



**Handling Components**

**SERVOLINE**

**USER MANUAL  
COMMUNICATIONS PROFILE  
PROFIBUS DP**

**507805**

Edition 07/02

## Change index

### Editions issued so far:

<b>Edition</b>	<b>Comments</b>	<b>Article number (English version)</b>
02/01	First edition, valid from software version 2.00	507113
04/02	Completely reworked	507805
07/02	Kapitel 5.4.1 Aufnahme von SHA-340, SHX-340 und SHY-340 Kapitel 5.4.1 Faktor Kn korrigiert Kapitel 7.2 Änderung der Programmnamen Kapitel 7.2.4 Änderung der Programmnamen Kapitel 8.4 Immer BIT	507805

<b>1. General</b> .....	<b>1—1</b>
<b>1.1 Introduction</b> .....	<b>1—1</b>
<b>1.2 Software package</b> .....	<b>1—2</b>
<b>1.3 “Use as directed”</b> .....	<b>1—2</b>
<b>2. Overview of work required</b> .....	<b>2—3</b>
<b>2.1 PC</b> .....	<b>2—3</b>
2.1.1 Cabling.....	2—3
2.1.2 Settings.....	2—3
<b>2.2 Amplifier</b> .....	<b>2—3</b>
2.2.1 Cabling (minimal).....	2—3
2.2.2 Settings.....	2—3
<b>2.3 SPS</b> .....	<b>2—4</b>
2.3.1 Cabling.....	2—4
2.3.2 Settings.....	2—4
<b>3. Overview of the equipment profile</b> .....	<b>3—5</b>
<b>4. Service channel PKW</b> .....	<b>4—6</b>
<b>4.1 Parameter ID PKE in PKW</b> .....	<b>4—6</b>
4.1.1 Task- / Response ID AK in PKE.....	4—7
4.1.2 Significance of the toggle bit for spontaneous message SPM in PKE (Master ↔ Slave).....	4—8
4.1.3 Significance of the parameter number PNU in PKE (Master ↔ Slave).....	4—8
<b>4.2 Sub-index IND</b> .....	<b>4—10</b>
<b>4.3 Parameter value PWE</b> .....	<b>4—10</b>
4.3.1 Error numbers in PWE byte 8 with response identification AK= 7 in PKE.....	4—11
4.3.2 Parameter values PWE with associated PNU numbers (in decimal).....	4—12
<b>4.4 Using the parameter channel</b> .....	<b>4—19</b>
4.4.1 Writing/reading an amplifier parameter.....	4—19
<b>4.5 Example telegram in PKW</b> .....	<b>4—20</b>
4.5.1 Setting the acceleration.....	4—20
<b>5. Process data channel PZD</b> .....	<b>5—21</b>
<b>5.1 Process data PZD1</b> .....	<b>5—21</b>
5.1.1 The control word (STW).....	5—22
5.1.2 The status word (ZSW).....	5—23
<b>5.2 Servo amplifier control</b> .....	<b>5—24</b>
5.2.1 States of the state machine.....	5—24
5.2.2 Transitions of the state machine.....	5—25
<b>5.3 Process data PZD2 to PZD6</b> .....	<b>5—26</b>
<b>5.4 Using the process data channel</b> .....	<b>5—27</b>
5.4.1 Standard allocation of the process data compartments.....	5—27
5.4.2 Deviating allocation of the process data compartments.....	5—28
<b>5.5 Example telegram in PZD</b> .....	<b>5—28</b>
5.5.1 Starting the home run (referencing).....	5—28
5.5.2 Starting a saved motion task.....	5—29
5.5.3 Starting a direct motion task.....	5—30
<b>6. Steps necessary to perform motion tasks</b> .....	<b>6—31</b>
<b>6.1 Setting the operating mode</b> .....	<b>6—31</b>
6.1.1 Locking the setpoint values and process data.....	6—31
6.1.2 Resetting PKW.....	6—32

6.1.3	Selecting the new operating mode with PNU 930 .....	6—32
6.1.4	Releasing the setpoint values .....	6—33
6.1.5	Error .....	6—33
<b>6.2</b>	<b>Home run (referencing) .....</b>	<b>6—34</b>
6.2.1	Starting the home run (referencing) .....	6—34
<b>6.3</b>	<b>Starting a direct motion task with saved next motion task .....</b>	<b>6—35</b>
6.3.1	Number of the motion task (only the saved motion task) .....	6—35
6.3.2	Starting a direct motion task .....	6—35
<b>7.</b>	<b>Program description for Siemens S7 .....</b>	<b>7—36</b>
7.1	Hardware configuration of the CPU .....	7—36
7.2	Program overview .....	7—37
7.2.1	Generally .....	7—37
7.2.2	Description of the modules .....	7—37
7.2.3	Program structure .....	7—38
7.2.4	Programs 1DEMO_D (E), 2DEMOi_D (E) and 3DEMOi_D (E) .....	7—38
<b>8.</b>	<b>Appendix .....</b>	<b>8—46</b>
8.1	Abbreviations and acronyms .....	8—46
8.2	Glossary .....	8—47
8.3	Number systems .....	8—48
8.4	Data types .....	8—48

# 1. General

## 1.1 Introduction

This manual deals with the range of functions and the software protocol of the PROFIBUS DP expansion card for the SERVOSTAR™ 600. Information is arranged as follows:

- Chapter 1: General information
- Chapter 2: Overview of work required
- Chapter 3: Overview of the equipment profile
- Chapter 4: Service channel PKW
- Chapter 5: Process data channel PZD
- Chapter 6: Steps necessary to perform motion tasks
- Chapter 7: Program description for Siemens S7
- Chapter 8: Appendix

This manual forms part of the complete documentation of the SERVOLINE® products. The complete documentation consists of the following documents:

- Communications profile PROFIBUS DP
- Setup software (version Profibus DP)
- Servo amplifier (version Profibus DP)
- Mechanical design of the SERVOLINE® products

The documentation is available in the following language versions: German, English, French, Italian, Spanish and Swedish. Please refer to the spare parts list for the respective order number.



- **Please read this documentation before carrying out installation and wiring.** Incorrect handling of the servo amplifier can lead to personal injury or material damage and render the warranty void. It is vital that you keep to the technical data and information on connection requirements.
- **Only properly qualified SPS programmers who are familiar with PROFIBUS DP are permitted to perform programming tasks.**
- Only properly qualified personnel are permitted to perform activities such as installation, commissioning and maintenance. Properly qualified persons are those who are familiar with the assembly, installation, commissioning and operation of the product, and who have the appropriate qualifications for their job. The qualified personnel must know and observe the following standards or guidelines:

IEC 364 and CENELEC HD 384 or DIN VDE 0100  
IEC report 664 or DIN VDE 0110  
National accident prevention regulations or VBG4

## 1.2 Software package

- MONTECH setup software
- GSD library file
- Programs for Siemens S7 for controlling up to three axes (servo cartesian system)

## 1.3 “Use as directed”

The communications profile is used to change and store the operation parameters of the servo amplifier of the SERVOLINE® products. The connected servo amplifier is operated using the communications profile.

The equipment manufacturer is obliged to prepare a hazard analysis of the equipment, and is also responsible for functional, mechanical and personal safety of the equipment. Servo amplifiers are installed as components in electrical equipment or machinery. They must only be operated as integrated components of the equipment.

The BTB contact must be wired into the safety loop of the electrical equipment. The safety loop as well as the stop and emergency stop functions must comply with the requirements of EN60204, EN292 and VDI2853.

Before operating the servo axes, all work is to be carried out as specified in the operator manuals of the servo amplifier and the Servoline axes. Observe all safety regulations.



- **Assembly, installation, wiring and final check according to the operator manual of the servo amplifier have been carried out.**

## 2. Overview of work required

This chapter provides a step-by-step description of all the stages required for initial operation. Please refer to the following chapters for detailed information.

Reference documents:

User manual Montech Servo Amplifier (PROFIBUS DP) hereinafter referred to as **BMS**.

User manual Montech Setup Software (Version PROFIBUS) hereinafter referred to as **BMIS**.

### 2.1 PC

#### 2.1.1 Cabling

- 1 Cable for communication between the servo amplifier and the PC. See also BMS Chapter 4.8.
- 1 Cable for communication between the SPS and the PC.

#### 2.1.2 Settings

- Load the supplied Montech software 2.0 (CD-ROM) on the PC.
- Load the Siemens S7 software.

## 2.2 Amplifier

### 2.2.1 Cabling (minimal)

- Connect the reference switch. Wire the "black" cable of the reference switch to connector X3 terminal 11 to 14 (select one). See also BMS Chapter 4.2, and BMIS Chapter 2.8.
- Connect the mains connection (3 PH 400 VAC). See also BMS Chapter 4.3.1.
- Connect the 24VDC auxiliary connection. See also BMS Chapter 4.3.2.
- Connect the supplied motor cables. See also BMS Chapter 4.4.
- Connect the supplied resolver or encoder cable. See also BMS Chapter 4.5.1 and 2.
- Connect the enable signal. See also BMS Chapter 4.6.1.1.
- Wire DGND connector X3 terminal 18 to the auxiliary voltage GND. See also BMS Chapter 4.6.1.
- Profibus cable between the servo amplifier and the SPS. See also BMS Chapter 4.6.3.

This is the minimum cabling required to operate an axis.

### 2.2.2 Settings

Servo amplifier settings are entered via Montech Software 2.0.

- Enter positioning data if you work with stored motion tasks. See also BMIS Chapter 2.4. For limit values of positioning data refer to BMIS Chapter 2.16.2.
- Set the reference switch to the wired input. See also BMIS Chapter 2.8.
- Enter the field bus address of the servo amplifier in the display page <<Settings>>, see BMIS Chapter 2.17.1.

## 2.3 SPS

### 2.3.1 Cabling

- Connect the mains supply (3 PH 240 VAC) to the SPS.

### 2.3.2 Settings

- Hardware configuration. See Chapter 7.1.
- Generate an application program. 3 demonstration programs are available, see Chapter 7.2.

Essentially, the following three steps are to be carried out with the program:

- **Setting the operating mode**  
After switching on the auxiliary voltage 24 VDC or after a RESET of the servo amplifier.
- **The homing (reference) traverse**  
After switching on the auxiliary voltage 24 VDC.
- **The motion task**

It is absolutely essential that the above steps be carried out in the order indicated.



### 3. Overview of the equipment profile

#### Type 2: Octet-String 20

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----

BYTE																			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
PKW								PZD											
PKE		IND		PWE				STW ZSW PZD1		HSW HIW PZD2		PZD3		PZD4		PZD5		PZD6	

Abbreviations: PKW      Parameter ID value  
 PKE      Parameter ID  
 IND      Sub-index  
 PWE      Parameter value

PZD      Process data  
 PZD1-6      Process data  
 STW      Control word  
 ZSW      Status word  
 HSW      Main setpoint  
 HIW      Main actual value

**Servo amplifier only uses the PPO type 2 with 4 words PKW section and 6 words PZD section (according to PROFIBUS profile PROFIDRIVE).**

The instrument profile can be divided into two sections or data channels:

1. PKW section (4 words)
2. PZD section (6 words)

#### PKW (Service channel)

The PKW data channel can also be called a service channel. The service channel only uses confirmed communication services; it is used by servo amplifier as a parameter channel. **This channel has no real-time capability.**

This channel is primarily used for settings and status inquiries in connection with the servo amplifier, or for transmitting parameters.

#### PZD (process data channel)

The PZD data channel can also be called a process data channel. The process data channel uses unconfirmed communication services. The response of the servo amplifier to an unconfirmed service can only be seen in the reaction of the servo amplifier (status word, actual values).

**This channel has real-time capability.** - A.4.016.3/3

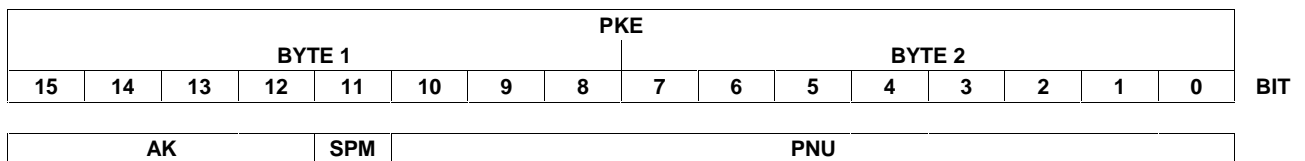
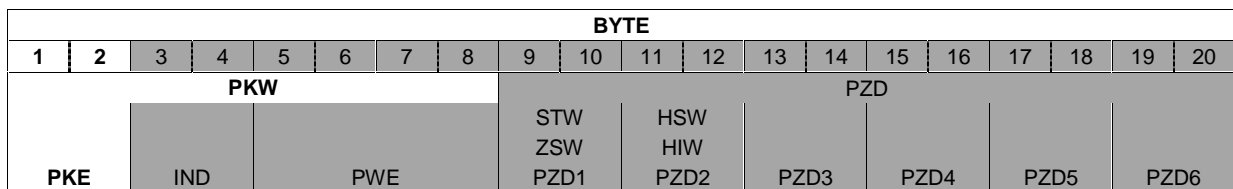
This channel is used for the transmission of data during the home run, during jogging and during motion tasks.

## 4. Service channel PKW

This channel can be divided into three segments:

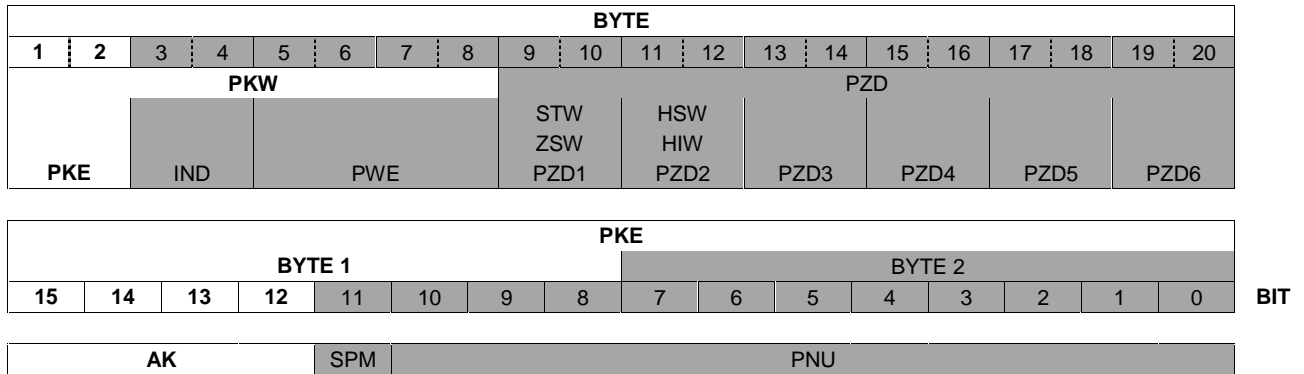
- PKE:** Parameter identification  
This is where the type of execution AK (parameter read or write) and the identification number PNU (parameter number) are transmitted to the servo amplifier.
- IND:** Subindex  
An additional identification number for PNU greater than 1600 is possible here.
- PWE:** Parameter value  
The value of the corresponding PNU is indicated here.

### 4.1 Parameter ID PKE in PKW



- Abbreviations:
- AK Task / response ID
  - SPM Toggle-bit for spontaneous message (not implemented at present)
  - PNU Parameter number

### 4.1.1 Task- / Response ID AK in PKE



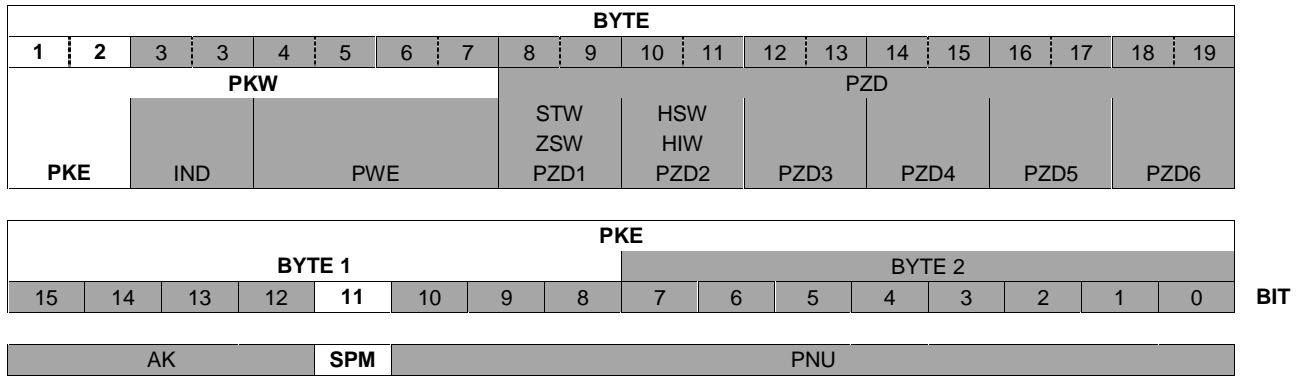
#### Significance of the task identification AK in PKE (Master → Slave)

Master → Slave	
Task ID in decimal	Function
0	No task
1	Request parameter value
2	Not valid for servo amplifier
3	Alter parameter value [DW]
4 – 9	Not valid for servo amplifier
10 – 15	Reserved

#### Significance of the response IDs AK in PKE (Slave → Master)

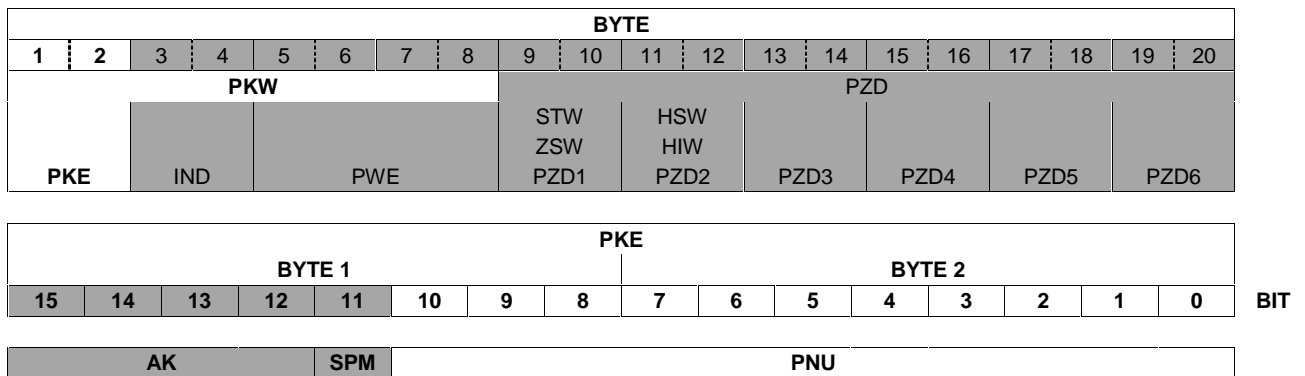
Slave → Master	
Response IDs in decimal	Significance
0	No task
1	Not valid for servo amplifier
2	Transmit parameter value
3 – 6	Not valid for servo amplifier
7	<b>Task not possible (with error number)</b> The error number is indicated in PWE. See Chapter 4.1.3
8 – 12	Not valid for servo amplifier

### 4.1.2 Significance of the toggle bit for spontaneous message SPM in PKE (Master ↔ Slave)



Not implemented at present. **SPM is always on 0.**

### 4.1.3 Significance of the parameter number PNU in PKE (Master ↔ Slave)



## List of parameter numbers PNU

PNU		Data type	Access no. (hex)		Description
Dec	Hex		Write AK=3	Read AK=1	
<b>Profile parameter</b>					
904	388	UINT32	---	1388	Number of the supported PPO-Write, always 2
911	38F	UINT32	---	138F	Number of the supported PPO-Read, always 2
918	396	UINT32	---	1396	Participant address on PROFIBUS
930	3A2	UINT32	33A2	13A2	Selector for operating mode
963	3C3	UINT32	---	13C3	PROFIBUS baud rate
965	3C5	Octet-String2	---	13C5	Number of the PROFIDRIVE profile (0302H)
971	3CB	UINT32	33CB	---	Non-volatile parameter save
<b>Manufacturer-specific parameters SERVOSTAR™</b>					
<b>General parameters</b>					
1000	3E8	Visible String4	---	13E8	Instrument ID
1001	3E9	UINT32	---	13E9	Manufacturer-specific error register
1002	3EA	UINT32	---	13EA	Manufacturer-specific status register
<b>Position controller parameters</b>					
1250	4E2	UINT32	34E2	14E2	Multiplier for speeds, jogging, homing/ref.
1252	4E4	INTEGER32	34E4	14E4	In-position window
1253	4E5	INTEGER32	34E5	14E5	Following-error window
1254	4E6	INTEGER32	34E6	14E6	Position register 1
1255	4E7	INTEGER32	34E7	14E7	Position register 2
1256	4E8	INTEGER32	34E8	14E8	Position register 3
1257	4E9	INTEGER32	34E9	14E9	Position register 4
1265	4F1	UINT32	34F1	14F1	Maximum velocity for positioning mode
<b>Position data for the position control mode</b>					
1300	514	INTEGER32	3514	1514	Position
1301	515	INTEGER16	3515	1515	Velocity
1302	516	UINT32	3516	1516	Motion task type
1304	518	UINT32	3518	1518	Starting time (acceleration)
1305	519	UINT32	3519	1519	Braking time (deceleration)
1306	51A	UINT32	351A	151A	Jolt limiting (acceleration)
1307	51B	UINT32	351B	151B	Jolt limiting (deceleration)
1308	51C	UINT32	351C	151C	Number of the next motion task
1309	51D	UINT32	351D	151D	Start delay for next motion task
1310	51E	2 * UINT16	251E	---	Copy a motion task
<b>Position set-up mode</b>					
1354	54A	UINT32	354A	154A	Reference offset
1356	54C	UINT32	---	154C	Jogging velocity
<b>Actual values</b>					
1403	57B	INTEGER32	---	157B	SI - position, actual value
1404	57C	INTEGER32	---	157C	SI - velocity, actual value
1405	57D	INTEGER32	---	157D	SI - contouring error
1406	57E	INTEGER32	---	157E	RMS current
1407	57F	INTEGER32	---	157F	SI - speed, actual value
1408	580	INTEGER32	---	1580	Heat sink temperature
1409	581	INTEGER32	---	1581	Internal temperature
1410	582	INTEGER32	---	1582	DC-bus (DC-link) voltage
1411	583	INTEGER32	---	1583	Ballast power
1412	584	INTEGER32	---	1584	I <sup>2</sup> t - loading
1413	585	INTEGER32	---	1585	Running time
1462	5B6	UINT32	---	15B6	State of the digital I/Os

See Chapter 4.3 for a detailed description of the PNU.

### 4.2 Sub-index IND

BYTE																			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
PKW								PZD											
PKE		IND		PWE				STW ZSW PZD1		HSW HIW PZD2		PZD3		PZD4		PZD5		PZD6	

IND															BIT	
BYTE 3							BYTE 4									
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0							0 Reserved									

For PNU numbers smaller than 1600 the subindex is **always set to 0**.

### 4.3 Parameter value PWE

BYTE																			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
PKW								PZD											
PKE		IND		PWE				STW ZSW PZD1		HSW HIW PZD2		PZD3		PZD4		PZD5		PZD6	

PWE																																
BYTE 5								BYTE 6								BYTE 7								BYTE 8								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	BIT
MSW								MSB								LSB																

The value of the PNU variable is in PWE; it is filed so as to be right-aligned (start: byte 8):

Length:      4-byte data (double word)                      PWE 5-8 (PWE 8 LSB)

### 4.3.1 Error numbers in PWE byte 8 with response identification AK= 7 in PKE

Commands are transferred with task ID 3 (AK in PKE). If a command cannot be executed, the error is signalled by response identification AK = 7, and an error number in PWE Byte 8 is issued.

<b>Error no. (Decimal)</b>	<b>Description</b>
<b>0</b>	<b>Illegal PNU</b>
<b>1</b>	<b>Parameter value cannot be altered</b>
<b>2</b>	<b>Lower or upper limit exceeded</b>
<b>3</b>	Not used
<b>4</b>	<b>No array</b>
<b>5</b>	<b>Incorrect data type</b>
<b>6</b>	<b>Setting not allowed (can only be reset)</b>
7 - 16	Not valid for servo amplifier
<b>17</b>	<b>Task cannot be executed due to operating status</b>
<b>18</b>	<b>Other error</b>
<b>19-100</b>	<b>Reserved</b>
<b>101</b>	<b>Faulty task ID</b>
<b>102</b>	<b>Software error (command table)</b>
<b>103</b>	<b>Only possible in disabled state</b>
<b>104</b>	<b>Only possible in enabled state</b>
<b>105</b>	<b>BCC error in the EEPROM data</b>
<b>106</b>	<b>Only possible after motion task has been stopped</b>
<b>107</b>	<b>Incorrect value [16, 20]</b>
<b>108</b>	<b>Incorrect parameter (OCOPY x [- y] z)</b>
<b>109</b>	<b>Incorrect motion block no (0, 1..180, 192..255)</b>
<b>110</b>	<b>Incorrect parameter (PTEACH x [y])</b>
<b>111</b>	<b>EEPROM write error</b>
<b>112</b>	<b>Incorrect value</b>
<b>113</b>	<b>BCC error in the motion block</b>
<b>&gt;113</b>	<b>Reserved</b>

### 4.3.2 Parameter values PWE with associated PNU numbers (in decimal)

<b>PNU 904</b>	<b>Description</b>
	<b>Number of the PPO Write supported</b> Number of the supported PPO type. Since exclusively PPO type 2 is supported (see Chapter 3), this parameter is always set to 2.
<b>PNU 911</b>	<b>Description</b>
	<b>Number of the PPO Write supported</b> Number of the supported PPO type. Since exclusively PPO type 2 is supported (see Chapter 3), this parameter is always set to 2.

<b>PNU 918</b>	<b>Description</b>
	<b>Participant address on PROFIBUS</b>

<b>PNU 930</b>	<b>Description</b>
	<b>Setting the operating mode</b> <b>Always set the operating mode to 2! This is the positioning mode according to the PROFIDRIVE profile.</b> After it has been switched on, the servo amplifier is in operating mode –126. While in this state, the state machine can be controlled, no motion functions can be initiated.



**Caution**  
*In the safe operating mode (-126) no motion functions can be initiated via the PROFIBUS. However, it is possible to perform motion functions by means of the setup software. If the operating mode is changed, then motion functions can only be operated via the PROFIBUS. If the operating mode is changed via another communication channel, then the emergency brake is applied to the drive, and the error F21 (handling error, plug-in card) is signalled.*

<b>PNU 963</b>	<b>Description</b>
	<b>PROFIBUS baud rate</b> See description below

This parameter shows the index of the baud rate used for PROFIBUS communication; it is read-only. The baud rate is defined by the PROFIBUS master (CPU).

The following table shows the allocation of numbers to the individual baud rates (in Kbaud):

<b>Index</b>	0	1	2	3	4	5	6	7	8	9
<b>Baud rate</b>	12000	6000	3000	1500	500	187.5	93.75	45.45	19.2	9.6

<b>PNU 965</b>	<b>Description</b>
	<b>Number of the PROFIDRIVE profile (0302H)</b> The number of the PROFIDRIVE profile can be read out via this parameter. Profile number 3, version 2 is used.

<b>PNU 971</b>	<b>Description</b>
	<b>Non-volatile parameter save</b> With this parameter you can save all parameter settings in the EEPROM (non-volatile save). To do so, the value of the parameter must be PWE = 1 during transmission



PNU 1000	Description
	The instrument ID consists of four ASCII characters together with the characters "S601".T

PNU 1001	Description
	<b>Manufacturer-specific error register</b> The assignment of the error register is shown in the following table.

Bit	System-behaviour	Number	Description	Explanation
0		F01	Heat sink temperature	Heat sink temperature too high
1		F02	Overvoltage	Overvoltage in the DC-bus (DC-link)
2		F03	Contouring error	Message from the position controller
3	Cold-start	F04	Feedback	Cable break, short circuit, short circuit to earth
4		F05	Undervoltage	Undervoltage in the DC-link (limit value 100V)
5	Cold-start	F06	Motor temperature	Temperature sensor defective or motor temp. too high
6	Cold-start	F07	Auxiliary voltage	Internal auxiliary voltage not OK
7		F08	Overspeed	Motor races; speed excessive
8	Cold-start	F09	EEPROM	Checksum error
9	Cold-start	F10	Flash-EPROM	Checksum error
10	Cold-start	F11	Brake	Cable break, short circuit, short circuit to earth (SVA only)
11	Cold-start	F12	Motor phase	Motor phase missing (broken cable or similar)
12		F13	Internal temperature	Amplifier internal temperature too high
13	Cold-start	F14	Output stage	Error in the output stage
14		F15	I <sup>2</sup> t max.	I <sup>2</sup> t-maximum value exceeded
15		F16	Mains supply-BTB	2 or 3 mains supply phases are missing
16	Cold-start	F17	A/D-converter	Error in the analog/digital conversion
17	Cold-start	F18	Ballast	Ballast circuit defective or setting not OK
18		F19	Mains supply phase	One mains supply phase is missing
19	Cold-start	F20	Slot error	Hardware error of the expansion card (PROFIBUS)
20	Cold-start	F21	Handling error	Software error of the expansion card (PROFIBUS)
21	Cold-start	F22	Short circuit to earth	Inactive
22	Cold-start	F23	CAN Bus inactive	Communication interruption, CAN Bus
23-30	Cold-start	F24-F31	Reserved	Reserved
31	Cold-start	F32	System error	System software does not react properly

When the cause of the error has been cleared, the error state can be cancelled by setting bit 7 in the control word.

In the case of a reset, the error response of the servo amplifier will differ, depending on the error that has occurred:

- For errors marked «Cold start», setting the reset bit initiates a cold start of the drive, whereby the PROFIBUS communication to the servo amplifier will also be interrupted for several seconds. This break in communication may have to be handled separately by the PLC program.
- In the case of all other error messages, a reset leads to a warm start during which communication will not be interrupted.

PNU 1002	Description
	<b>Manufacturer-specific status register</b> The assignment of the bits for the status register is shown in the following table:

### PWE Byte 5 and 6

Bit	Description	
0	Warning 1:	I <sup>2</sup> t threshold exceeded (is set as long as I rms is above the threshold)
1	Warning 2:	Ballast power exceeded (is set as long as the set ballast power is exceeded)
2	Warning 3:	Contouring error (is set as soon as the difference between the actual position and the position-control trajectory exceeds the set value for PEMAX. This is cancelled by the command CLRFAULT or by activating the function "Cancel error/contouring error".)
3	Warning 4:	Threshold monitoring active (is set when the threshold monitoring time is exceeded. This is cancelled by the command CLRFAULT or by activating the function "Cancel error/contouring error".)
4	Warning 5:	Mains supply phase missing (is set as long as the three mains supply phases cannot be detected). By default, monitoring for mains phase failure is deactivated (Emergency-Off function).
5	Warning 6:	Software limit-switch 1 has been activated (is set as soon as the position goes below the set position of software limit-switch 1; or if a motion task is started with a target position that is below SWE1 (at the same time, bit 8 "Faulty motion task started" is set). This is cancelled when the position SWE1 is exceeded and a positive speed/velocity setpoint is applied, or if a motion task is started with a target position within the valid range of movement).
6	Warning 7:	Software limit switch 2 has been activated (see above).
7	Warning 8:	Faulty motion task has been started (is set if an attempt is made to start a non-existing motion task). This is cancelled when a valid motion task is started.
8	Warning 9:	No reference point was set at the start of the motion task (is set if a motion block is started without a previous homing run). This is cancelled after completion of a homing run.
9	Warning 10:	PSTOP active (is set as long as the hardware limit-switch PSTOP is active). In the default setting, PSTOP is not used.
10	Warning 11:	NSTOP active (is set as long as the hardware limit-switch NSTOP is active). In the default setting, NSTOP is not used.
11	Warning 12:	Motor default values were loaded (is set at switch-on of the amplifier if the motor number from the serial EEPROM and the motor number from the SINCOS encoder are different). If a valid motor number is entered, and the data is stored in the encoder and in the internal EEPROM, the warning will no longer be issued from the next switch-on of the encoder. Is not used in the default setting.
12	Warning 13:	Reserved
13	Warning 14:	Reserve
14	Warning 15:	Reserve
15	Warning 16:	Reserve

Warnings 3 and 4 can be reset via bit 13 in the control word.

## PWE Byte 7 and 8

Bit	Description
16	<b>Motion task active</b> is set as long as a position control task is active (motion task, jogging, homing)
17	<b>Reference point set</b> is set after a homing run; is cancelled when the amplifier is switched on or when a homing run is started
18	<b>Actual position = home position</b> is set as long as the reference switch is activated
19	<b>InPosition</b> is set as long as the difference between the target position for a motion task and the actual position is smaller than PNU 1252. The InPosition signal is suppressed if a following task is started at the target position.
20	—
21	—
22	<b>Software limit switch 1</b> has been triggered. See User Manual Setup Software (Version PROFIBUS) Chapter 2.16.1
23	<b>Software limit switch 2</b> has been triggered See User Manual Setup Software (Version PROFIBUS) Chapter 2.16.1
24	<b>Position index 1</b> See User Manual Setup Software (Version PROFIBUS) Chapter 2.5
25	<b>Position index 2</b> See User Manual Setup Software (Version PROFIBUS) Chapter 2.5
26	<b>Initialisation completed</b> is set if the internal initialisation of the amplifier has been completed
27	—
28	—
29	<b>Safety relay has been triggered</b> is set as long as the safety relay is open – (AS option)
30	<b>Output stage enabled</b> is set when software and hardware enabling are set
31	<b>Error present</b> is cancelled when the amplifier is switched on, or if the function “Cancel error” is called

In the process data (PZD5 Input), bits 16 to 31 of the manufacturer-specific status registers are issued. Warnings 3 and 4 can be reset through bit 13 in the control word.

PNU 1250	Description
	<p><b>Velocity multiplier</b> This parameter is used to enter a multiplier for the jogging/homing velocity. The velocity for jogging/homing is given through PZD2 in the control word when jogging/homing is started. The actual jogging velocity is calculated according to the following formula:</p> $V_{\text{tipp,ges.}}(32\text{bit}) = V_{\text{tipp,PZD2}}(16\text{bit}) \times \text{multiplier}(16\text{bit})$

PNU 1300	Description
	<p><b>Position</b> Position values are specified in <math>\mu\text{m}</math>. Condition: bit13 = 1 (PNU 1302)</p>

PNU 1301	Description
	<p><b>Velocity</b> Velocity values are specified in <math>\mu\text{m/s}</math>. Condition: bit13 = 1 (PNU 1302)</p>

PNU 1302	Description
	<p><b>Type of motion task</b> See table below and PZD6 output Chapter 5.4.2</p>

Kat.	Bit	Value	Significance	
Type of motion task (ABS or rel.)	0	0	The position value that is given, is evaluated as an absolute position.	
		1	The position value that is given, is evaluated as a relative traversing distance. The two following bits then determine the type of relative motion.	
	1	0	If bits 1 and 2 are set to 0 and bit 0 is set to 1, then the relative motion task is performed according to the "InPosition" bit.	
		1	The new target position is given by the old target position plus the traversing distance. Bit 1 has priority over bit 2.	
	2	0	If bits 1 and 2 are set to 0, and bit 0 is set to 1, then the relative motion task is performed according to the "InPosition" bit.	
		1	The new target position is given by the actual position plus the traversing distance.	
Next motion task	3	0	No following task available	
		1	There is a following task but it must be defined through PNU 1308	
	4	0	Change over to next motion task, with braking to 0 at the target position.	
		1	Change over to next motion task, without standstill at the target position. The type of velocity transition is determined by bit 8.	
	5	0	Change over to next motion task without evaluating inputs.	
		1	A following motion task is started by a correspondingly configured input.	
	6	0	Start the next motion task via Input state = Low.	
		1	Start the next motion task via Input state = High or, if bit 7 = 1, in any case after the delay time set via PNU 1309.	
	7	0	The next motion task starts immediately.	
		1	The next motion task starts after the delay time set via PNU 1309 or, if bit 6 = 1, prior to that, in response to a respective input signal.	
	8	0	Only for following motion tasks and bit 4 = 1; from the target position for the previous motion task onwards, the velocity is altered to the value for the following motion task.	
		1	The setpoint speed of the next motion task should already be reached in the target position. Only sensible if no change in direction occurs.	
	-	9	-	Reserved
	-	10		
-	11			
Acc. Dec.	12	0	Accelerations are taken from the data record; alternatively the limit values of accelerations are used. The limit values are specified in the user manual "Setup Software (Version PROFIBUS)" Chapter 2.16.2.	
		1	Accelerations are specified via PNU 1304 and / or PNU 1305.	
-	13	1	Target position and target speed are entered in SI units.	
-	14	0	The programmed speed is used for the motion task speed.	
-	15	-	Reserved	

PNU 1304	Description
	<p><b>Acceleration time</b> This parameter defines acceleration time of the motion task, specified in <math>\text{mm/s}^2</math>. <b>Prerequisite:</b> Bit 12=1 in PNU1302</p>

<b>PNU 1305</b>	<b>Description</b>
	<b>Deceleration</b> Via this parameter, the delay of the motion task is indicated in <b>mm/s<sup>2</sup></b> . <b>Prerequisite:</b> Bit 12=1 in PNU1302
<b>PNU 1306</b>	<b>Description</b>
	<b>Acceleration, jolt limiting</b> This parameter defines the form of the acceleration ramp. Generally, linear axes need to be operated via the sin <sup>2</sup> -ramp. The sin <sup>2</sup> -ramp is used if the input value is ½ of the acceleration time.
<b>PNU 1307</b>	<b>Description</b>
	<b>Deceleration, jolt limiting</b> This parameter defines the form of the deceleration ramp. Generally, linear axes need to be operated via the sin <sup>2</sup> -ramp. The sin <sup>2</sup> -ramp is used if the input value is ½ of the acceleration time.
<b>PNU 1308</b>	<b>Description</b>
	<b>Number of the saved next motion task</b> The motion block no. of the motion task to be started can range from 1 to 180 (EEPROM motion tasks) or from 192 to 255 (RAM motion tasks). <b>Prerequisite:</b> bit 3=1 in PNU1302
<b>PNU 1309</b>	<b>Description</b>
	<b>Start delay for the next motion task</b> This parameter is used to set a delay time, specified in <b>ms</b> , before the start of a motion task. <b>Prerequisite:</b> bit 6=1 and / or bit 7=1 in PNU1302
<b>PNU 1310</b>	<b>Description</b>
	<b>Copy a motion task</b> This parameter can be used to copy motion tasks. The source motion task must be entered. The MSW of PWE (bytes 5 & 6) and the target motion task in LSW of PWE (bytes 7 & 8) must be entered.
<b>PNU 1403</b>	<b>Description</b>
	<b>Position, actual value</b> The parameter value is the actual position value in <b>µm</b> .
<b>PNU 1404</b>	<b>Description</b>
	<b>Velocity, actual value</b> The parameter value is the actual velocity value in <b>µm/s</b> .
<b>PNU 1405</b>	<b>Description</b>
	<b>Contouring error</b> The parameter value is the actual contouring error in <b>µm</b> .
<b>PNU 1406</b>	<b>Description</b>
	<b>Active current</b> The parameter value is the actual active current in <b>mA</b> .

<b>PNU 1407</b>	<b>Description</b>
	<b>SI-speed, actual value</b> The parameter value is the actual speed value in <b>min<sup>-1</sup></b> .
<b>PNU 1408</b>	<b>Description</b>
	<b>Heat sink temperature</b> The parameter value is the actual heat sink temperature in <b>°C</b> .
<b>PNU 1409</b>	<b>Description</b>
	<b>Internal temperature</b> The parameter value is the actual internal temperature in <b>°C</b> .
<b>PNU 1410</b>	<b>Description</b>
	<b>Intermediate circuit voltage</b> The parameter value is the actual intermediate circuit voltage in <b>V</b> .
<b>PNU 1411</b>	<b>Description</b>
	<b>Ballast power</b> The parameter value is the actual ballast power in <b>W</b> .
<b>PNU 1412</b>	<b>Description</b>
	<b>I<sup>2</sup>t- load</b> The parameter value is the actual loading in
<b>PNU 1413</b>	<b>Description</b>
	<b>Running time</b> The parameter value is the actual running time in <b>Min</b> .
<b>PNU 1462</b>	<b>Description</b>
	<b>State of the digital I/Os</b> Bit 0 corresponds to DIGITAL-OUT2 Bit 1 corresponds to DIGITAL-OUT1 Bit 2 corresponds to ENABLE Bit 3 corresponds to NSTOP Bit 4 corresponds to PSTOP Bit 5 corresponds to DIGITAL-IN2 Bit 6 corresponds to DIGITAL-IN1

## 4.4 Using the parameter channel

The controller can be parameterised via the operator software or via the PROFIBUS.

### 4.4.1 Writing/reading an amplifier parameter

The amplifier parameter (PKW) contains 3 parts: PKE, IND and PWE

- **PKE** again contains 3 parts:
  - AK: determines whether the parameter is written to the amplifier (AK=3) or read from the amplifier (AK=1).
  - SPM: not implemented at present; it is always set to 0.
  - PNU: the parameter number indicates the "identification" of the amplifier parameter.
- **IND** is not relevant; it is always set to 0.
- **PWE** contains the desired value.

With the command AK write (AK=3) or read (AK=1), the amplifier parameter is transferred to the volatile memory of the servo amplifier. With PNU 971 this amplifier parameter can also be transferred to the nonvolatile memory.

#### Telegram structure:

BYTE																				
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
PKW								PZD												
PKE			IND			PWE			STW ZSW PZD1		HSW HIW PZD2		PZD3		PZD4		PZD5		PZD6	

Telegram Send (from the SPS to the servo amplifier)	Telegram Reply (from the servo amplifier to the SPS)
<b>PKE</b> PKE is made up from AK & SPM & PNU: AK = 3 (write) or AK=1 (read) SPM= 0 PNU= 904 to 1462 (see Chapter 4.1.3)	<b>PKE</b> PKE is made up from AK & SPM & PNU: AK= 2 (transmission OK) or AK= 7 (error) SPM= 0 PNU as sent
<b>IND=0</b> <b>PWE</b> With AK = 3, the desired value is stated here. For significance see Chapter 4.3.2 With AK = 1, no entry or "0"	<b>IND=0</b> <b>PWE</b> With AK = 2, as sent With AK = 7, the error number is stated here. See Chapter 4.3.1

## 4.5 Example telegram in PKW

### 4.5.1 Setting the acceleration

If acceleration other than the limit value acceleration (maximum permissible acceleration, see Montech operating instructions "Setup Software (Version PROFIBUS) Chapter 2.16.2" is desired, it can be conveyed to the servo amplifier via the PKW channel.

Prerequisite: PNU 1302 bit 12= 1 so that the servo amplifier carries out the motion task with the following acceleration.

Parameter number (PNU): 1304 dec = 101 00011000 bin

Parameter value (acceleration in mm/s<sup>2</sup>): 9000 dec = 100011 00101000 bin

The parameterising telegram then has the following appearance:

PKW								BYTE BIT
1	2	3	4	5	6	7	8	
15..12 11 10...8	7.....0	15.....8	7.....0	15.....8	7.....0	15.....8	7.....0	
0011 0 101	00011000	00000000	00000000	00000000	00000000	00100011	00101000	
PKE		IND		PWE				

PKE is made up of:

PNU bit 0 to 10:1304 dec = 101 00011000 bin

SPM bit 11: 0

AK bit 12 to 15: 3 dec (write parameter value) = 11 bin

This results in PKE = AK & SPM & PNU:

0011 & 0 & 101 00011000 or 00110101 00011000 which are entered in bytes 1 and 2.

IND is always 0 for PNU less than 1600.

PWE is the area where the value is entered, with the value always being entered so as to be right aligned or starting with byte 8.

The servo amplifier sends a reply telegram with AK=2 and identical values for PNU and PWE.

The reply telegram then has the following appearance:

PKW								BYTE
1	2	3	4	5	6	7	8	
0010 0 101	00011000	00000000	00000000	00000000	00000000	00100011	00101000	
PKE		IND		PWE				

A negative reply from the servo amplifier (AK= 7) has the following appearance:

PKW								BYTE
1	2	3	4	5	6	7	8	
0111 0 101	00011000	00000000	00000000	00000000	00000000	00000000	00010001	
PKE		IND		PWE				

PWE now contains the error number which has been described in detail in Chapter 4.3.1. In the present case this is error number 17: "Task cannot be executed due to operating status".



## 5. Process data channel PZD

BYTE																				
1	2	3	4	4	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
PKW								PZD												
PKE		IND		PWE				STW	ZSW	PZD1	HSW	HIW	PZD2	PZD3		PZD4		PZD5		PZD6

The process data region of the 20 byte telegram is used for exchanging cyclical data via the PROFIBUS. At each PROFIBUS cycle, an interrupt is triggered in the servo amplifier, with the interrupt causing new process data to be exchanged and processed.

The process data channel is used for real-time communication. This channel can effectively be divided into two telegram portions:

**PZD1:** Control word (STW) / status word (ZSW)  
The control word is used to control, the status word is used to supervise the status of the servoamplifier.

**PZD2-6:** Setpoints / actual values depending on the operating modes.  
Setpoints and actual values such as position, velocity etc. are exchanged in this section.

### 5.1 Process data PZD1

BYTE																				
1	2	3	4	4	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
PKW								PZD												
PKE		IND		PWE				STW	ZSW	PZD1	HSW	HIW	PZD2	PZD3		PZD4		PZD5		PZD6

The designation of the data word of the process data PZD1 is as follows:

- **STW** or **Control word** if the data flow is from the SPS to the servo amplifier. In this case the data is used to control the servo amplifier.
- **ZSW** or **Status word** if the data flow is from the servo amplifier to the SPS. In this case the data is used to monitor the servo amplifier.

### 5.1.1 The control word (STW)

By means of the control word you can switch from one servo amplifier state to another. The diagram for the state machine Chapter 5.2 shows which servo amplifier states can be reached via which transitions. The momentary servo amplifier state can be taken from the status word. Several states may be passed through during a telegram cycle (e.g. from Ready for switch-on → via Ready for operation → to Operation enabled).

The bits in the control word can be (operating) **mode-dependent** or **mode-independent**.

The following table describes the bit assignment in the control word:

Bit	Name	Comment
0	Switch on	---
1	Inhibit voltage	---
2	Fast stop, switch-on inhibited	1 -> 0 the drive brakes, using emergency ramp; axis is disabled
3	Operation enabled	---
4	Fast stop (inhibit ramp function generator)	1 -> 0 the drive brakes, using emergency ramp
5	Pause (stop RFG)	Position mode: <ul style="list-style-type: none"> <li>• Motion block: the parameter that is set in the motion block is used.</li> <li>• Setup operation: the parameter that is set as a ramp for homing and jogging is used</li> </ul>
6	Start_MT (transition edge)	Position mode: Start a motion task with every change of transition edge (toggle bit)
7	Reset Fault	Only effective with errors in PNU 1001
8	Jogging (On / Off)	Start jogging
9	Reserved	---
10	PZD (enable / inhibit)	---
11	Start homing run (transition edge)	Position mode: Start homing
12	Manufacturer-specific	Reset the position
13	Manufacturer-specific	Acknowledge warnings in PNU 1002
14	Manufacturer-specific (direct- or EEPROM MT)	Only in position mode: <ul style="list-style-type: none"> <li>• Bit14 = 1: PZD section is interpreted as a direct motion block (velocity 32-bit, position 32-bit, type of motion block 16-bit)</li> <li>• Bit14 = 0: PZD section (HSW) is interpreted as a motion block number.</li> </ul>
15	Manufacturer-specific	Reserved

Priority of the bits 6, 8, 11 in position-control mode (standard): 6 (high), 11, 8 (low).

Depending on the bit combination in the control word, a corresponding control command is defined. The following table shows the bit combinations and also determines the priorities of the individual bits, in case several bits are altered in one telegram cycle.

Transitions	Bit 13	Bit 7	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Transitions
Shutdown	x	x	X	x	1	1	0	2, 6, 8
Switch-on	x	x	X	x	1	1	1	3
Inhibit voltage	x	x	X	x	X	0	x	7, 9, 10, 12
Fast stop (disable)	x	x	X	x	0	1	x	7, 10, 11→12
Fast stop (enable)	x	x	0	1	1	1	1	11
Inhibit operation	x	x	X	0	1	1	1	5
Release operation	x	x	1	1	1	1	1	4, 16
Reset error	x	1	X	x	X	x	x	15
Acknowledge warnings	1	x	X	x	X	x	x	-

Bits marked x are irrelevant.

### 5.1.2 The status word (ZSW)

By means of the status word, the servo amplifier state can be represented and the transmitted control word can be verified. If an unexpected condition is reported as the result of a transmitted control word, then the boundary conditions for the expected servo amplifier state must first be clarified (e.g. enable the output stage – hardware + software, application of the DC-link voltage).

The bits in the status word can be (operating) **mode-dependent** or **mode-independent**.

The following table describes the bit assignment in the status word.

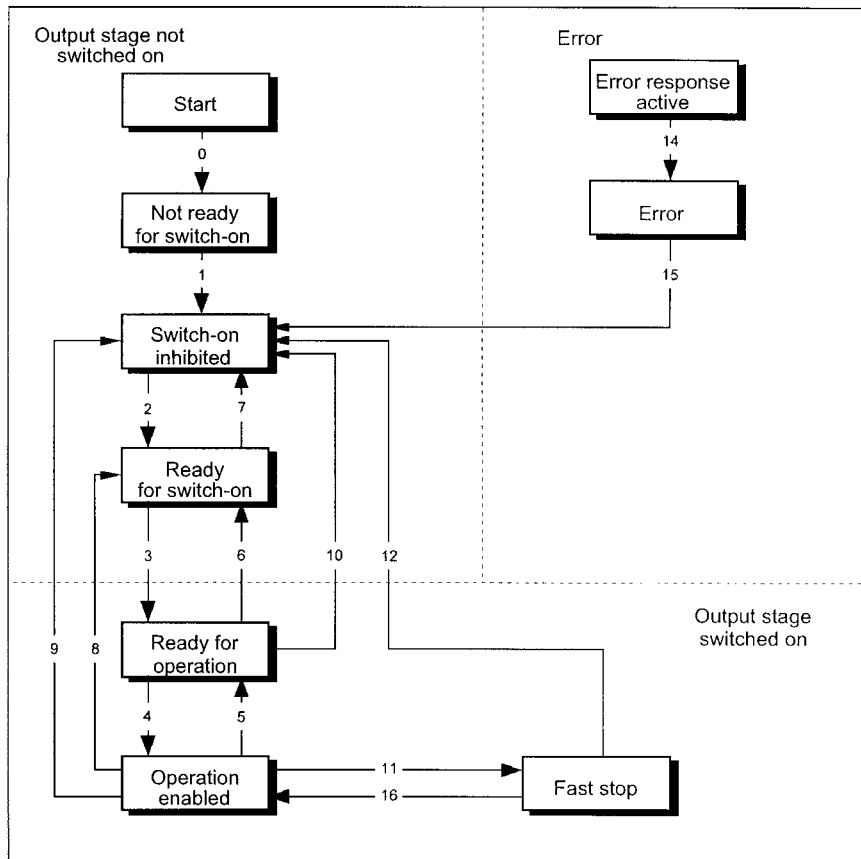
Bit	Nam	Comment
0	Ready for switch-on	-
1	Switched on	-
2	Operation enabled	-
3	Error	-
4	Voltage inhibited	-
5	Fast stop	-
6	Switch-on inhibited	-
7	Warning	-
8	Setpoint / actual value monitoring	Contouring error indicator
9	Remote	Not supported, fixed to 1
10	Setpoint reached	In position
11	Limit active	Not supported at present
12	Mode-dependent	Reserved
13	Mode-dependent	Reserved
14	Manufacturer-specific	Reserved
15	Manufacturer-specific	Reserved

States of the state machine:

State	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Not ready for switch-on	0	x	x	0	0	0	0
Switch-on inhibited	1	x	x	0	0	0	0
Ready for switch-on	0	1	x	0	0	0	1
Ready for operation	0	1	x	0	0	1	1
Operation enabled	0	1	x	0	1	1	1
Error	0	x	x	1	0	0	0
Error response	0	x	x	1	0	0	0
Fast stop active	0	0	x	0	1	1	1
Acknowledge warnings	1	x	x	x	X	x	x

## 5.2 Servo amplifier control

Control of the servo amplifier is described by means of a state machine. The state machine is defined in the drive profile in a flow diagram for all operating modes. The following diagram shows the possible instrument states of servo amplifier.



### 5.2.1 States of the state machine

Not ready for switch-on	The servo amplifier is not ready for switch-on. No operation readiness (BTB/RTO) is signalled from the amplifier software.
Switch-on inhibited	The servo amplifier is ready for switch-on. Parameters can be transferred, DC-link (DC-bus) can be switched on, motion functions cannot be carried out yet.
Ready for switch-on	DC-link voltage must be applied. Parameters can be transferred, motion functions cannot be carried out yet.
Ready for operation	DC-link voltage must be switched on. Parameters can be transferred, motion functions cannot be carried out yet. Output stage is switched on (enabled).
Operation enabled	No error present. Output stage is switched on, motion functions are enabled.
Fast stop activated	The drive has been stopped using the emergency stop ramp. Output stage is switched on (enabled), motion functions are enabled.
Error response active / error	If an instrument error occurs, the servo amplifier changes to the instrument state "Error response active". In this state, the power stage is switched off immediately. After this error response has taken place, it changes to the state "Error". This state can only be terminated by the bit-command "Error-reset". To do this, the cause of the error must have been removed (see Chapter 4.2.3 PNU1001)

## 5.2.2 Transitions of the state machine

Transition 0	Event	Reset / 24 V supply is switched on.
	Action	Initialisation started.
Transition 1	Event	Initialisation successfully completed, the servo amplifier switch-on inhibited.
	Action	None.
Transition 2	Event	Bit 1 (inhibit voltage) and bit 2 (fast stop) are set in the control word (command: shutdown). DC-link voltage is present.
	Action	None.
Transition 3	Event	Bit 0 (switch-on) is also set (command: switch-on).
	Action	Output stage is switched on (enabled). Drive has torque.
Transition 4	Event	Bit 3 (operation enabled) is also set (command: operation enable).
	Action	Motion functions are enabled, depending on the operating mode that is set.
Transition 5	Event	Bit 3 is cancelled (command: inhibit).
	Action	Motion functions are disabled. Brake is applied to drive, using the relevant ramp (depends on operating mode).
Transition 6	Event	Bit 0 is cancelled (ready for switch-on).
	Action	Output stage is switched off (disabled). Drive has no torque.
Transition 7	Event	Bit 1 or bit 2 is cancelled.
	Action	(Command: "Fast stop" or "Inhibit voltage").
Transition 8	Event	Bit 0 is cancelled (operation enabled -> ready for switch-on).
	Action	Output stage is switched off (disabled) - motor loses torque.
Transition 9	Event	Bit 1 is cancelled (operation enabled -> switch-on inhibited).
	Action	Output stage is switched off (disabled) - motor loses torque.
Transition 10	Event	Bit 1 or 2 is cancelled (ready for operation -> switch-on inhibited).
	Action	Output stage is switched off (disabled) - motor loses torque.
Transition 11	Event	Bit 2 is cancelled (operation enabled -> fast stop).
	Action	Drive is stopped using the emergency ramp. The output stage remains "enabled". Setpoints are cancelled (e.g. motion block number, digital setpoint).
Transition 12	Event	Bit 1 is cancelled (fast stop -> switch-on inhibited).
	Action	Output stage is switched off (disabled) - motor loses torque.
Transition 13	Event	Error response active.
	Action	Output stage is switched off (disabled) - motor loses torque.
Transition 14	Event	Error.
	Action	None.
Transition 15	Event	Bit 7 is set (error -> switch-on inhibited).
	Action	Acknowledge error (depending on error - with / without reset).
Transition 16	Event	Bit 2 is set (fast stop -> operation enabled).
	Action	Motion function is enabled again.

The state transitions are affected by internal events (e.g. switching off the DC-link voltage) and by the flags in the control word (bits 0, 1, 2, 3, 7).

### 5.3 Process data PZD2 to PZD6

BYTE																			
1	2	3	4	4	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
PKW								PZD											
PKE		IND		PWE				STW ZSW PZD1	HSW HIW PZD2	PZD3		PZD4		PZD5		PZD6			

The designation of the data word of the process data PZD2 is as follows:

- **HSW** or **Main setpoint value** if the data flow is from the SPS to the servo amplifier. In this case this value is used to control the servo amplifier.
- **HIW** or **Main actual value** if the data flow is from the servo amplifier to the SPS. In this case this value is used to monitor the servo amplifier.

**The significance of process data PZD2 to PZD6 varies;** this is explained in the next Chapter.

## 5.4 Using the process data channel

### 5.4.1 Standard allocation of the process data compartments

The standard allocation is used in the following cases:

- Home run (referencing)
- Jogging
- Saved motion task (STW bit 14="0")

PZD 1	PZD 2	PZD 3	PZD 4	PZD 5	PZD 6
STW	Motion task number or $v_{soll}$ (jogging/homing)	-	-	-	-
ZSW	$n_{ist}$ (16 bit)	Actual position (32 bit)		Manuf.-spec. status register PNU 1002	-

#### Motion task number

The motion task number of the motion task to be started can range from 1 to 180 (EEPROM motion tasks) or from 192 to 255 (RAM motion tasks).

#### $v_{soll}$ (jogging/homing)

The speed is specified in  $\mu\text{m/s}$  and is transferred as a 16-bit value. The 32-bit speed is determined from it through multiplication with the value of the parameter PNU1250.

#### $n_{ist}$ (16 bit) actual speed

Note:

**The actual speed in  $\text{min}^{-1}$  is directly displayed in PNU 1407.**

The 16 bit actual speed value is standardised and shown in increments. Positive and negative directions of rotation are shown. The actual motor speed is determined as follows:

Motor speed [1/min] =  $n_{ist}$  (16 bit) [Ink] x  $K_n$  where  $K_n$ :

Axis type	SHA-340 SHX-340 SHY-340	SHA-470 SHX-470 SHY-470	SHE-130 SVE-130	SVA-130
Factor $K_n$	0,18005	0,10986	0,14648	0,14648

#### Actual position (32 bit)

Note:

**The actual position in  $\mu\text{m}$  is directly displayed in PNU 1403.**

The 32 bit actual position value is shown in increments. Conversion of the actual position to SI units is as follows:

Actual position (SI unit) [ $\mu\text{m}$ ] = actual position (32 bit) [Ink] x  $K_v$  where  $K_v$ :

Axis type	SHA-340 SHX-340 SHY-340	SHA-470 SHX-470 SHY-470	SHE-130 SVE-130	SVA-130
Factor $K_v$	0,02622604	0,04450142	0,01525879	0,01901318

#### Manufacturer-specific status register

The process data provide the upper 16 bits of the manufacturer-specific status register (PNU 1002), with numbering starting at 0 again. The significance of the status register bits is shown in the table in Chapter 4.3.2 PNU 1002 bits 16 - 32.

### 5.4.2 Deviating allocation of the process data compartments

The deviating allocation is used in the following case:

- **Direct motion task** (STW bit 14="1")

PZD 1	PZD 2	PZD 3	PZD 4	PZD 5	PZD 6
STW	<b>Vsoll (32 bit)</b>		<b>Setpoint position (32 bit)</b>		<b>Type of motion task PNU 1302</b>
ZSW	nist (16 bit)	Actual position (32 bit)	Manuf.-spec. status register PNU1002	-	

**Vsoll [Vsetpoint] (32 bit)**

Speed values are specified in  $\mu\text{m/s}$ .

Condition: Bit13 = 1 in PNU 1302 or type of motion task

**Setpoint position (32 bit)**

Specification of position values is in  $\mu\text{m}$ .

Condition: Bit13 = 1 in PNU 1302 or type of motion task

**Type of motion task**

the type of motion task is set here, according to PNU 1302 Chapter 4.3.2.

## 5.5 Example telegram in PZD

### 5.5.1 Starting the home run (referencing)

The home run is started via bit 11 = 1 in STW.

Prerequisites: Hardware enable is active (connector X3 terminal 15)  
 Amplifier is in the state „operation enabled“  
 Operating mode (PNU 930) is set to 2

HSW (speed in  $\mu\text{m/s}$ ): 32768 dec = 10000000 0000000 bin

The process data telegram now has the following appearance:

PZD...					
9	10	11	12	13	14
C3F		8000		0000	
00001100	00111111	10000000	00000000	00000000	00000000
STW		HSW		PZD3	
STW		V <sub>soll</sub>		-	

BYTE  
Hex  
Bin

...PZD					
15	16	17	18	19	20
0000		0000		0000	
00000000	00000000	00000000	00000000	00000000	00000000
PZD4		PZD5		PZD6	
-		-		-	

BYTE  
Hex  
Bin



### 5.5.2 Starting a saved motion task

Motion tasks are started via a flank (positive or negative) in bit 6 in STW.

A saved motion task is selected via bit 14 = 0 in STW. In this case, the parameters have to be input to the amplifier via the Montech software display page <<Positioning data>> Chapter 2.4 (User Manual "Setup Software Version Profibus").

Note: Motion tasks from number **1** to **255** can be selected, with motion tasks 181 to 255 being resident in the volatile memory (RAM). Motion task 0 is reserved for the direct motion task.

Prerequisites: Hardware enable is present (terminal X3 terminal 15)  
 Amplifier is in the state „operation enabled“  
 Operating mode (PNU 930) is set to 2  
 Home run has been carried out or reference point has been set.

HSW (motion task number): 1 = 00000000 0000001 bin

The process data telegram then has the following appearance:

PZD...					
9	10	11	12	13	14
47F*		0001		0000	
00000100	0F**1111111	00000000	00000001	00000000	00000000
STW		HSW		PZD3	
STW		Motion task number		-	

**BYTE**  
**Hex**  
**Bin**

...PZD					
15	16	17	18	19	20
0000		0000		0000	
00000000	00000000	00000000	00000000	00000000	00000000
PZD4		PZD5		PZD6	
-		-		-	

**BYTE**  
**Hex**  
**Bin**

47F\* hex. In this case a rising flank is used to give the start signal.

F\*\* represents a flank change; the state of bit 6 in STW thus depends on the previous state.

By setting bit 16 in the manufacturer-specific status register PNU 1002 or bit 0 in PZD5 **Input**, the servo amplifier indicates that it has taken over the motion task and is carrying it out.

### 5.5.3 Starting a direct motion task

Motion tasks are started via a flank (positive or negative) in bit 6 in STW.  
 A direct motion task is started via bit 14 = 1 in STW.

Prerequisites:     Hardware enable is present (terminal X3 terminal 15)  
                   Amplifier is in the state „operation enabled“  
                   Operating mode (PNU 930) is set to 2  
                   Home run has been carried out or reference point has been set.

HSW + PZD3 (speed  $V_{soll}$  in  $\mu\text{m/s}$ ):                     1,500,000 dec = 10110 11100011 01100000 bin  
 PZD4 + PZD5 (position  $S_{soll}$  in  $\mu\text{m}$ ):                     800,000 dec =   1100 00110101 00000000 bin  
 PZD6 (type of motion task PNU 1302):                                     1000000 00000000 bin

Significance of PZD6:

In this case, in the saved motion task with absolute positioning (bit 0 = 0) without motion task (bit 3 = 0), the factory default acceleration (bit 12 = 0) with values in metric units (bit 13 = 1) is selected.

The process data telegram then has the following appearance:

PZD...						
9	10	11	12	13	14	
447F*		16		E360		BYTE Hex Bin
01000100	0F**1111111	00000000	00010110	11100011	01100000	
STW		HSW		PZD3		
STW		$V_{soll}$ (32 bit)				

...PZD						
15	16	17	18	19	20	
C		3500		2000		BYTE Hex Bin
0000000	0001100	00110101	0000000	00100000	00000000	
PZD4		PZD5		PZD6		
$S_{soll}$ (32 bit)				Type of motion task PNU 1302		

447F\*. In this case a rising flank is used to give the start signal.

F\*\* represents a flank change; the state of bit 6 in STW thus depends on the previous state.

By setting bit 16 in the manufacturer-specific status register PNU 1002 or bit 0 in PZD5 **Input**, the servo amplifier indicates that it has taken over the motion task and is carrying it out.

## 6. Steps necessary to perform motion tasks

This chapter describes all the important stages for successful setup.

The following three steps are required:

- **Operating mode setting**
- **Home run (referencing)**
- **Motion task**

It is absolutely essential that the above steps be carried out in the order indicated.

The following table illustrates the data exchange with the servo amplifier. The table shows the display page <<PROFIBUS>>, see Montech User Manual Setup Software (Version Profibus) Chapter 2.6, or the above mentioned display page in the Montech software 2.0.

	PKW				PZD					
	PKE	IND	PWE1	PWE2	STW	HSW	PZD3	PZD4	PZD5	PZD6
Output	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX
	PKE	IND	PWE1	PWE2	ZSW	HIW	PZD3	PZD4	PZD5	PZD6
Input	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX

Note:

- Values shown are hexadecimal.
- Output refers to data which is sent by the control system (SPS).
- Input refers to data which is read by the control system (SPS).
- PWE (32 bit) is divided into PWE1 (16 bit) and PWE2 (16 bit). This division also applies to the program in the next chapter.
- In the above-mentioned table, all data fields are set to XXXX. This means that there is no Profibus communication with the SPS.

### 6.1 Setting the operating mode

After switching it on or after a RESET, the servo amplifier is in the operating mode –126, in which it cannot carry out any motion functions. In order to carry out positioning processes (home run, jogging, motion tasks) it must first be brought into the positioning mode (operating mode 2).

Changing the operating mode is carried out in the following order:

#### 6.1.1 Locking the setpoint values and process data

Bit 10 in the control word is set to 0, so that no new setpoint values can be taken over from the servo amplifier and no new control functions can be triggered.

Prerequisites: Hardware enable is present (terminal X3 terminal 15)  
Amplifier is in the state „operation enabled“

Example:

	PKW				PZD					
	PKE	IND	PWE1	PWE2	STW	HSW	PZD3	PZD4	PZD5	PZD6
Output	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
	PKE	IND	PWE1	PWE2	ZSW	HIW	PZD3	PZD4	PZD5	PZD6
Input	0000	0000	0000	0000	0250	0000	0000	0000	0000	0000

Note:

In this case the entire STW was set to 0.

### 6.1.2 Resetting PKW

In the following cases PKW should be reset (zero telegram):

- At the start of communicating with the amplifier.
- When changing between writing (AK=3) and reading (AK=1) a parameter.
- After an error has occurred (AK=7)

The zero telegram should be present until the servo amplifier replies with a zero telegram.

Prerequisites: Hardware enable is present (terminal X3 terminal 15)  
 Amplifier is in the state „operation enabled“

Example:

	PKW				PZD					
	PKE	IND	PWE1	PWE2	STW	HSW	PZD3	PZD4	PZD5	PZD6
Output	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
	PKE	IND	PWE1	PWE2	ZSW	HIW	PZD3	PZD4	PZD5	PZD6
Input	0000	0000	0000	0000	0250	0000	0000	0000	0000	0000

### 6.1.3 Selecting the new operating mode with PNU 930

With parameter 930 the new operating mode is selected via the parameter channel, but not taken over yet.

Prerequisites: Hardware enable is present (terminal X3 terminal 15)  
 Amplifier is in the state „operation enabled“

Example:

	PKW				PZD					
	PKE	IND	PWE1	PWE2	STW	HSW	PZD3	PZD4	PZD5	PZD6
Output	33A2	0000	0000	0002	0000	0000	0000	0000	0000	0000
	PKE	IND	PWE1	PWE2	ZSW	HIW	PZD3	PZD4	PZD5	PZD6
Input	23A2	0000	0000	0002	2227	VAR	0000	VAR	VAR	0000

Note:

Operating mode number 2 is set (PWE2).

PKE input is set to 23A2 hex. This means that the parameter value (operating mode 2) has been transferred.

VAR in PZD Input: values vary; this indicates that the operating mode has been set.

### 6.1.4 Releasing the setpoint values

Bit 10 STW is set to 1. The setpoint values are immediately taken over and processed. The new actual values with the respective standardisation and data format are issued.

Prerequisites: Hardware enable is present (terminal X3 terminal 15)  
 Amplifier is in the state „operation enabled“

Example:

	PKW				PZD					
	PKE	IND	PWE1	PWE2	STW	HSW	PZD3	PZD4	PZD5	PZD6
Output	0000	0000	0000	0002	0400	0000	0000	0000	0000	0000
	PKE	IND	PWE1	PWE2	ZSW	HIW	PZD3	PZD4	PZD5	PZD6
Input	0000	0000	0000	0002	2220	<b>VAR</b>	0000	<b>VAR</b>	<b>VAR</b>	0000

Note:

PKE was reset to 0

VAR: values vary; this indicates that the operating mode has been set.

### 6.1.5 Error

If an error occurs, the amplifier responds as follows

Example:

	PKW				PZD					
	PKE	IND	PWE1	PWE2	STW	HSW	PZD3	PZD4	PZD5	PZD6
Output	33A2	0000	0000	0002	0000	0000	0000	0000	0000	0000
	PKE	IND	PWE1	PWE2	ZSW	HIW	PZD3	PZD4	PZD5	PZD6
Input	73A2	0000	0000	0011	0250	0000	0000	0000	0000	0000

Note:

Operating mode number 2 is set.

PKE input is set to 73A2 hex. This means that an error has occurred.

PWE2 input indicates the type of error. In this case it is 11 hex or 17 dec: "Task cannot be executed due to operating status". This error occurs for example if the STW is not locked right from the start.

## 6.2 Home run (referencing)



**Important**

After switching on the 24V auxiliary voltage, a home run (referencing) must first be carried out.

If for example the reference point (mechanical zero point) is approached at too high a speed, overshooting may occur, and in the most unfavourable case, the axis can move against the hardware limit switch or the mechanical end stop. There is a danger of damage occurring.

- The home run is started by setting bit 11 = 1 in STW.
- In PZD5 Input (manufacturer-specific status register PNU 1002) bit 0, the servo amplifier reports the start of the home run by setting bit 0 = 1.
- When the reference point has been reached, the servo amplifier sets bit 1 = 1 in PZD5 Input, thus reporting that the reference point has been set, and it then enables the position controller.
- After completion of the home run, bit 11 has to be reset to 0.
- If bit 11 is set to 0 before the reference point has been reached, then the home run is interrupted and bit 1 = 0 is set in PZD5 Input (reference point not set).

The speed of the home run is transmitted with the main setpoint value (HSW) as a 16-bit value. By multiplying it with the value of parameter PNU 1250, the 32-bit speed is determined from it. The operational sign is not evaluated.

Note:

If the result, 16-bit speed multiplied by parameter PNU 1250, is very much larger than 32 bits, a very low actual reference speed may result. Only the first 32 bits are used.

### 6.2.1 Starting the home run (referencing)

Prerequisites:     Hardware enable is present (terminal X3 terminal 15)  
                           Amplifier is in the state „operation enabled“  
                           Operating mode (PNU 930) is set to 2

Example:

	PKW				PZD					
	PKE	IND	PWE1	PWE2	STW	HSW	PZD3	PZD4	PZD5	PZD6
Output	0000	0000	0000	0000	C3F	8000	0000	0000	0000	0000
	PKE	IND	PWE1	PWE2	ZSW	HIW	PZD3	PZD4	PZD5	PZD6
Input	0000	0000	0000	0000	0227	VAR	VAR	VAR	4401	0000

Note:

No multiplication value was entered in the parameter PNU 1250 dec (multiplication of the reference speed).

### 6.3 Starting a direct motion task with saved next motion task

If the motion record data is to be specified freely, then a direct motion task must be used. This is where target position, speed and type of motion task are transferred to the process data together with calling up the motion task. If required, further parameters (e.g. acceleration, etc.) of this motion task can be entered beforehand as parameter tasks.

First, the number of the next motion task is entered via parameter PNU 1308 dec.

#### 6.3.1 Number of the motion task (only the saved motion task)

Output:

- PKE output= [AK=3 (write)&SPM=0&PNU=1308] dec = [0011&0&10100011100] bin = 351C hex
- PWE2 output= 1 (motion task no.)

Example:

	PKW				PZD					
	PKE	IND	PWE1	PWE2	STW	HSW	PZD3	PZD4	PZD5	PZD6
Output	351C	0000	0000	0001	0000	0000	0000	0000	0000	0000
	PKE	IND	PWE1	PWE2	ZSW	HIW	PZD3	PZD4	PZD5	PZD6
Input	251C	0000	0000	0001	2227	VAR	0000	VAR	VAR	0000

Input:

- PKE input is set to 251C hex. This means that the parameter value was transferred without any errors.
- PWE2 input = 1 (next motion task no.)

The amplifier has confirmed the data transfer; the direct motion task can be started.

#### 6.3.2 Starting a direct motion task

Prerequisites: Hardware enable is present (terminal X3 terminal 15)  
 Amplifier is in the state „operation enabled“  
 Operating mode (PNU 930) is set to 2  
 Home run successfully carried out

Output:

- STW: 447F hex (bit 14= 1, starting with rising flank in bit 6)
- Vsoll (HSW + PZD3): 1,500,000 µm/s dec = 16E360 hex
- Ssoll (PZD4 + PZD5): 1,000,000 µm dec = F4240 hex
- Type of motion task (PZD6): 2008 hex or motion task absolute, with next motion task and SI units

Example:

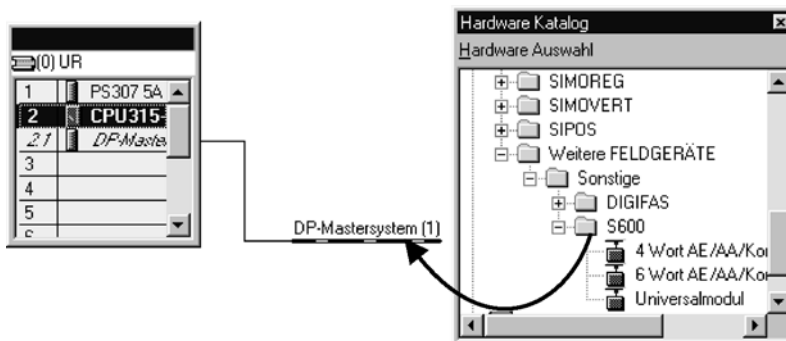
	PKW				PZD					
	PKE	IND	PWE1	PWE2	STW	HSW	PZD3	PZD4	PZD5	PZD6
Output	351C	0000	0000	0001	447F	0016	E360	000F	4240	2008
	PKE	IND	PWE1	PWE2	ZSW	HIW	PZD3	PZD4	PZD5	PZD6
Input	251C	0000	0000	0001	0A27	VAR	VAR	VAR	4403	0000

## 7. Program description for Siemens S7

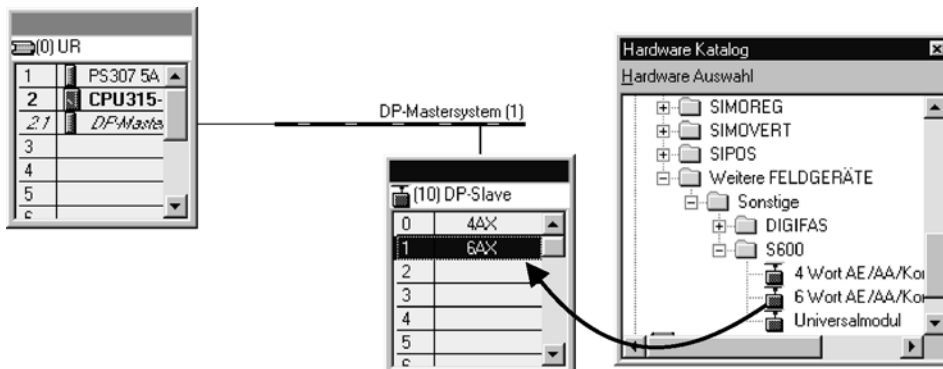
### 7.1 Hardware configuration of the CPU

The graphical interface makes it very easy to configure the Siemens S7 for the PROFIBUS network. After the control layout has been set up, the interface module used can be configured as follows:

Open the hardware catalogue and drag the symbol for the corresponding field unit onto the representation of the bus system (drag & drop). A window automatically opens for general parameterisation of the field unit. Enter the participant's address here.



Next, using the same method as above, drag the modules from the hardware catalogue into the box for the field unit. Note that the 4-word module must be placed into cell 0 and the 6-word module into cell 1.



Another window opens where you can set the parameters for the modules.





## 7.2 Program overview

The supplied 3.5" diskette contains 3 demonstration programs:

- 1xDEMO\_D (in German) or 1xDEMO\_E (in English)  
It controls 1 axis. This program has been prepared with absolute, symbolic addresses.
- 2xDEMi\_D (in German) or 2xDEMi\_E (in English)  
It controls 2 axes. This program has been prepared with register-indirect area-internal addresses.
- 3xDEMi\_D (in German) or 3xDEMi\_E (in English)  
It controls 1 servo cartesian system (3 axes). This program has been prepared with register-indirect area-internal addresses.

All 3 programs are structured in a similar way; the following description thus applies to all 3 programs.

**System requirements:** S7-315-2 DP or other S7 control systems with DP operation  
Simatic Step 7, version 2.1 or later.

### 7.2.1 Generally



**Important**

The program always starts immediately when the key switch on the CPU is switched from STOP to RUN; in other words, setting the operating mode, carrying out the home run as well as motion tasks, are performed immediately! Always ensure that no personal injury or damage to property can occur, before switching the key switch to RUN.

- Neither input nor output modules are used.
- After switching the program on (key switch on the CPU from STOP to RUN), the following steps **always** occur: the operating mode is set; a home run (referencing) is carried out; and then the motion task is carried out.

### 7.2.2 Description of the modules

The program comprises the following modules:

These modules monitor the program sequence

- OB1 Calls FC0
- OB100 Is called up by OB1 at program start.

These modules contain the program (logic)

- FC0 Contains the logic for moving the axes. Calls FC3, FC4, FC1 and FC2.
- FC3 Setting of the operating mode (calls FC5) and the home run are carried out.
- FC5 Reading and writing of the parameter PNU are carried out.
- FC4 The motion task is carried out.

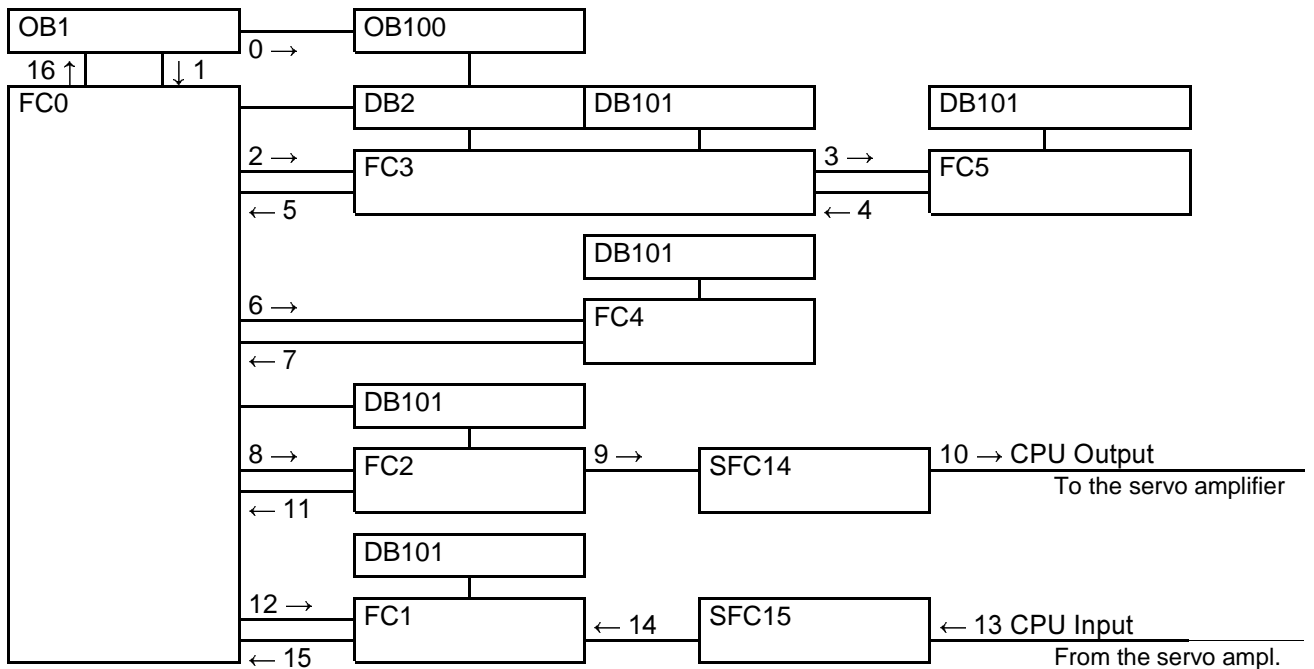
These modules are responsible for data exchange between the CPU (DB) and the servo amplifier

- FC2 Prepares to read the data from the servo amplifier (calls SFC14).
- SFC14 Reads consistent data from the servo amplifier.
- FC1 Prepares to write the data to the servo amplifier (calls SFC15).
- SFC15 Writes data consistently to the servo amplifier.

These modules are the program memory

- DB2 Flag database.
- DB101 Database of the useful data or PKW data and PZD data.

### 7.2.3 Program structure



Note:  
 The program sequence follows the numbering system.  
 OB100 is called up only at program start.

### 7.2.4 Programs 1xDEMO\_D (E), 2xDEMi\_D (E) and 3xDEMi\_D (E)

#### Equipment-specific settings

Before switching the CPU to RUN, settings must be entered in the following modules:

- **FC0** see Chapter 7.2.4.2
- **FC4** see Chapter 7.2.4.6

#### 7.2.4.1 OB100

At each program start (CPU switch from STOP to RUN) the organisation module OB100 is called up by OB1.

In this module, flags are programmed which are deleted at each start. This ensures that at startup, the operating mode is always set and the home run (referencing) is always carried out.

### 7.2.4.2 FC0

This is the central module in the program.

Network 1: This is where all FC3s are called up or the operating mode is set and the home run for all axes is carried out.

Network 2: After network 1 has been blocked, the program jumps to network 2.  
 In a first step basic settings are made or the axes are prepared for movement.  
 On completion of the basic settings the program enters the logic part where a determination is made as to which axis is to move.  
 At the end FC4 or move axis is called up, as are FC2 as well as FC1, which are responsible for the data flow between the CPU and the servo amplifier.

<b>FC0</b>				
<b>Functions</b>	Call up the modules which set the operating mode (FC3); perform the home run (FC3) as well as the motion tasks (FC4).			
	Logic (select the axis which is to move)			
	Call up the modules which carry out the data flow (FC1 + FC2)			
	Open DB2 and DB101			
<b>Technical data</b>	Module name	<b>Drive XYZ</b>		
	Bit setting	None		
	Access to:	FC1, FC2, FC3, FC4, DB2, DB101		
<b>Function of parameters</b>				
<b>Inputs</b>	-	-	None	
<b>Outputs</b>	-	-	None	
<b>Transit parameters</b>	AX_GO	BOOL	Enable movement of axis X	
	AY_GO	BOOL	Enable movement of axis Y	
	AZ_GO	BOOL	Enable movement of axis Z	
	YMove	BOOL	YMove=1 → Y axis moves. YMove=0 → Y axis at standstill	
	ZHome		Z axis is in the home position (reference position)	
<b>Equipment-specific settings</b>				
<b>Network 1 - 1.2) &amp; 1.3 &amp; 1.4)</b>	<b>FC3</b>	<b>DataDBStart3</b>	<b>WORD</b>	<b>Start address of the data (PKW + PZD) in DB101</b>
		<b>BitDBStart3</b>	<b>WORD</b>	<b>Start address of the bits in DB2</b>
<b>Network 2 - 2.5)</b>	<b>FC4</b>	<b>DataDBStart4</b>	<b>WORD</b>	<b>Start address of the data (PKW + PZD) in DB101</b>
		<b>Ssoll</b>	<b>DINT</b>	<b>Only for direct motion task. Stroke in µm (32 bit)</b>
				<b>Vsoll" DINT" Only for direct motion task. Speed in µm/s (32 " Network 2 - 2.6)" FC2" PKWPeriStart2" WORD " Start address PKW area in the periphery" " " " PZDPeriStart2" WORD" Start address PZD area in the periphery</b>
		<b>DataDBStart2</b>	<b>WORD</b>	<b>Start address of the data (PKW + PZD) in DB101</b>
<b>Network 2 - 2.6)</b>	<b>FC1</b>	<b>PKWPeriStart1</b>	<b>WORD</b>	<b>Start address PKW area in the periphery</b>
		<b>PZDPeriStart1</b>	<b>WORD</b>	<b>Start address PZD area in the periphery</b>
		<b>DataDBStart1</b>	<b>WORD</b>	<b>Start address of the data (PKW + PZD) in DB101</b>

### 7.2.4.3 FC1

In this module the data from module DB101 is loaded, stored in the temporary memory of FC1, from which it is transmitted to the periphery of the CPU by means of SFC15.

Network 2: Write from PKW part.

Network 3: Write from PZD part.

<b>FC1</b>				
<b>Function</b>	Send data from the CPU to the servo amplifier			
<b>Technical data</b>	Module name	<b>Send Data</b>		
	Bit setting	None		
	Access to:	DB101, SFC15		
<b>Function of parameters</b>				
<b>Inputs</b>	PKWPeriStart1	WORD	Start address PKW area in the periphery	
	PZDPeriStart1	WORD	Start address PZD area in the periphery	
	DataDBStart1	WORD	Start address of the data (PKW + PZD) in DB101	
<b>Outputs</b>	None	-	-	
<b>Transit parameters</b>	PKW	ARRAY [1...4] WORD		
	PZD	ARRAY [1...6] WORD		
	ResPKW	INT	Result of processing PZD (=0 → no error)	
	ResPZD	INT	Result of processing PZD (=0 → no error)	
<b>Equipment-specific settings</b>				
<b>None</b>	-	-	-	-

Note:

For each servo amplifier, FC1 must be called up at least once in the cycle.

FC1 and FC2 access the same parameter record and must be parameterised identically for each drive.

**FC2 must be called up before FC1.**

### 7.2.4.4 FC2

In this module the data from the CPU periphery is stored in the temporary memory of FC1, by means of SFC14. From there it is loaded to the data module DB101.

Network 2: Read from PKW part.

Network 3: Read from PZD part.

<b>FC2</b>				
<b>Function</b>	Send data from the servo amplifier to the CPU			
<b>Technical data</b>	Module name	<b>Read Data</b>		
	Bit setting	None		
	Access to:	DB101, SFC14		
<b>Function of parameters</b>				
<b>Inputs</b>	PKWPeriStart2	WORD	Start address PKW area in the periphery	
	PZDPeriStart2	WORD	Start address PZD area in the periphery	
	DataDBStart2	WORD	Start address of the data (PKW + PZD) in DB101	
<b>Outputs</b>	None	-	-	
<b>Transit parameters</b>	PKW	ARRAY [1...4] WORD		
	PZD	ARRAY [1...6] WORD		
	ResPKW	INT	Result of processing PZD (=0 → no error)	
	ResPZD	INT	Result of processing PZD (=0 → no error)	
<b>Equipment-specific settings</b>				
<b>None</b>	-	-	-	-

Note:

For each servo amplifier, FC2 must be called up at least once in the cycle.

FC1 and FC2 access the same parameter record and must be parameterised identically for each drive.

**FC2 must be called up before FC1.**

### 7.2.4.5 FC3

Setting of the operating mode as well as the home run are controlled in this module.

Network 2: At the start of communication, a zero telegram is sent. If the zero telegram is acknowledged by the servo amplifier, the network is skipped during subsequent program passes.

Network 3: By calling up FC5, setting of the operating mode is carried out. Thus the operating mode is set to 2 or to positioning. If setting the operating mode has been carried out successfully, the flag BEA\_OK\_(X, Y or Z) is set in DB2.

Network 4: After network 3 has been blocked, the program jumps to network 4. In this part, the home run (referencing) is carried out. If the home run has been carried out successfully, the flag REF\_OK\_(X, Y or Z) is set in DB2.

<b>FC3</b>			
<b>Functions</b>	Call up the module (FC5) which carries out transfer of the parameter data PNU		
	Set the operating mode and home run		
<b>Technical data</b>	Module name	<b>OpModRef</b>	
	Bit setting	OpModS_OK (X, Y or Z) in DB2 REF_OK (X, Y or Z) in DB2 CommStart (X, Y or Z) in DB2 REF_0 (X, Y or Z) in DB2	
	Access to:	FC5, DB2, DB101	
<b>Function of parameters</b>			
<b>Inputs</b>	DataDBStart3	WORD	Start address of the data (PKW + PZD) in DB101
	BitDBStart3	WORD	Start address of the Bit in DB2
<b>Outputs</b>	None	-	-
<b>Transit parameters</b>	PKW_OpModS_OK	BOOL	Change of the operating mode settings has been carried out successfully
<b>Equipment-specific settings</b>			
<b>None</b>	-	-	-

**7.2.4.6 FC4**

The motion tasks are carried out in this module.

Network 1: This is where a selection is made whether a saved motion task or a direct motion task is to be carried out.

Network 2: This part carries out a saved motion task. In this case it is **not** necessary to set Ssoll (Ssetpoint) and Vsoll (Vsetpoint) in FC0.

Network 3: This part carries out a direct motion task. In this case, Ssoll (Ssetpoint) and Vsoll (Vsetpoint) must be set in FC0

<b>FC4</b>			
<b>Function</b>	Perform motion tasks		
<b>Technical data</b>	Module name	<b>Move</b>	
	Allocated flags	None	
	Access to:	DB101	
<b>Function of parameters</b>			
<b>Inputs</b>	Start	BOOL	Start a motion task by setting to 1 (Stop by setting to 0)
	DataDBStart4	WORD	Start address of the data (PKW + PZD) in DB101
	Ssoll	DINT	Only for direct motion task. Stroke in µm (32 bit)
	Vsoll	DINT	Only for direct motion task. Speed in µm/s (32 bit)
<b>Outputs</b>	None	-	-
<b>Transit parameters</b>	NbrMoTasBlo	INT	Motion task number for the saved motion task
	Ssollvar	DINT	Computed stroke in µm (32 bit)
<b>Equipment-specific settings</b>			
<b>Network 1 – 1.2)</b>	JU MveD	<b>Absolute jump to MveD in order to perform a direct motion task.</b>	
	JU MveB	<b>Absolute jump to MveB in order to perform a motion task block.</b>	
<b>Network 2 – 1st line</b>	MveB	<b>Jump target to perform a motion task block.</b>	
<b>Network 3 – 1st line</b>	MveD	<b>Jump target to perform a direct motion task.</b>	

### 7.2.4.7 FC5

The parameter data of the servo amplifier is read or written in this module.

Network 2: This is where a determination is made as to whether a change between AK of PKES.DB101 and AK of PKE (input) has taken place. If no change has taken place, network 3 is skipped. A change takes place at the time of:

- Start of communication with the service channel PKW
- Change between read and write

Network 3: This is where the reset from PKW to "0" is carried out.

Network 4: This is where the new parameter data is transferred. Each time the reply is checked by the servo amplifier.

<b>FC5</b>			
<b>Function</b>	Carry out transfer of parameter data		
<b>Technical data</b>	Module name	<b>R&amp;W_parameters</b>	
	Bit setting	PKW_0 (X, Y or Z) in DB2	
	Access to:	DB2, DB101	
<b>Function of parameters</b>			
<b>Inputs</b>	PKE	WORD	Parameter identification
	IND	WORD	Subindex
	PWE1	WORD	Parameter value 1
	PWE2	WORD	Parameter value 2
	DataDBStart5	WORD	Start address of the data (PKW + PZD) in DB101
	BitDBStart5	WORD	Start address of the Bit in DB2
<b>Outputs</b>	PKW_OK	BOOL	Transfer of parameters carried out successfully
<b>Transit parameters</b>	None	-	-
<b>Equipment-specific settings</b>			
<b>None</b>	-	-	-



### 7.2.4.8 DB2 "BitDB"

This data module contains the "bits" which are necessary for controlling the operating mode setting and the home run in FC3.

Address	Name	Data type	Comment
0.0	X_axis	STRUCT	Bit for the X axis
n + 0.0	OpModS_OK	BOOL	Operating mode setting done
n + 0.1	REF_OK	BOOL	Home (reference) run done
n + 0.2	PKW_0	BOOL	PKW="0" done
n + 0.3	CommStart	BOOL	PKW="0" at communication start done
n + 0.4	REF_0	BOOL	Home position (reference_0) must be left
n + 0.5	Reserve1	BOOL	Reserve
= 2.0		END_STRUCT	

Note:

(n = 0, 2, 4, 6, ... start address of the Bit data)  
 A new structure must be created for each axis.

### 7.2.4.9 DB101 "DataDB"

This data module contains all parameter data and process data; in other words, the useful data.

Address	Name	Comment	
0.0	X_axis	STRUCT	
n + 0	PKES	Parameter identification	PKW – Output
n + 2	INDS		
n + 4	PWE1S	Parameter value 1	
n + 6	PWE2S	Parameter value 2	PZD – Output
n + 8	STW	Control word	
n + 10	HSW	Main setpoint	
n + 12	PZD3S		
n + 14	PZD4S		
n + 16	PZD5S		PKW – Input
n + 18	PZD6S		
n + 20	PKEE	Parameter identification	
n + 22	PKWE		
n + 24	PWE1E		PZD - Input
n + 26	PWE2E		
n + 28	ZSW	Status word	
n + 30	HIW	Main actual value	
n + 32	PZD3E		
n + 34	PZD4E		
n + 36	PZD5E		
n + 38	PZD6E		
= 40.0		END_STRUCT	

Note:

(n = 0, 2, 4, 6, ... start address of the data)  
 A new structure must be created for each axis.

## 8. Appendix

### 8.1 Abbreviations and acronyms

The table below shows abbreviations and acronyms used in this manual.

Abbreviation / acronym	Explanation
AGND	Analog mass (ground)
Bin	Binary
BTB/RTO	Ready for operation
CE	Communauté Européenne
CENELEC	Comité Européen de Normalisation Electronique
CLK	Clock (timing signal)
COM	Serial interface of a PC-AT
Dec	Decimal
DGND	Digital mass (ground)
DIN	Deutsches Institut für Normung
Disk	Magnetic storage (diskette, hard disk)
EEPROM	Electrically erasable and programmable read-only memory
EMV	Electromagnetic compatibility
EN	European standard
ESD	Electrostatic discharge
Hex	Hexadecimal
IDC	Analog current monitor
IEC	International Electrotechnical Commission
IGBT	Insulated Gate Bipolar Transistor
INC	Incremental interface
ISO	International Standardization Organization
LED	Light Emitting Diode
MB	Megabyte
MS-DOS	Operating system for PC-AT
NI	Zero impulse
NSTOP	Limit switch input, anticlockwise rotation
PELV	Protective low voltage
PGND	Mass (ground) of the interface used
PSTOP	Limit switch input, clockwise rotation
PWM	Pulse-width modulation
RAM	Volatile memory
Rballast	Ballast resistor
Rbext	External ballast resistor
Rbint	Internal ballast resistor
RES	Resolver
SPS	Programmable logic controller PLC
SRAM	Static RAM
SSI	Synchronous serial interface
SW/SETP.	Setpoint
UL	Underwriter's Laboratories [USA, fire]
V AC	Alternating voltage
V DC	Direct voltage
VDE	Association of German electrical engineers
VGA	Graphics standard with at least 640x480 pixels
VTA	Analog rotation monitor
XGND	Mass (ground) of the 24V supply voltage

## 8.2 Glossary

<b>C</b>	Current controller	Controls the difference between current setpoint and actual value of the current to 0.
<b>E</b>	E <sup>2</sup> PROM	See EEPROM
	EEPROM	<b>E</b> lectrically <b>E</b> rasable and <b>P</b> rogramable <b>R</b> ead- <b>O</b> nly <b>M</b> emory Electrically erasable memory in the servo amplifier. Data stored in the EEPROM is not lost when the auxiliary voltage is switched off.
	Enable	Enable signal for the servo amplifier (+24)
<b>G</b>	GRAY format	Special form of binary code (with only one bit changing between sequential numbers)
	GSD-Datei	<b>G</b> eräte <b>S</b> tamm <b>D</b> aten-Datei The GSD file for the servo amplifier is included in the 3.5" diskette supplied
<b>H</b>	Holding brake	Motor brake which must only be applied with the motor at a standstill.
	HLG	<b>H</b> och <b>L</b> auf <b>G</b> eber Speed up probe of the frequency converter (according to Profidrive standard)
<b>I</b>	I <sup>2</sup> t	Monitoring of the effectively required root-mean-square (RMS) current
	Intermediate circuit	Rectified and smoothed output voltage
	I <sub>peak</sub> , peak current	Effective value of peak current
	I <sub>rms</sub> , effective current	Effective value of continuous current
<b>L</b>	LSB	<b>L</b> ower <b>S</b> ignificant <b>B</b> yte Byte is right-aligned
	LSW	<b>L</b> ower <b>S</b> ignificant <b>W</b> ord Word is right-aligned
<b>M</b>	Motion block	Data packet with all position control parameters which are required for a motion task.
	MSB	<b>M</b> ost <b>S</b> ignificant <b>B</b> yte Byte is left-aligned
	MSW	<b>M</b> ost <b>S</b> ignificant <b>W</b> ord Word is left-aligned
<b>O</b>	Optical coupler	Optical connection between two electrically independent systems
<b>P</b>	Position controller	Controls the difference between position setpoint and actual value to 0.
	PPO-Typ	<b>P</b> arameter <b>P</b> rocess data <b>O</b> bject type Communication with the servo amplifier takes place exclusively with PPO type 2.
<b>R</b>	RAM	<b>R</b> andom <b>A</b> ccess <b>M</b> emory Volatile memory in the servo amplifier. Data stored in RAM are lost if the auxiliary voltage is switched off.
	Reset	Restart of the micro processor
	ROD interface	Incremental position output
<b>S</b>	SI Einheiten	<b>S</b> ystème <b>I</b> nternational d'Unités or International System of Units which among others comprises the following base units: metre [m]; kilogram [kg]; second [s]; ampere [A]; etc.
	SSI interface	Cyclically absolute serial position-output
<b>Z</b>	Zero pulse	Is issued once per revolution by incremental transmitters; it is used for zeroing the machine.

### 8.3 Number systems

<b>Bin</b>	0; 1	Bin 11010 $\equiv$ dec 26 $\equiv$ hex 1A
<b>Dez</b>	0; 1; 2; 3; 4; 5; 7; 8; 9	Dec 8 $\equiv$ hex 8 $\equiv$ bin 1000
<b>Hex</b>	0; 1; 2; 3; 4; 5; 7; 8; 9; A; B; C; D; E; F	Hex FF $\equiv$ bin 11111111 $\equiv$ dec 255

### 8.4 Data types

<b>ARRAY</b>	ARRAY[1...4] ( WORD ) ARRAY[1...6] ( WORD )	Composite data types or 4 WORD Composite data types or 6 WORD
<b>BIT</b>		Can have a value of 0 or 1
<b>BOOL</b>		Equals bit (Siemens S7 format)
<b>BYTE</b>		1 Byte $\equiv$ 8 Bit
<b>INT</b>	<b>INTEGER</b> INTEGER16 INTEGER32 2*UINT16 UINT32	Integer > 0, or = 0, or < 0 Integer, length = 16 Bit Integer, length = 32 Bit Integer > = 0, length = 2 x 16 Bit Integer > = 0, length = 32 Bit
<b>Octet-String</b>	<b>Octet-String</b> Octet-String 2 Octet-String 20	1 octet string (1 Octet $\equiv$ 8 Bit) 2 octet strings ( $\equiv$ 16 Bit) 20 octet strings ( $\equiv$ 160 Bit)
<b>Visible-String</b>	Visible-String 4	4 ASCII characters ( $\equiv$ 4 Byte)
<b>WORD</b>	<b>WORD</b> DWORD	1 WORD $\equiv$ 2 Byte $\equiv$ 16 Bit <b>DoubleWORD</b> : DWORD $\equiv$ 4 Byte $\equiv$ 32 Bit