified
FFD7	Invalid C-variable number
FFD8	Value of the machine datum too small
FFD9	Value of the machine datum too great
FFDA	Value of the R-parameter too small
FFDB	Value of the R-parameter too great
FFE4	Curve is not defined
FFFC	Machine datum not defined
FFFD	R-parameter not defined

Overview of possible errors with FC 11

SEND

The SEND output bit signals that FC 11 transferred a user data block to the HNC 100 and that the associated reply is awaited.

SEND	Meaning
0	Reply received
1	Reply awaited

**Caution:**

If a data word is used for each of the transfer parameters AUS1 and AUS2, it must be transferred with the absolute address of FC 11 ( e.g. DB20.DBW10 ). This is required, because the data block, whose number is included in transfer parameter DBNR, is opened in FC 11.

**Summary**

The following table shows the possible combinations of the individual transfer parameters in a summarized form. The parameters marked with an "X" in this table are relevant for the execution of the individual functions ( FUNK ).

FUNK	L/S	NR	ACHS	EIN1/EIN2 <sup>1</sup>	AUS1/AUS2 <sup>2</sup>
RP	X	X	X	X	X
MD	X	X	X	X	X
CV	X	X	X	X	X
BV	X	X		X	X
PG	X	X		X	X
ES	X	X		X	X
ER	X	X		X	X
AS	X	X		X	X
AR	X	X		X	X
MS	X	X			X
MR	X	X			X
KX	X	X	X	X	X
KY	X	X	X	X	X

<sup>1</sup>) The value in EIN1/EIN2 is only considered, if L/S is equal to zero

<sup>2</sup>) A value is only output via AUS1/AUS2 if L/S is not equal to zero

**Possible combinations of transfer parameters****8.5.3 Technical data**

Function number	FC 11
Function name	KODPS711
Nesting depth	2 ( function SFC 14 and SFC15 are called )
Assigned data blocks	Data block, whose number is written in transfer parameter DBNR
Local data	26 bytes
Assigned timers <sup>1)</sup>	T (HNC addresses )
Average running time (S7-300)	CPU 315-2 DP: 0.4 ms

<sup>1)</sup> For each HNC that is connected to the Profibus-DP a timer must be used. The number of the timer is identical with the address of the HNC. Example: Two HNCs with the addresses 5 and 12 are connected via the Profibus-DP with the S7-300. Thus, timers T5 and T12 are assigned.

## 8.5.4 Application examples

### Fetching a process datum from the HNC

The value of process datum 3 ( = actual position value Axis 1 ) is to be fetched from the HNC 100 ( address = 11 ), when a rising edge has been detected on input E0.7 and FC 11 does not send ( SEND = 0 ).

The value of the process datum is included in flag words MW21 and MW23, when FC 11 has been completed without error ( ENDE = 1, OK = 1 ). The related S7-program is shown in the following.

```
// Flag M0.1 is set when a rising edge has been detected at input E0.7 and
// function FC 11 does not send (SEND = 0).
    U     E     0.7
    UN    M     0.0
    UN    A     0.1
    S     M     0.1
// The status of input E0.7 is saved in flag M0.1.
    U     E     0.7
    M 0.0
// When flag M0.1 is set, function FC 11 is called.
    U     M     0.1
    SPB   FC11
    BEA
// The address of the HNC is loaded.
FC11: L     11
    T     MB    10
// The direction of data is loaded (read).
    L     1
    T     MB    11
// The function is loaded (process datum).
    L     'PG'
    T     MW    12
// The number of the process datum is loaded.
    L     3
    T     MW    14
// Function FC 11 is called.
    CALL  FC    11
    DBNR:=10
    HNC :=MB10
    L_S :=MB11
    FUNK:=MW12
    NR  :=MW14
    ACHS:=MB16
    EIN1:=MW17
    EIN2:=MW19
    AUS1:=MW21
    AUS2:=MW23
    FEHL:=MW25
    ENDE:=A0.2
    OK  :=A0.3
    SEND:=A0.1
// When the ENDE-bit of FC 11 is set, flag M0.1 is reset.
    U     A     0.2
    R     M     0.1
```

### Fetching the value of an R-parameter from the HNC

The value of R-parameter R200, which is used in the NC-program of axis 1, is to be fetched. For this, function FC 11 must be called as follows:

```
L/S = 1
FUNK = RP
NR = 200
ACHS = 1
EIN1 = optional (is ignored, because L/S = 1)
EIN2 = optional (is ignored, because L/S = 1)
```

The value of R-parameter R200 is received as 32-bit value in parameters AUS1 and AUS2. The received value must be interpreted with three digits after the decimal point. If, for example, the value 313500 is written in parameters AUS1 and AUS2, the value of the R-parameter on the HNC is 313.5.

### Sending the value of an R-parameter to the HNC

A new value is to be transferred for R-parameter R400 that is used in the auxiliary NC-program. The value of the R-parameter is to be set to 100.4. Since R-parameter values are transferred with three digits after the decimal point, 100400 must be sent. For this, function FC 11 must be called as follows:

```
L/S = 0
FUNK = RP
NR = 400
ACHS = 3
EIN1 = 0001 (decimal: 100400 = hexadecimal: 18830)
EIN2 = 8830
```

### Fetching the value of a machine datum from the HNC

The value of machine datum M013 is to be fetched from axis 1 ( = inching velocity plus Axis 1 ). For this, function FC 11 must be called as follows:

```
L/S = 1
FUNK = MD
NR = 13
ACHS = 1
EIN1 = optional (is ignored, because L/S = 1)
EIN2 = optional (is ignored, because L/S = 1)
```

The value of machine datum M013 is received as 32-bit value in parameters AUS1 and AUS2. The received value must be interpreted with three digits after the decimal point. If, for example, the value 20000 is written in parameters AUS1 and AUS2, an inching velocity of 20.000 units/s is set on the HNC.



### Fetching the value of a C-variable from the HNC

The value of C-variable C30, which is used in the NC-program of the auxiliary axis, is to be fetched. For this, function FC 11 must be called as follows:

```
L/S = 1
FUNK = CV
NR = 30
ACHS = 3
EIN1 = optional (is ignored, because L/S = 1)
EIN2 = optional (is ignored, because L/S = 1)
```

The value of C-variable C30 is received as 32-bit value in parameters AUS1 and AUS2. The received value must be interpreted with three digits after the decimal point. If, for example, the value 106100 is written in the parameters AUS1 and AUS2, the value of the C-variable on the HNC is 106.100.

### Sending the value of a C-variable to the HNC

A new value is to be transferred for C-variable C45 that is used in the NC-program of axis 1. The value of the C-variable is to be set to 310.5. Since C-variable values are transferred with three digits after the decimal point, 310500 must be sent. For this, function FC 11 must be called as follows:

```
L/S = 0
FUNK = CV
NR = 45
ACHS = 1
EIN1 = 0004 (decimal: 310500 = hexadecimal: 4BCE4)
EIN2 = BCE4
```

### Fetching the value of a B-variable from the HNC

The value of B-variable B31 is to be fetched. For this, function FC 11 must be called as follows:

```
L/S = 1
FUNK = BV
NR = 31
ACHS = optional (B-variables are axis-independent)
EIN1 = optional (is ignored, because L/S = 1)
EIN2 = optional (is ignored, because L/S = 1)
```

The value of B-variable B31 is received as 32-bit value in the parameters AUS1 and AUS2. The received value must be interpreted with three digits after the decimal point. If, for example, the value 281200 is written in the parameters AUS1 and AUS2, the value of the C-variable on the HNC is 281.200.

### Sending the value of a B-variable to the HNC

A new value for B-variable B10 is to be transferred. The value of the B-variable is to be set to 450.11. Since B-variable values are transferred with three digits after the decimal point, 450110 must be sent. For this, function FC 11 must be called as follows:

```
L/S = 0
FUNK = CV
NR = 45
ACHS = 1
EIN1 = 0006 (decimal: 450110 = hexadecimal: 6DE3E)
EIN2 = DE3E
```

### Fetching digital inputs from the HNC

The status of the hardware inputs is to be fetched ( the hardware inputs are queried via E1.1 - E1.32 in the NC-program ). For this, function FC 11 must be called as follows:

```
L/S = 1
FUNK = ES (oder ER)
NR = 1 (1 = E1.1-E1.32, 2 = E2.1-E2.32)
ACHS = optional (inputs are axis-independent)
EIN1 = optional (is ignored, because L/S = 1)
EIN2 = optional (is ignored, because L/S = 1)
```

The status of inputs E1.1 -E1.32 is received as 32-bit value in the parameters AUS1 and AUS2. The inputs are written as follows in AUS1 and AUS2.

AUS1		AUS2	
E1.8 - E1.1	E1.16 - E1.9	E1.24 - E1.17	E1.32 - E1.25

If, for example, AUS1 includes the value 0300 ( hexadecimal ), the hardware inputs E1.1 and E1.2 are set on the HNC.

If the status of the virtual inputs is to be fetched, the value "2" must be transferred with parameter NR when function FC 11 is called. ( The virtual inputs are queried via E2.1 - E2.32 in the NC-program ).

### Setting digital inputs on the HNC

Function FC 11 does not allow the setting of hardware inputs ( E1.1 - E1.32 ), but only the setting of virtual inputs ( E2.1 - E2.32 ). Which inputs are to be set is transferred as bit pattern in the parameters EIN1 and EIN2. The following assignment is valid:

EIN1		EIN2	
E2.8 - E2.1	E2.16 - E2.9	E2.24 - E2.17	E2.32 - E2.25

If, for example, inputs E2.1 and E2.9 and E2.18 are to be set, the value 0101 ( hexadecimal ) = 0000 0001 0000 0001 ( binary ) must be saved in EIN1, and 0200 ( hexadecimal ) = 0000 0010 0000 0000 ( binary ) in EIN2. For this, function FC 11 must be called as follows:

```

L/S = 0
FUNK = ES
NR = 2
ACHS = optional (inputs are axis-independent)
EIN1 = 0101
EIN2 = 0200
    
```

### Resetting digital inputs on the HNC

Function FC 11 does **not** allow the resetting of hardware inputs ( E1.1 - E1.32 ), but only the resetting of virtual inputs ( E2.1 - E2.32 ). Which inputs are to be reset is transferred as bit pattern in the parameters EIN1 and EIN2. The following assignment is valid:

EIN1		EIN2	
E2.8 - E2.1	E2.16 - E2.9	E2.24 - E2.17	E2.32 - E2.25

If, for example, inputs E2.2 and E2.16 and E2.17 are to be reset, the value 0280 ( hexadecimal ) = 0000 0010 1000 0000 ( binary ) must be saved in EIN1, and 0100 ( hexadecimal ) = 0000 0001 0000 0000 ( binary ) in EIN2. For this, function FC 11 must be called as follows:

```

L/S = 0
FUNK = ER
NR = 2
ACHS = optional (inputs are axis-independent)
EIN1 = 0280
EIN2 = 0100
    
```

### Fetching digital outputs from the HNC

The status of the hardware outputs is to be fetched ( the hardware outputs are queried via A1.1 - A1.32 in the NC-program ). For this, function FC 11 must be called as follows:

```

L/S = 1
FUNK = AS          (or AR)
NR = 1              (1 = A1.1-A1.32, 2 = A2.1-A2.32)
ACHS = optional (outputs are axis-independent)
EIN1 = optional (is ignored, because L/S = 1)
EIN2 = optional (is ignored, because L/S = 1)
    
```

The status of outputs A1.1 -A1.32 is received as 32-bit value in the parameters AUS1 and AUS2. The outputs are written as follows in AUS1 and AUS2.

AUS1		AUS2	
A1.8 - A1.1	A1.16 - A1.9	A1.24 - A1.17	A1.32 - A1.25

If, for example, AUS1 includes the value 0300 ( hexadecimal ), the hardware outputs A1.1 and A1.2 are set on the HNC.

If the status of the virtual outputs is to be fetched, the value "2" must be transferred with parameter NR when function FC 11 is called. ( The virtual outputs are queried via A2.1 - A2.32 in the NC-program ).

### Fetching the status of a flag from the HNC

The status of flag M100 is to be fetched. For this, function FC 11 must be called as follows:

```
L/S = 1
FUNK = MS      (or MR)
NR = 100
ACHS = optional (flags are axis-independent)
EIN1 = optional (is ignored, because L/S = 1)
EIN2 = optional (is ignored, because L/S = 1)
```

The status of the flag is received in the parameters AUS1 and AUS2. If AUS1 and AUS2 contain the value 1, the flag is set; if the value is 0, the flag is reset.

### Setting a flag on the HNC

Flag M200 is to be set. For this, function FC 11 must be called as follows:

```
L/S = 0
FUNK = MS
NR = 200
ACHS = optional (flags are axis-independent)
EIN1 = optional (is ignored)
EIN2 = optional (is ignored)
```

### Resetting a flag on the HNC

Flag M300 is to be reset. For this, function FC 11 must be called as follows:

```
L/S = 0
FUNK = MR
NR = 300
ACHS = optional (flags are axis-independent)
EIN1 = optional (is ignored)
EIN2 = optional (is ignored)
```

### Fetching the x- or y-value of a curve point from the HNC

The x-value of curve point 10 of curve 3 is to be fetched. For this, function FC 11 must be called as follows:

```
L/S = 1
FUNK = KX      (KY for y-value)
NR   = 10      (Number of the curve point)
ACHS = 3      (Number of the curve)
EIN1 = optional (is ignored, because L/S = 1)
EIN2 = optional (is ignored, because L/S = 1)
```

The x-value of the curve point is received as 32-bit value in the parameters AUS1 and AUS2. The received value must be interpreted with three digits after the decimal point. If, for example, the parameters AUS1 and AUS2 contain the value 140500, the x-value of the curve point is 140.5.

### Setting the x- or y-value of a curve point on the HNC

A new value is to be transferred for the x-value of curve point 5 of curve 2. The x-value is to be set to 210.4. Since x-and y-values of the curve points are transferred with three digits after the decimal point, 210400 must be sent. For this, function FC 11 must be called as follows:

```
L/S = 0
FUNK = KX      (KY for y-value)
NR   = 5       (Number of the curve point)
ACHS = 2      (Number of the curve)
EIN1 = 0003   (decimal: 210400 = hexadecimal: 335E0)
EIN2 = 35E0
```

After a curve point was changed, the NC-command **KURVE .. UPDATE** must be called on the HNC in order that the changed curve is taken over by the HNC.

The curve points are changed only temporarily in the RAM of the HNC. After a **reset**, the curves are again reset to their default values saved in the flash.

### Sending a complete curve to the HNC

If a complete curve is to be sent, it is possible to suppress the X-value checking of the individual curve points. Usually, when an x-value of a curve point is being sent, a check is carried out to determine whether the x-value is greater than the x-value of the preceding curve point and smaller than the x-value of the subsequent curve point. This check can be disturbing, when a complete curve is sent and the x-range of the curve is shifted.

The x-value check can be switched off by calling function FC 11 as follows:

```
L/S = 0
FUNK = KX          (or KY)
NR   = 0           (0 = Switch off check)
ACHS = 5           (Number of the curve)
EIN1 = optional    (is ignored)
EIN2 = optional    (is ignored)
```

After this call, the individual curve points are taken over without verification of the x-values. The individual x- and y-values of the curve points are now transferred by several calls of function FC 11.

When the complete curve has been transferred, the received x-values of the individual curve points must be checked. This process checks for all curve points, whether an x-value is greater than the x-value of the preceding curve point and smaller than the x-value of the subsequent curve point. The x-value check can be switched on by calling function FC 11 as follows:

```
L/S = 0
FUNK = KX          (or KY)
NR   = 4095        (4095 = Switch on check)
ACHS = 5           (Number of the curve)
EIN1 = optional    (is ignored)
EIN2 = optional    (is ignored)
```

If an error is detected during the check, error number FFB2 will be written in parameter FEHL.

As long as a complete curve is being sent ( checking of x-values switched off ), the curve is **not** taken over by the HNC with NC-command **KURVE .. UPDATE**. Only when the complete curve has been sent and the x-values of the curve points have been checked and no error has occurred can the curve be taken over using the NC-command **KURVE .. UPDATE**.

If the check of the x-values detects an error, the curve must be resent with corrected x-values and the x-values must be rechecked. Only then can be curve be taken over using the command **KURVE .. UPDATE**.

## 9. Diagnosis data from the HNC 100

### 9.1 Structure of diagnosis data

If an error occurs on the HNC 100 (e.g. encoder error or traversing range limit), the HNC 100 outputs an external diagnosis message via the Profibus-DP. This external diagnosis message is saved in the DP master and can be read out there. The external diagnosis message is structured to DIN 19245, part 3

Byte	Meaning
0	Station status 1
1	Station status 2
2	Station status 3
3	Master address
4	Ident. number ( high-byte )
5	Ident. number ( low-byte )
6	External diagnosis
-	( diagnosis with a length of
16	11 bytes from HNC 100 )

} Meaning determined to  
DIN 19245 part 3

At present, only the bytes 7, 8, 10, 14 and 16 are of relevance in the diagnosis message of the HNC 100 (bytes 6 to 16 ). The other bytes are reserved for future extensions.

Byte 6 contains the length ( = 0x0B ) of the device-related diagnosis from the HNC 100. Bytes 7, 8, 10, 14 and 16 contain in a bit-coded form the type of error present on the HNC 100.

#### Byte 7:

Bit 0	EEPROM error
Bit 1	Cyclical data exchange was interrupted ( Profibus )
Bit 2	Software error on the CAN interface
Bit 3	Error while initializing the bus interface.
Bit 4	Wrong checksum in the EEPROM.
Bit 5	Wrong hardware configuration.

#### Byte 8:

Bit 0	Error when recalculating the machine data
Bit 1	Error when recalculating the NC-programs
Bit 2	Address switch defective
Bit 3	Wrong memory contents
Bit 4	Hardware error
Bit 5	Division by 0
Bit 6	RAM error
Bit 7	Hardware error on the CAN interface

**Byte 10:**

Bit 0	Following error Axis 1
Bit 1	Limit switch Axis 1
Bit 2	Traversing range limit + Axis 1
Bit 3	Traversing range limit - Axis 1
Bit 4	Encoder error Axis 1
Bit 5	Synchronization error
Bit 6	Reference run error Axis 1
Bit 7	Encoder jump Axis 1

**Byte 12:**

Bit 0	Following error Axis 2
Bit 1	Limit switch Axis 2
Bit 2	Traversing range limit + Axis 2
Bit 3	Traversing range limit - Axis 2
Bit 4	Encoder error Axis 2
Bit 5	Synchronization error
Bit 6	Reference run error Axis 2
Bit 7	Encoder jump Axis 2

**Byte 14:**

Bit 0	Encoder connection Encoder 1
Bit 1	Encoder connection Encoder 2
Bit 3	Encoder connection Encoder 4
Bit 4	Encoder connection Encoder 5
Bit 5	Encoder connection Encoder 6
Bit 6	Encoder connection Encoder 7

**Byte 16:**

Bit 0	Encoder jump Encoder 1
Bit 1	Encoder jump Encoder 2

**9.2 Fetching diagnosis data (IM 308-C)**

When using master assembly IM 308-C, you cannot fetch diagnosis data directly. To this end, function block FB 192 ( IM308C ) must be used. This function block is included in Siemens' software package COM PROFIBUS. How to parameterize and call FB 192 is explained in detail in the manual of the IM 308-C master assembly.



### **9.3 Fetching diagnosis data (S5-95U)**

When using the S5-95U with DP master interface, you can request an overview of diagnosis data via I/O addresses 56 and 57. You can fetch the complete diagnosis data for a DP slave ( HNC 100 ) using the integrated function block FB 230. How to parameterize and call FB 230 is explained in detail in the manual of the S5-95U with DP master interface.

### **9.4 Fetching diagnosis data (S7-300 and S7-400)**

When using the S7-300 or S7-400 with integrated DP master, you can request the complete diagnosis data for a DP slave ( HNC 100 ) with the help of standard function SFC 13 "DPNRM\_DG". How to parameterize the SFC 13 is explained in detail in the system software manual of the S7-300/400.

## 10. Time

The time required for fetching data from the HNC 100 to the S5 or sending data from the S5 to the HNC 100 is composed of three components:

- Bus cycle time ( Profibus-DP )
- Processing time of the function blocks
- Time required for processing the data on the HNC 100

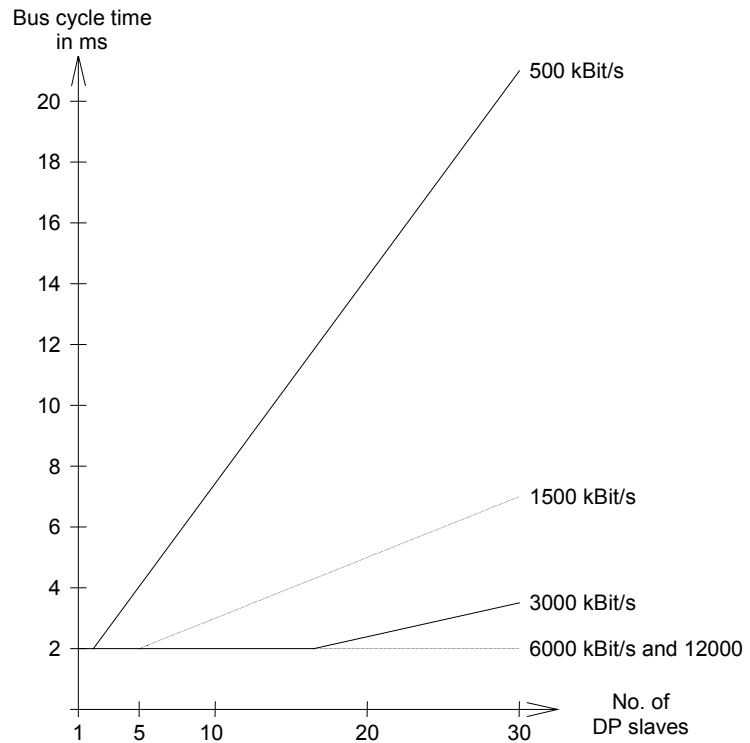
### 10.1 Bus cycle time (Profibus-DP)

The time required for transferring user data from the DP master to a DP slave is determined by the following factors:

- Transfer rate
- Number of DP slaves
- Number of transferred user data bytes ( HNC 100: 8 bytes)

The following diagram shows the theoretical bus cycle time of a DP master (class 1) in dependence upon the number of connected DP slaves, each with input and output data of 8 bytes.

It is assumed that all provided DP slaves are in the user data operating mode and contain no diagnosis messages. After the time  $\min T_{SDR}$  of 11  $t_{bit}$  (minimum delay time of the DP slave) has elapsed, the DP slaves respond. For the DP master (class 1), time  $T_{SDI}$  is assumed to be 37 bit times ( waiting time of the master ).



## 10.2 Processing time of the function blocks

The average processing times of the function blocks depend on the CPU ( S5 ) used. The times in the following table are given as a guideline.

	CPU 943:	CPU 944:	S5-95U	CPU 928A
FB 10	3.3 ms	0.2 ms	3.1 ms	2.3 ms

Remark

For the data exchange between the HNC 100 and S5, FB 10 must be called **at least** twice. In the first call, a telegram is compiled, which contains in a coded form, which parameter is to be fetched from the HNC 100 or changed. This telegram is sent to the HNC 100. In the second call of FB 10 a check is carried out to verify whether the associated reply has been received from the HNC 100.

## 10.3 Time required for processing the data on the HNC 100

The time required by the HNC 100 for processing the data depends on the scanning time set on the HNC 100.

On the HNC 100 the controller has highest priority. This means that the processing of data that were sent via the Profibus-DP can be interrupted. If processing is not interrupted, reading of the actual value can take, for example, 0.5 ms. In the case of an interruption, the time prolongs accordingly.

## 11. GSE file (component master file)

```

;
;
;=====
;GSE file for HNC100 with Profibus-DP interface (slave functionality)
;Status: 10.01.2002
;Contact:  Bosch Rexroth AG
;          BRI-IC/ESY1
;          M. Tuebel
;          Postfach
;          D-97813 Lohr am Main
;          Tel.: 09352/18-3095
;          Fax:  09352/18-2065
;=====
; Version 2.5:
; - Correcting of comments
; - Adding the diagnosis data for the axis 2
;=====
;
;
#Profibus_DP
;
GSD_Revision      = 1
;
Vendor_Name       = "Bosch Rexroth AG"
Model_Name        = "VT-HNC100-Y-2X"
Revision          = "Index 2.5"
Ident_Number      = 0x0476
;
Protocol_Ident    = 0                ; Profibus-DP
Station_Type      = 0                ; DP-Slave
FMS_supp          = 0                ; FMS is not supported
;
Hardware_Release  = "VTS0234-40"
Software_Release  = "5_02"
;
; Supported baud rates
9.6_supp          = 1
19.2_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
;
; Maximum response time for different baud rates (default values)
MaxTsdr_9.6       = 60
MaxTsdr_19.2      = 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
Repeater_Ctrl_Sig = 2
24V_Pins          = 0
;
Implementation_Type = "SPC3"
;

```

```

Freeze_Mode_supp      = 1 ; Freeze-mode is supported
Sync_Mode_supp       = 1 ; Sync-mode is supported
Auto_Baud_supp       = 1 ; Automatic baud rate recognition is supported
Set_Slave_Add_supp   = 0 ; Address assignment via bus impossible
;
Max_Diag_Data_Len    = 17 ; Length of diagnosis data
;
Max_User_Prm_Data_Len = 0x50 ; Maximum length of User_Prm_Data
Ext_User_Prm_Data_Const(0) = 0x00,0x00,0x00,0x00,0x00
;
Slave_Family         = 8 ; The HNC100 is filed under NC/RC.
;
; Minimum rerun time of a call-up telegram to the slave
Min_Slave_Intervall = 10
;
; Different configurations can be written to the HNC100 as compact unit.
; At present, only one configuration is possible. The following features
; are being developed.
Modular_Station      = 1 ; Modular station
Max_Module           = 32 ; Maximum number of modules
Max_Input_Len        = 32 ; Max. 32-byte inputs
Max_Output_Len       = 32 ; Max. 32-byte outputs
Max_Data_Len         = 64 ; Max. 64-byte inputs and outputs
Modul_Offset         = 0 ; Start with the first module
;
; Diagnosis
;
Unit_Diag_Bit(0)     = "EEPROM error"
Unit_Diag_Bit(1)     = "Error: cycl. data exchange"
Unit_Diag_Bit(2)     = "Can interface software error"
Unit_Diag_Bit(3)     = "Error: bus initialization"
Unit_Diag_Bit(4)     = "Wrong checksum in the EEPROM"
Unit_Diag_Bit(5)     = "Wrong hardware configuration"
Unit_Diag_Bit(6)     = "Reserved"
Unit_Diag_Bit(7)     = "Reserved"
;
Unit_Diag_Bit(8)     = "Error in the machine data"
Unit_Diag_Bit(9)     = "Error in the NC-programs"
Unit_Diag_Bit(10)    = "Address switch defective"
Unit_Diag_Bit(11)    = "Wrong memory contents"
Unit_Diag_Bit(12)    = "Hardware error"
Unit_Diag_Bit(13)    = "Division by 0"
Unit_Diag_Bit(14)    = "RAM error"
Unit_Diag_Bit(15)    = "Can interface hardware error"
;
Unit_Diag_Bit(16)    = "Reserved"
Unit_Diag_Bit(17)    = "Reserved"
Unit_Diag_Bit(18)    = "Reserved"
Unit_Diag_Bit(19)    = "Reserved"
Unit_Diag_Bit(20)    = "Reserved"
Unit_Diag_Bit(21)    = "Reserved"
Unit_Diag_Bit(22)    = "Reserved"
Unit_Diag_Bit(23)    = "Reserved"
;
Unit_Diag_Bit(24)    = "Following error Axis 1"
Unit_Diag_Bit(25)    = "Limit switch Axis 1"
Unit_Diag_Bit(26)    = "Traversing range limit + Axis 1"
Unit_Diag_Bit(27)    = "Traversing range limit - Axis 1"
Unit_Diag_Bit(28)    = "Encoder error Axis 1"
Unit_Diag_Bit(29)    = "Synchronization error"
Unit_Diag_Bit(30)    = "Reference run error Axis 1"
Unit_Diag_Bit(31)    = "Encoder jump Axis 1"
;
Unit_Diag_Bit(32)    = "Reserved"
Unit_Diag_Bit(33)    = "Reserved"
Unit_Diag_Bit(34)    = "Reserved"
Unit_Diag_Bit(35)    = "Reserved"

```

```
Unit_Diag_Bit(36) = "Reserved"
Unit_Diag_Bit(37) = "Reserved"
Unit_Diag_Bit(38) = "Reserved"
Unit_Diag_Bit(39) = "Reserved"
;
Unit_Diag_Bit(40) = "Following error Axis 2"
Unit_Diag_Bit(41) = "Limit switch Axis 2"
Unit_Diag_Bit(42) = "Traversing range limit + Axis 2"
Unit_Diag_Bit(43) = "Traversing range limit - Axis 2"
Unit_Diag_Bit(44) = "Encoder error Axis 2"
Unit_Diag_Bit(45) = "Synchronization error"
Unit_Diag_Bit(46) = "Reference run error Axis 2"
Unit_Diag_Bit(47) = "Encoder jump Axis 2"
;
Unit_Diag_Bit(48) = "Reserved"
Unit_Diag_Bit(49) = "Reserved"
Unit_Diag_Bit(50) = "Reserved"
Unit_Diag_Bit(51) = "Reserved"
Unit_Diag_Bit(52) = "Reserved"
Unit_Diag_Bit(53) = "Reserved"
Unit_Diag_Bit(54) = "Reserved"
Unit_Diag_Bit(55) = "Reserved"
;
Unit_Diag_Bit(56) = "Encoder connection Encoder 1"
Unit_Diag_Bit(57) = "Encoder connection Encoder 2"
Unit_Diag_Bit(58) = "Reserved"
Unit_Diag_Bit(59) = "Encoder connection Encoder 4"
Unit_Diag_Bit(60) = "Encoder connection Encoder 5"
Unit_Diag_Bit(61) = "Encoder connection Encoder 6"
Unit_Diag_Bit(62) = "Encoder connection Encoder 7"
Unit_Diag_Bit(63) = "Reserved"
;
Unit_Diag_Bit(64) = "Reserved"
Unit_Diag_Bit(65) = "Reserved"
Unit_Diag_Bit(66) = "Reserved"
Unit_Diag_Bit(67) = "Reserved"
Unit_Diag_Bit(68) = "Reserved"
Unit_Diag_Bit(69) = "Reserved"
Unit_Diag_Bit(70) = "Reserved"
Unit_Diag_Bit(71) = "Reserved"
;
Unit_Diag_Bit(72) = "Encoder jump Encoder 1"
Unit_Diag_Bit(73) = "Encoder jump Encoder 2"
Unit_Diag_Bit(74) = "Reserved"
Unit_Diag_Bit(75) = "Reserved"
Unit_Diag_Bit(76) = "Reserved"
Unit_Diag_Bit(77) = "Reserved"
Unit_Diag_Bit(78) = "Reserved"
Unit_Diag_Bit(79) = "Reserved"
;
; S7 <- HNC: 8 byte standard
; S7 -> HNC: 8 byte standard
; In this configuration, the 4 words are consistently transferred
; together. This configuration must be selected when an S7 system is used.
Module = "S7 <- HNC: 8 byte standard" 0xD3
EndModule
Module = "S7 -> HNC: 8 byte standard" 0xE3
EndModule
; S5 <- HNC: 8 byte standard
; S5 -> HNC: 8 byte standard
; In this configuration, the 4 words are transferred consistently individually.
; This configuration must be selected when an S5 system is used.
Module = "S5 <- HNC: 8 byte standard" 0x53
EndModule
Module = "S5 -> HNC: 8 byte standard" 0x63
EndModule
; 2 byte value <- HNC
```

```
; In this configuration a 2 byte value is fetched from the HNC. The
; contents of the 2 byte value is defined in the bustable on the HNC.
Module = "2 byte value <- HNC" 0xD0
EndModule
; 2 byte value -> HNC
; In this configuration a 2 byte value is sent to the HNC. The
; contents of the 2 byte value is defined in the bustable on the HNC.
Module = "2 byte value -> HNC" 0xE0
EndModule
; 4 byte value <- HNC
; In this configuration a 4 byte value is fetched from the HNC. The
; contents of the 4 byte value is defined in the bustable on the HNC.
Module = "4 byte value <- HNC" 0xD1
EndModule
; 4 byte value -> HNC
; In this configuration a 4 byte value is sent to the HNC. The
; contents of the 4 byte value is defined in the bustable on the HNC.
Module = "4 byte value -> HNC" 0xE1
EndModule
; 6 byte block -> HNC
; In this configuration a 6 byte block is sent to the HNC. The
; contents of the 6 byte block is defined in the bustable on the HNC.
Module = "6 byte block -> HNC" 0xE2
EndModule
```

## 12. Annex

### 12.1 Software

The drivers ( function blocks ) and files ( GSD file and type files ) that are required for the HNC 100's operation with the Profibus-DP interface are provided on a floppy disk which is included in Bosch Rexroth's scope of supply.

#### 12.1.1 Software for S5-115U (CPU: 941 - 944)

The floppy disk contains the following files:

- a:\liesmich.txt	Contents of the floppy disk
- a:\im308c\fb10\fb10past.s5d	Function block FB 10
- a:\im308c\fb21\fb21past.s5d	Function block FB 21
- a:\im308c\db200\d200past.s5d	Sample for DB 200
- a:\gsd\mmrr0476.gsd	GSD for the HNC 100
- a:\doku\siemdp2x.pdf	Description of function blocks and the Profibus-DP-interface

#### 12.1.2 Software for S5-115U (CPU: 945)

The floppy disk contains the following files:

- a:\liesmich.txt	Contents of the floppy disk
- a:\im308c\fb10\fb10p4st.s5d	Function block FB 10
- a:\im308c\fb21\fb21p4st.s5d	Function block FB 21
- a:\im308c\db200\d200p4st.s5d	Sample for DB 200
- a:\gsd\mmrr0476.gsd	GSD file for the HNC 100
- a:\doku\siemdp2x.pdf	Description of function blocks and the Profibus-DP-interface



### 12.1.3 Software for S5-115U (CPU: 928 and 948)

The floppy disk contains the following files:

- a:\liesmich.txt	Contents of the floppy disk
- a:\im308c\fb10\fb010p3st.s5d	Function block FB 10
- a:\im308c\fb21\fb021p3st.s5d	Function block FB 21
- a:\im308c\db200\db200p3st.s5d	Sample for DB 200
- a:\gsd\mmrr0476.gsd	GSD file for the HNC 100
- a:\doku\siemdp2x.pdf	Description of function blocks and the Profibus-DP-interface

### 12.1.4 Software for S5-95U with DP master interface

The floppy disk contains the following files:

- a:\liesmich.txt	Contents of the floppy disk
- a:\s595u\fb10\fb010p9st.s5d	Function block FB 10
- a:\s595u\fb21\fb021p9st.s5d	Function block FB 21
- a:\s595u\db200\db200p9st.s5d	Sample for DB 200
- a:\gsd\mmrr0476.gsd	GSD file for the HNC 100
- a:\doku\siemdp2x.pdf	Description of function blocks and the Profibus-DP-interface

### 12.1.5 Software for S7-300 and S7-400 with integrated DP master

The floppy disk contains the following files:

- a:\liesmich.txt	Contents of the floppy disk
- a:\hnc_s7	Library with functions FC 10, FC 21, FC 11 and FC 22, with examples of calling FC 10 and FC 21, with a sample structure of DB 10
- a:\gsd\mmrr0476.gsd	GSD file for the HNC 100
- a:\doku\siemdp2x.pdf	Description of function blocks and the Profibus-DP-interface

© 2003  
by Bosch Rexroth AG, Industrial Hydraulics, D-97813 Lohr am Main

All rights reserved. No part of this document may be reproduced or stored, processed, duplicated or circulated using electronic systems, in any form or by any means, without the prior written authorisation of Bosch Rexroth AG. In the event of contravention of the above provisions, the contravening party is obliged to pay compensation.

This document was prepared with the greatest of care, and all statements have been examined for correctness. This document is subject to alterations for reason of the continuing further developments of products. No liability can be accepted for any incorrect or incomplete statements.

Bosch Rexroth AG  
Industrial Hydraulics  
Zum Eisengießer 1  
97816 Lohr, Germany  
info@boschrexroth.de  
[www.boschrexroth.com/hnc100](http://www.boschrexroth.com/hnc100)