KOLLMORGEN
Seidel

PROFIBUS CONNECT

Interface module integrated in digifas™ 7100-L2/DP and digifas™ 7200-L2/DP

CE

Commissioning / technical description

Edition 08/97

Previous editions

Edition	Comments
07 / 96	First edition
08 / 97	Various corrections, reference traverse type 5, valid from Software Version 7x40

INTEL	is a registered trademark of Intel Corporation
MOTOROLA	is a registered trademark of Motorola Inc.
SINEC	is a registered trademaark of Siemens AG
digifas™	is Regd.TM. No 1549070 of Seidel Corporation

Technical changes to improve the performance of the equipment may be made without prior notice!

Printed in the Federal Republic of Germany 08/97

Mat. no. 83106

All rights reserved. No part of this work may be reproduced in any form (by printing, photocopying, microfilm or any other method) or stored, processed, copied or distributed by electronic means without the written permission of Seidel Corporation.

Contents

Diagram Page

Contents . . . . . A
Safety instructions . . . . . E
Directives and standards . . . . . F
CE-conformance . . . . . F

I General
I.1 About this manual I-1
I.2 “Use as directed” for the PROFIBUS CONNECT interface module I-1
I.3 Abbreviations used in this manual. I-2
I.4 Features of PROFIBUS CONNECT I-2
I.5 Frontal view digifas™ 7100/7200 with PROFIBUS CONNECT I-3
I.6 Pin assignments digifas™ 7100/7200 with PROFIBUS CONNECT . . . . . - A.4.016.4/2.4 . . . I-3
I.7 Numerical format I-4
I.8 Bus cable. I-4

II Installation / Commissioning
II.1 Hard- and Software installation for connection to SIMATIC S5 / IM308B(C) II-1
II.1.1 Connection methods II-1
II.1.2 Connection diagram digifas™ 7100-L2/DP / digifas™ 7200-L2/DP . . . . . - A.4.016.1/2. . . . II-2
II.1.3 Parameterization of the master interface modules II-3
II.1.3.1 Interface module IM308B(C) II-3
II.1.3.2 Interface module CP5412 (A2) II-3
II.1.4 PLC programming II-4
II.1.4.1 Standard function blocks for data exchange with digifas™ II-4
II.1.4.2 SEIDEL handling package DIGI_DP1. . . . . II-4
II.1.5 SINEC DP programming interface II-4
II.2 Commissioning. II-5
II.2.1 Commissioning the basic functions of the servo-amplifier II-5
II.2.2 Linear axis. II-6
II.2.2.1 Commissioning the position controller for a linear axis II-6
II.2.2.2 Notes on optimizing the linear axis II-7
II.2.3 Rotary axis II-8
II.2.3.1 Commissioning the position controller for a rotary axis II-8
II.2.3.2 Notes on optimizing the rotary axis II-9

III Telegram structure
III.1 Parameter process data object (PPO) to PROFIBUS-profile VDI/VDE 3689 . . . . - A.4.016.3/1 . . . III-1
III.2 Specific digifas™ instrument profile III-2
III.2.1 Parameter ID with PPO (PKE) . . . . . - A.4.016.3/1 . . . III-2
III.2.2 Subindex IND III-3
III.2.3 Parameter value PWE . . . . . - A.4.016.3/2 . . . III-3
III.2.4 Control word CW digifas™ profile III-4
III.2.5 Status word SW digifas™ profile III-5
III.2.6 Main setpoint HSW III-6
III.2.7 Main actual value HIW III-6

Contents

Diagram Page

IV Description of the functions

IV.1 Movement functions IV-1
IV.1.1 Setup and general functions IV-1
IV.1.1.1 Cyclical transmission of a control word to the digifas™ IV-1
IV.1.1.2 Cyclical transmission of a status word from digifas™ to the master. IV-1
IV.1.1.3 Jog operation IV-1
IV.1.1.4 Set reference point IV-1
IV.1.1.5 Reference traverse (Go to Reference) IV-2
IV.2 Positioning functions IV-3
IV.2.1 Direct order IV-3
IV.2.2 Starting a motion task from the digifas™ motion-task memory IV-4
IV.3 Data transmission functions IV-5
IV.3.1 Load data block IV-5
IV.3.2 Read data block IV-6
IV.3.2.1 Read actual values IV-7
IV.3.2.2 Read status register (PNU 1118) IV-8
IV.4 Setting the amplifier parameters IV-10
IV.4.1 Data format IV-10
IV.4.2 Write / read an amplifier parameter IV-10
IV.4.3 Summary of the extended parameter numbers IV-11
IV.4.3.1 Symbolic labels IV-11
IV.4.3.2 List of the parameter numbers IV-12
IV.4.4 Save the amplifier parameters to the EEPROM (PNU 1021) IV-14
IV.4.5 Mode changeover (PNU 1048) IV-14
IV.4.6 Digital setpoint (PNU 1049) IV-14
IV.4.7 Teach In (PNU 1113) IV-15
IV.5 Sample telegrams IV-15
IV.5.1 Transmit motion task IV-15
IV.5.2 Start reference traverse IV-15
IV.5.3 Start direct order IV-16
IV.5.4 Parameterizing of the amplifier, v_max IV-16

Contents

Diagram Page

V Parameter description

V.1 BS7200 : SERVICE menu page, fieldbus monitor V-1
V.2 BS7200 : CONNECT menu page V-1
V.3 Description of CONNECT parameters V-2
V.3.1 Kp, P-gain (PNU 1072) V-2
V.3.2 Ff, feed-forward factor (PNU 1073) V-2
V.3.3 Control from (PNU 1102) V-2
V.3.4 Mode (PNU 1048) V-2
V.3.5 Axis type (PNU 1101) V-3
V.3.6 Count direction (PNU 1097) V-3
V.3.7 Resolution (PNU 1079 / 1082) V-4
V.3.8 v_max, maximum speed (PNU 1076) V-5
V.3.9 Ramp type (PNU 1106) V-5
V.3.10 t_acc_min, maximum accelaration (PNU 1075) V-6
V.3.11 t_emergency, maximum braking deceleration (PNU 1074) V-6
V.3.12 Type of reference traverse (PNU 1107) . . . . . - A.024.3/8,10,13. V-7
V.3.13 Zero point offset (PNU 1104) V-13
V.3.14 Reference offset (PNU 1108) V-13
V.3.15 Limit-switch 2 (PNU 1100) V-13
V.3.16 Limit-switch 1 (PNU 1099) V-13
V.3.17 InPosition (PNU 1078) V-14
V.3.18 Contouring error (PNU 1077) V-14
V.3.19 Motion task V-14
V.3.20 Response monitoring (PNU 1103) V-14
V.3.21 Station address V-15
V.3.22 Jog speed 1 (PNU 1109) V-15
V.3.23 Jog speed 2 (PNU 1110) V-15
V.4 Description of the motion block parameter BS7200 . . . . . V-16
V.4.1 No. V-16
V.4.2 Type V-16
V.4.3 s_nom V-16
V.4.4 v_nom V-17
V.4.5 t_acc V-17
V.4.6 t_brake V-17
V.4.7 Next no. V-17
V.4.8 Pause V-17
V.5 Additional motor parameters V-18
V.5.1 K_E, voltage constant of the motor (PNU 1060) V-18
V.5.2 L, motor inductance (PNU 1063) V-18
V.6 Error log V-19
V.6.1 Error index (PNU 1111) V-19
V.6.2 Error statistics (PNU1112) V-19
V.7 Print parameters V-20
V.7.1 Print amplifier parameters V-20
V.7.2 Print motion-block parameters . . . . . - A.4.024.3/12. . V-20

Contents

Diagram Page

VI Appendix

VI.1 Function-block package DIGI_DP1 . . . VI-1
VI.1.1 FB190 DIGISTAT . . . VI-2
VI.1.2 FB 191 DIGIHAND . . . VI-2
VI.1.3 FB195 DIGIDIRE . . . VI-3
VI.1.4 FB196 DIGISATZ . . . VI-3
VI.1.5 FB197 DIGILADE . . . VI-4
VI.1.6 FB198 DIGILESE . . . VI-5
VI.1.7 FB199 DIGI-IST . . . VI-6
VI.1.8 FB 200 DIGIPARA . . . VI-7
VI.2 Function-block library (FB) . . . VI-8
VI.2.1 Further documentation / manuals . . . VI-8
VI.2.2 System requirements . . . VI-8
VI.2.3 Installation of the function-block library . . . VI-8
VI.2.4 Demo-program “profdemo.exe” . . . VI-9
VI.2.5 Description of the transfer structures of the function blocks . . . VI-10
VI.2.5.1 “Verwaltung” structure . . . VI-10
VI.2.5.2 “Datensatz” structure . . . VI-11
VI.2.5.3 “Istwerte” structure . . . VI-12
VI.2.5.4 Structure for starting a motion block from the motion-block memory . . . VI-13
VI.2.5.5 Structure for setting-up functions . . . VI-13
VI.2.5.6 Structure for transferring parameters, commands and actual values . . . VI-14
VI.2.6 Overview of the error IDs . . . VI-15
VI.2.7 Description of the function blocks for the CP5412 (A2) . . . VI-16
VI.2.7.1 Setting-up and general functions . . . VI-17
VI.2.7.1.1 Transmitting a control command (control word CW) to the digifas™ . . . VI-17
VI.2.7.1.2 Jog mode . . . VI-18
VI.2.7.1.3 Reference traverse . . . VI-18
VI.2.7.2 Position functions . . . VI-19
VI.2.7.2.1 Direct motion task . . . VI-19
VI.2.7.2.2 Starting a motion task from the digifas™ motion-block memory . . . VI-20
VI.2.7.3 Data transfer functions . . . VI-20
VI.2.7.3.1 Load data block . . . VI-20
VI.2.7.3.2 Read data block . . . VI-21
VI.2.7.3.3 Read out actual values and the error register . . . VI-21
VI.2.7.3.4 Read / write parameters, execute commands, request actual values . . . VI-22
VI.2.7.4 Auxiliary function for the control word SW . . . VI-23
VI.2.8 Description of the files for the function-block library . . . VI-24
VI.3 Sample parameterization der ET200 . . . VI-25
VI.3.1 Parameterization of the master module . . . VI-25
VI.3.1.1 Bus parameters . . . VI-25
VI.3.1.2 AG parameters . . . VI-25
VI.3.2 Configuration of digifas™ - PROFIBUS as a slave . . . VI-25
VI.4 Sample program . . . VI-26
VI.4.1 General notes . . . VI-26
VI.4.2 Listing for sample program . . . VI-27
VI.5 Parameter form for parameter CONNECT . . . VI-29
VI.6 Index . . . VI-30

Safety instructions

Warning signs : you must observe the important instructions in the text, which are indicated by the following symbols:

hazard from electricity and its effects

general warning general instruction

- **Only properly qualified personnel are permitted to perform activities such as transport, installation, commissioning and maintenance. Properly qualified persons are those who are familiar with transport, installation, assembly, commissioning and operation of the products, and who have the appropriate qualifications for their job. The qualified personnel must know and observe the following standards and directives:**
  IEC 364 and CENELEC HD 384 or DIN VDE 0100
  IEC Report 664 or DIN VDE 0110
  national accident prevention regulations or VBG 4

- **Read the available documentation before carrying out installation and commissioning. Incorrect treatment of the servo amplifier can lead to injury to persons or material damage. It is vital that you keep to the technical data and information on connection requirements, nameplate and documentation.**

- **The servo amplifiers contain electrostatically sensitive components which may be damaged by incorrect handling. Discharge your body before touching the servo amplifier. Avoid contact with highly insulating (artificial fabrics, plastic film etc.). Place the servo amplifier on a conductive surface.**

- **Do not open the units. Keep all covers and switchgear cabinet doors closed in operation. Otherwise there are deadly hazards with the possibility of severe danger to health or material damage.**

- **In operation, depending on the degree of enclosure protection, servo amplifiers can have bare components which are live, and hot surfaces. Control and power cables can carry a high voltage even when the motor is not rotating.**

- **Never undo the electrical connections of the servo amplifier when it is live. There is a danger of electric arcing and danger to persons and contact.**

- **Wait at least two minutes after disconnecting the servo amplifier from the mains supply voltage before touching live sections of the equipment or undoing connections (e.g. contacts, screwed connections). Capacitors can have dangerous voltages present up to two minutes after switching off the supply voltages. To be sure, measure the voltage in the intermediate circuit and wait until it has fallen below 40V.**

Directives and standards

Servo amplifiers are components which are intended to be incorporated into electrical machines and plant.

When the servo amplifiers are incorporated into machines or plant, the intended operation of the amplifiers is forbidden until it has been established that the machine or plant fulfills the requirements of the EC Directive on Machines 89/392/EEC and the EC Directive on EMC 89/336/EEC. EN 60204 and EN 292 must also be observed.

In connection with the Low Voltage Directive 73/23/EEC, the harmonized standards of the EN 50178 series are applied to the servo amplifiers, together with EN 60439-1, EN 60146 and EN 60204.

The manufacturer of the machine or plant is responsible for ensuring that the machine or plant meets the limits which are laid down by the EMC regulations. Advice on the correct installation for EMC – such as shielding, grounding, arrangement of filters, treatment of connectors and laying out the cabling - can be found in the installation intructions for the servo amplifier.

CE - Conformance

Conformance with the EC Directive on EMC 89/336/EEC is mandatory for the supply of servo amplifiers within the European Community from the 1st. of January 1996.

The correct installation for EMC is shown in the installation instructions for the servo amplifier. These instructions also show the components which are required (cables, mains filter etc.)

Any divergence from the configuration and installation which is described in the documentation means that you will be responsible for the performance of new measurements to ensure that the regulatory requirements are met.

Only on condition that the components which we have specified are used and the installation rules are observed can we guarantee that the servo amplifier conforms to the following standards for industrial areas:

EC EMC Directive 89/336/EWG

EC Low Voltage Directive 73/23/EWG

I General

I.1 About this manual

This manual describes the wiring, commissioning and range of functions of the PROFIBUS CONNECT interface module. It is part of the complete documentation for the digifas™ 7100 and digifas™ 7200 series of digital servo amplifiers.

The installation and commissioning of the servo amplifier, as well as all the standard functions, are described in the corresponding installation instructions.

Other parts of the complete documentation for the digifas™ 7100 and digifas™ 7200 series of digital servo amplifiers:

Title	Publisher	Order no.
Operator software BS7200, Manual	Seidel	83107
digifas™ 7200, Assembly, Installation, Commissioning	Seidel	83003
digifas™ 7100, Assembly, Installation, Commissioning	Seidel	83498

Further documentation :

Title	Publisher	Order no.
Decentral Peripheral System ET200, Manual	Siemens	EWA 4NEB 8126114-01a
Function blocks DVA_S5 incl. Manual	Siemens	E20125-C0002-S302-A1

This manual is intended for the use of qualified staff with the following knowledge:

Wiring : Electrically trained personnel
Programming : Software developers, PROFIBUS project engineers

We offer training and familiarisation courses on request.

I.2 “Use as directed” for the PROFIBUS CONNECT interface module

The PROFIBUS-CONNECT interface module is permanently built into digital servo amplifiers of the digifas™ 7100-L2/DP and digifas™ 7200-L2/DP series.
These servo-amplfiers are to be used **only** on earthed three-phase 400V industrial mains supplies (TN or TT), and to drive synchronous servomotors from the 6SM series.
The digifas™ 7100-L2/DP and digifas™ 7200-L2/DP series of servo amplifiers are **exclusively** intended for the closed-loop drive of brushless synchronous servomotors from the 6SM series of servomotors.
The PROFIBUS CONNECT interface module serves only to connect the servo amplifier to a master which has a PROFIBUS interface.
The servo-amplifiers are components which are installed in electrical equipment or machines and can only be commissioned as integral components of such equipment or machines.
You must prepare a hazard analysis before commissioning the machine or system.

Only on condition that the components described by us are used and the rules for installation instructions are followed can we guarantee the conformance of the servo amplifiers with the following standards for industrial areas:
EC EMC Directive 89/336/EC
EC Low-Voltage Directive 73/23/EC

I.3 Abbreviations used in this manual

Abbrev.	Meaning	Abbrev.	Meaning
A	Array	LSB	Least significant bit
AG	Automation unit (PLC)	LWL	Optical fibre
AK	Order / response ID	MB	Marker byte (flag byte)
AW	Output word	MSB	Most significant bit
AWL	Instruction list	MW	Marker word (flag word)
Bcc	Checksum	OB	Organisation block
DB	Data block	PB	Program block
DBND	Net data block	PKE	Parameter ID with PPO
DP	Distributed peripheral	PKW	Parameter ID-value
DPR	Dual-ported RAM	PNU	Parameter number
DW	Double word	PPO	Parameter process data object
EW	Input word	PWE	Parameter value
FB	Function block	PZD	Process data
HIW	Main actual value	QVZ	Acknowledgement delay
HSW	Main setpoint	SPM	Spontaneous message
IND	Index with PPO	CW	Control word
IR	Information report	SW	Status word

I.4 Features of PROFIBUS CONNECT

The following functions are available in connection with the position control which is integrated into the digifas™ 7100/7200 servo controller:

Setup and general functions:
- transmit a control word from the PLC to digifas™
- transmit a status word from digifas™ to the PLC
- jogging with variable speed
- go to reference point
- set reference point

Positioning functions:
- transmission and execution of a direct order
- start a motion task from the motion block storage of the digifas™

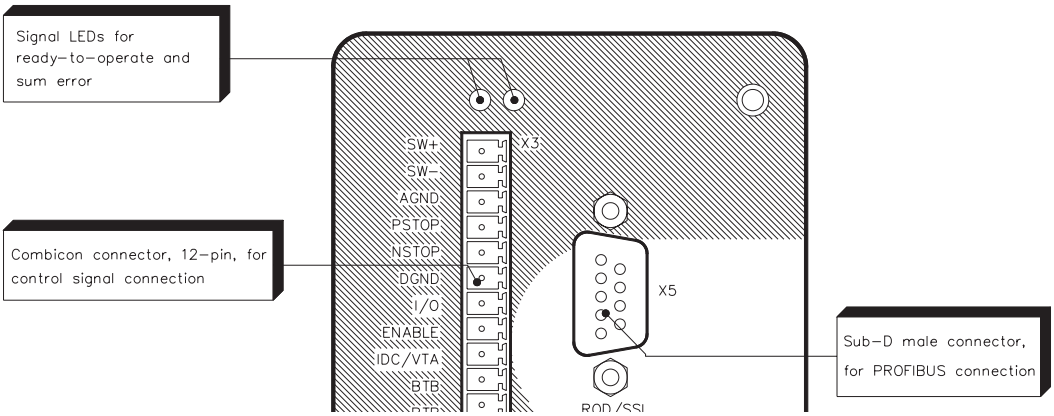
Data transfer functions:
- transmit a motion task from the PLC to the motion block storage of the digifas™
- read a motion task from the motion block storage of the digifas™ into the PLC
- read the actual position, actual values, error register

System requirements:
- digifas™ 7100/7200 with a PROFIBUS CONNECT interface
- Masterstation (PLC, PC...) with a PROFIBUS DP interface

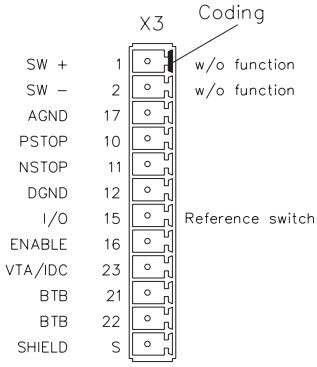
Transmission procedure:
- PROFIBUS DP to EN 50170
- communication profile to VDI/VDE 3689 (variable-speed drives)
- digifas™-specific instrument profile with PPO Type 1

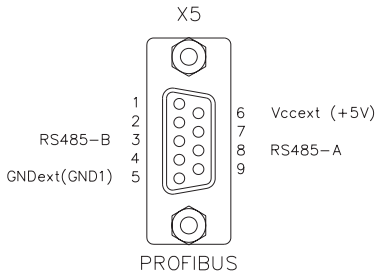
Bus medium:
- two-wire lead (twisted pair, shielded, characteristic impedance 160Ω), optional redundant
- transmission rate: max 1.5 Mbit/s

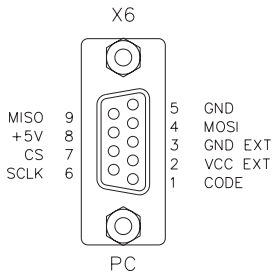
I.5 Frontal view digifas™ 7100/7200 with PROFIBUS CONNECT

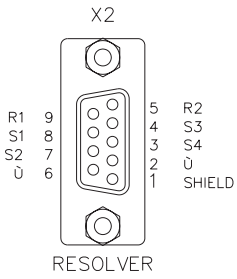


I.6 Pin assignments digifas™ 7100/7200 with PROFIBUS CONNECT









I.7 Numerical format

Not only parameter numbers, but also parameter values (INTEGER,FLOAT) are expected to be in the **Motorola-Format** (see below). The FLOAT format which is used conforms to the IEEE-754 standard format (32-bit) and has 24-bit precision.

INTEGER16	address n+1:	bit	7 ... 0 (LSB)
	address n+0:	bit	(MSB) 15 ... 8
INTEGER32	address n+3:	bit	7 ... 0 (LSB)
	address n+2:	bit	15 ... 8
	address n+1:	bit	23 ... 16
	address n+0:	bit	(MSB) 31 ... 24
FLOAT	address n+3:	bit	7 ... 0 (MMMM MMMM),
	address n+2:	bit	15 ... 8 (MMMM MMMM),
	address n+1:	bit	23 ... 16 (EMMM MMMM),
	address n+0:	bit	31 ... 24 (SEEE EEEE)

Interpretation:

n address (absolute)
M 23-bit normalised mantissa, the most significant bit is always ‘1’ and is therefore not stored
E exponent (2’s complement) with offset 127 (decimal)
S sign bit; 1 = negative, 0 = positive

Negative numbers are displayed as 2 ’s complement.

I.8 Bus cable

The maximum usable cable length for reliable communication decreases as the transmission rate increases. We have established the following measured values as a guide, they should not be interpreted as limit values:

Cable data:	characteristic impedance	135-165 Ω (3...20 MHz)
	cable capacitance	max. 30 nF/km
	lead resistance (loop)	110 Ω/km
	core diameter	> 0.64 mm
	core cross-section	> 0,34 mm²

The SubD connector housing must fulfill the following requirements for EMC:

— metal or metallic-coated housing
— provision for attaching the cable shielding to the housing, with a large contact area
— HF-quality contact between the SubD connector and the connector housing

II Installation / Commissioning

II.1 Hard- and Software installation for connection to SIMATIC S5 / IM308B(C)

Make sure that the switchgear cabinet is safely isolated (lock-out, warning signs etc.). The individual supply voltages will only be switched on again during commissioning.

Residual charges in the capacitors can still result in dangerous voltages several minutes after switching off the mains supply power. Measure the voltage in the DC-link circuit, and wait until it has fallen below 40V.

Even if the motor is not rotating, power and control leads may still be live.

Electronic equipment is basically not failure-proof. The user is responsible for ensuring that, in the event of a failure of the servoamplifier, the drive will be set to a state which is safe for both machinery and personnel, for instance with the aid of a mechanical brake.

Drives with servo-amplifiers and CONNECT modules are remote-controlled machines. They can start to move at any time without previous warning. Make sure that the operating and service personnel is aware of this danger.

Take appropriate protective measures to ensure that an unintended start of the machine cannot result in dangerous situations for personnel or machinery. Software limit switches are *not* a substitute for the hardware limit switches of the machine.

Install the servo-amplifier as described in the installation manuals for digifas™ 7100 or digifas™ 7200 as appropriate.
The wiring for the analog setpoint input and the positioning interface, in accordance with the wiring diagram in the installation manual, is not needed.

From Section II.1.2 of this manual, use the wiring diagram which is valid for the amplifier which is selected to connect up the CONNECT interface.

Always make sure that the equipment is de-energized during installation and wiring. Do not switch on the mains power supply, the 25V auxiliary supply, or the operating voltage of any other equipment which is to be connected.

Never break any of the electrical connections to the servo-amplifer while any equipment is live. This could result in destruction of the electronics.

Because of the internal representation of the position control parameters, the position control can only be operated if the final limit speed of the drive is not more than 6000 r.p.m. All the information about resolution, step size, positioning accuracy etc., refers to calculatory values.
Non-linearities in the mechanism (backlash, flexing etc.) are not taken into account.

If the final limit speed of the motor has to be altered, then all the parameters which were previously entered for positioning control and motion blocks must be adjusted.

II.1.1 Connection methods

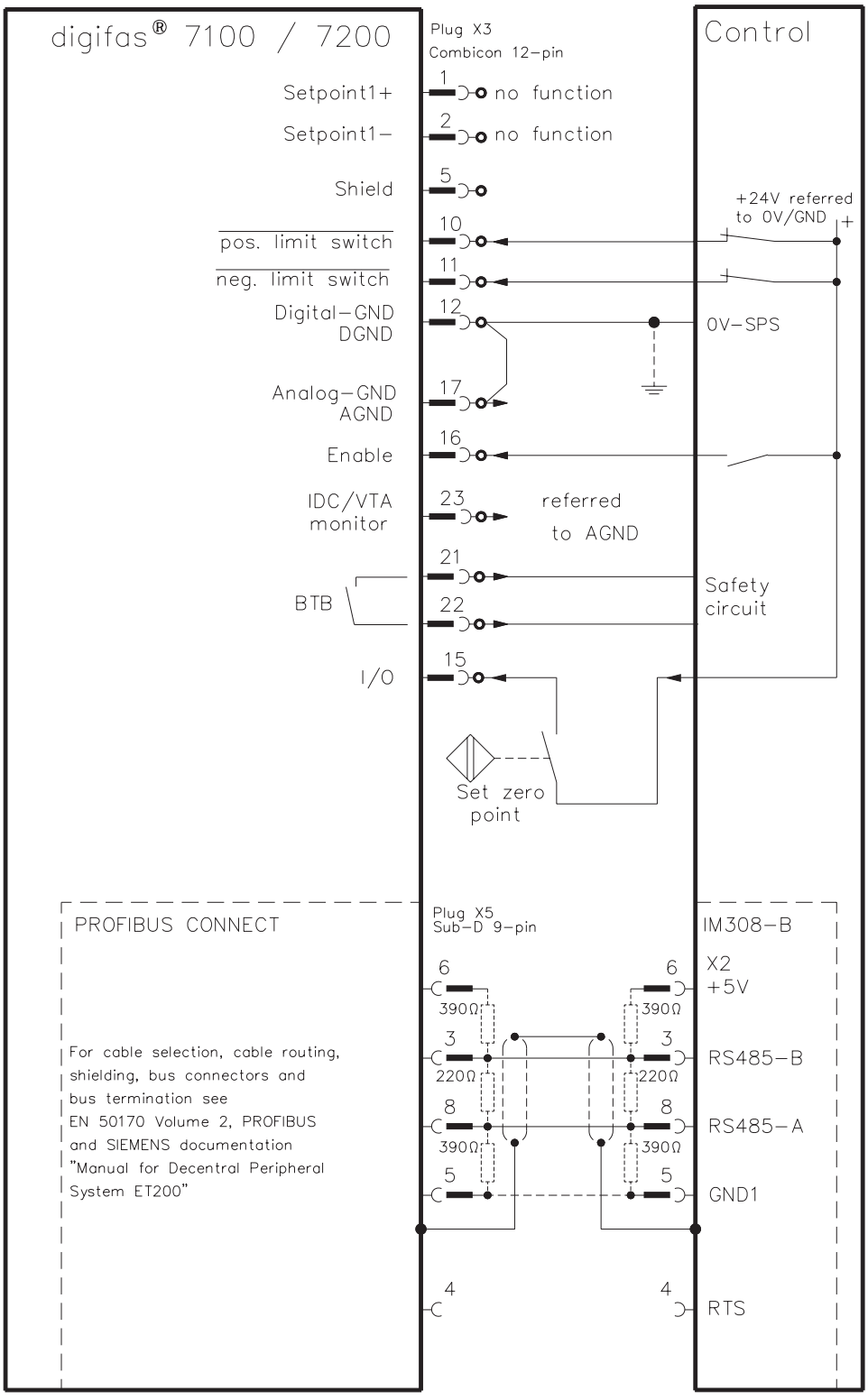
The selection of cables, cable layout, shielding, bus connector, bus termination and propagation times are decribed in the following documents:

PROFIBUS interface:
SIEMENS documentation “Manual for Decentral Peripheral System ET200"

Mains connection, motor connection, digital control cables:
Installation manual for digifas™ 7100 or digifas™ 7200

II.1.2 Connection diagram digifas™ 7100-L2/DP / digifas™ 7200-L2/DP

For motor connection and power feed see the installation instructions in the appropriate amplifier manual



II.1.3 Parameterization of the master interface modules

II.1.3.1 Interface module IM308B(C)

A system requirement is the interface module IM308B, implementation level 5 (DP Standard) or IM308Cbuilt into the SIMATIC S5. Parameterization is performed with the aid of the software tool COM ET200, Version 4.0 or higher.

If the IM308C interface module is used, then the DIAGNOSE parameter must be set to NO when the parameters are set up !

The procedure is as for the configuration described in the manual “Decentral Periphal System ET200”, Chap. 8, of a slave station with the following particulars:

a) **Mask : ET200 - SYSTEM PARAMETER**
bus profile : select DP Standard
response monitoring : yes

b) **Mask : CONFIGURATION**
Station type : Select DIGIFAS_1SPC. In order to do this, the Type file SPCDI1TD.200 must previously be loaded into the same directory as the COM ET200 Software.
The file SPCDI1TD.200 can be found in the directory ET200 on the function-block diskette DIGI_DP1.
Slot 0 : 4 AX = select 4-word E/A with consistency
Slot 1 : 2 AX = select 2-word E/A with consistency

An example of parameterization can be found in the appendix (Section VI.3)

II.1.3.2 Interface module CP5412 (A2)

The CP5412 (A2) is a plug-in card from Siemens for IBM-AT-compatible PCs. Project-planning for DP-master operation is performed with the aid of the software tool COML DP.
Use our master data file DIGIFAS.GSD for project-planning .
Set the data consistency to match the IM308B. A base data file in ASCII format (DIGIFAS.TXT) serves as an example for project planning. This base data file can be converted into a binary file with the aid of COML DP.
Further information can be found in the Siemens manual “Project-planning of DP-master operation with COML DP”.

II.1.4 PLC programming

II.1.4.1 Standard function blocks for data exchange with digifas™

This uses the Siemens function-block package DVA_S5, Version 2.0 or higher.
The following function blocks (FB) are absolutely necessary for each drive:
FB21, DPxxx-S as Transmit block **FB22**, DPxxx-E as Receive block
Also required are:

a) A data block (DB) “User data” (DBND), length 1 + 21 times the no. of drives
Up to 12 drives (1+21x12=253 words) can be put together into one DB.

b) A data block “Parameter sets”, length 4 + 5 times the no. of drives

Max. length of the data block: 256 words. PPO-type 1 is used exclusively.
FB21 and FB22 handle the data exchange and tracking of PKW-jobs between the DPR of the IM308 and the transmission/reception register in the DBND. The FB21 and the FB22 must be polled absolutely at least once, either cyclically or time-controlled. You can use the FB23 for the setting up and preliminary loading of the data blocks if you wish. The technical data and details on this application can be found in the Siemens manual.

II.1.4.2 SEIDEL handling package DIGI_DP1

This function-block package contains a number of function blocks which make it possible to handle all the control functions of the digifas™ in a simple fashion:

FB190	DIGISTAT	status information on digifas™ and communication, transmission CW bit 0-7
FB191	DIGIHAND	manual control of digifas™. Jogging, go to /set reference
FB195	DIGIDIRE	deposit and start a direct job with digifas™
FB196	DIGISATZ	call up and start a motion block
FB197	DIGILADE	load a motion block into the digifas™ motion-block store
FB198	DIGILESE	read out a motion block from the digifas™ motion-block store
FB199	DIGI-IST	switch on actual-value transmission and error-register reading
FB200	DIGI-PARA	parameterizing the servo-amplifier

The blocks take over the flow to the transmission register in DBND, take the data and status signals from the reception register and carry out the transmission of motion blocks independently, in successive telegrams. The user is relieved from the task of producing his own FBs.
The call is made from OB1, in a PB or from a FB. Apart from the FB190, it is only necessary to call the FBs which are required for the specific function. The others can be left out.
A description of the individual FBs can be found on the diskette, as a commentary to the sample program DIGIFBST.S5D (OB1), and in the appendix to this manual.

II.1.5 SINEC DP programming interface

The SINEC programming interface provides the software interface between the CP5412 (A2) and the application .
Seidel has a function-block library to make it easier to interface the application to the CP:
“Function-block library for PROFIBUS-DP master connection to SINEC CP 5412 (A2)”,
Mat.No.: 85578.
This library is available as source code in the high-level language C.

II.2 Commissioning

II.2.1 Commissioning the basic functions of the servo-amplifier

Only properly qualified personnel with professional expertise in control techology and drive technology are permitted to commission the servo-amplifier.

Check assembly + installation

Check that all the safety notes, which are included both in the installation manual for the servo-amplifier and in this manual, have been observed and implemented.

Connect PC, start BS7200

Use the operator software BS7200 for setting the parameters of the servo-amplifier. Set the communication to “Control from PC” (on the CONNECT menu page of the operator software).

Caution !
Make sure that any unintentional movement of the drive does not result in a hazard to personnel or machinery.

Commission basic functions

Now commission the basic functions of the servo-amplifier, and optimize the current and speed controllers. This part of commissioning is described more precisely in the installation and commissioning instructions for the servo-amplifier which has been selected.

Save parameters

When the optimization is finished, save the controller parameters in the servo-amplifier.

Test the bus connection

Remove the Enable signal (terminal X3.16), and switch off the power supply for the servo-amplifier.
The 24V DC auxiliary voltage remains switched on.

Test the communication

Test the installation of the ET200 bus connection and the IM308B interface, as described in the manual for the Siemens ET200 peripheral system.
Check the parameter setting for the ET200 and the station configuration.
Check the parameter setting for the interface module (see Section II.1.3).
Test the PLC-user program and check the parameter setting of the function-block packages DVA_S5 and DIGI_DP1.

Commission the position controller

Use the BS7200 operator software to switch the servo-amplifier over again to “Control from PC”.
Commission the position controller, as described in the following Sections II.2.2 (linear axis) and II.2.3 (rotary axis).

II.2.2 Linear axis

The parameter values given below for the commissioning of the position controller are not appropriate for all machines or may be dangerous in some machines.
It is therefore vital that you check the values. If you have to alter values, you must take into account that this is, for the moment, just a functional test.
Set up values which are abolutely safe, and which in no circumstances can lead to damage to the machine. Read Chapter V as well !

II.2.2.1 Commissioning the position controller for a linear axis

1. — Select the menu page CONNECT in the operator software and set the parameters for the position controller as follows (after checking whether the machine allows these values):

Parameter	Setting	Parameter	Setting
Kp	0.5 ... 1	t_emergency	min. braking time permitted by the machine
Ff	1	Go to Reference	depends on the application
control by	PC	zero-point offset	0
axis type	linear	limit-sw. 2	70% of the permitted distance, calculated from the zero point
count direction	depends on the application	limit-sw. 1	30% of the permitted distance, calculated from the zero point
resolution	distance moved / motor turn	InPosition	a window which is larger than required by the application
v_max	less than 50% of the maximum speed under load	contouring error (lag error)	a window which is larger than required by the application
t_accel_min	twice the min acceleration time permitted by the machine		

2.— Set the station address and the response monitoring

3.— Save the parameters which have been set in the EEPROM of the amplifier (menu page “Management”, save to EEPROM)

4.— Now set to “control by BUS”

5.— Switch off the 25V supply of the controller, and then switch it on again.
The new station address is first accepted at this moment.

6.— Switch on the main power supply, switch on the Enable signal for the amplifier (confirmation key)

7. — Set the reference point, or carry out a “Go to Reference” (reference traverse). Check that the load is at the reference point.

8.— Optimize the control-loop behaviour (see Section II.2.2.2)

9. — Finally, set the following parameters according to the application :
software limit-switch 1 and 2, window for contouring error, InPosition window, zero-point offset, t_acc_min, v_max, t_emergency

II.2.2.2 Notes on optimizing the linear axis

The current, speed and position controllers function as classic cascade controllers. The inner speed controller must be correctly set up, i.e. stiff, before beginning the optimization of the position controller.

1. — Traverse the drive, as a direct order, between two points at a low speed.
2. — Alter the Ff factor until the contouring error display (actual value display on the Connect page) is a minimum during acceleration.

User note:
With a positive direction of rotation the contouring error should be positive, since the drive is then lagging slightly (increase Ff). In the case of a negative contouring error (reduce Ff) the drive is leading the setpoint (oversteering).
The corresponding relationship is valid for a negative direction of rotation.

3. — Repeat points 1 and 2 in several steps with altered speed (v_setp) and altered acceleration/braking times, until the required load speed and braking/acceleration response is achieved.
   Depending on the driven mass, it is possible that the required load speed cannot be achieved with the Ff factor alone. In this case the Kp factor must be increased slightly.
4. — The Kp factor is increased until the controller just starts to oscillate, and then slightly reduced. With the aid of an oscilloscope, the transient response during acceleration can be observed at the speed monitor (VTA) of the controller and the Kp factor can be adjusted if necessary.

Warning:
If the final limit speed of the motor has to be changed, then all the position control and motion block parameters which were entered must be adjusted!
If Irms and / or Ipeak are changed after the position controller has been optimized, then Kp and Ff must be adjusted !

II.2.3 Rotary axis

The parameter values given below for the commissioning of the position controller are not appropriate for all machines or may be dangerous in some machines.
It is therefore vital that you check the values. If you have to alter values, you must take into account that this is, for the moment, just a functional test.
Set up values which are abolutely safe, and which in no circumstances can lead to damage to the machine. Read Chapter V as well !

II.2.3.1 Commissioning the position controller for a rotary axis

1. — Select the menu page CONNECT in the operator software and set the parameters for the position controller as follows (after checking whether the machine allows these values):

Parameter	Setting	Parameter	Setting
Kp	0.5 ... 1	t_emergency	min. braking time permitted by the machine
Ff	1	eference traverse	according to the application
control by	PC	zero-point offset	0
axis type	rotary	limit-sw. 2	not required
count direction	according to the application	limit-sw. 1	not required
resolution	distance moved / motor turn	InPosition	a window which is larger than required by the application
v_max	less than 50% of the maximum speed under load	contouring error	a window which is larger than required by the application
t_acc_min	twice the min. acceleration time permitted by the machine		

2.— Set the station address and the response monitoring

3.— Save the parameters which have been set in the EEPROM of the amplifier (menu page “Management” : Save to EEPROM)

4.— Now set “control by BUS”

5.— Switch off the 25V supply of the controller, and then switch it on again.
The new station address is first accepted at this moment.

6.— Switch on the main power supply, switch on the Enable signal for the amplifier (confirmation key)

7. — Set the reference point, or carry out a “Go to Reference” (reference traverse). Check that the load is at the reference point.

8.— Optimize the control-loop behaviour (see Section II.2.3.2)

9. — Finally, set the following parameters according to the application :
window for contouring error, In Position window,
zero-point offset, t_accel_min, v_max, t_emergency

II.2.3.2 Notes on optimizing the rotary axis

The current, speed and position controllers function as classic cascade controllers. The inner speed controller must be correctly set up, i.e. stiff, before beginning the optimization of the position controller.

1. — Traverse the drive, as a direct order, between two points at a low speed.
2. — Alter the Ff factor until the contouring error display (actual value display on the Connect page) is a minimum during acceleration.

User note:
With a positive direction of rotation the contouring error should be positive, since the drive is then lagging slightly (increase Ff). With a negative contouring error (reduce Ff) the drive is leading the setpoint (oversteering).
The corresponding relationship is valid for a negative direction of rotation.

3. — repeat points 1 und 2 in several steps with altered speed (v_setp) and altered acceleration/braking times, until the required load speed and braking/acceleration response is achieved.
   Depending on the mass to be driven, it is possible that the required load speed cannot be achieved with the Ff factor alone. In this case the Kp factor must be increased slightly.
4. — The Kp factor is increased until the amplifier just starts to oscillate, and then slightly reduced. With the aid of an oscilloscope, the transient reponse during acceleration can be observed at the speed monitor (VTA) of the controller and the Kp factor can be adjusted if necessary.

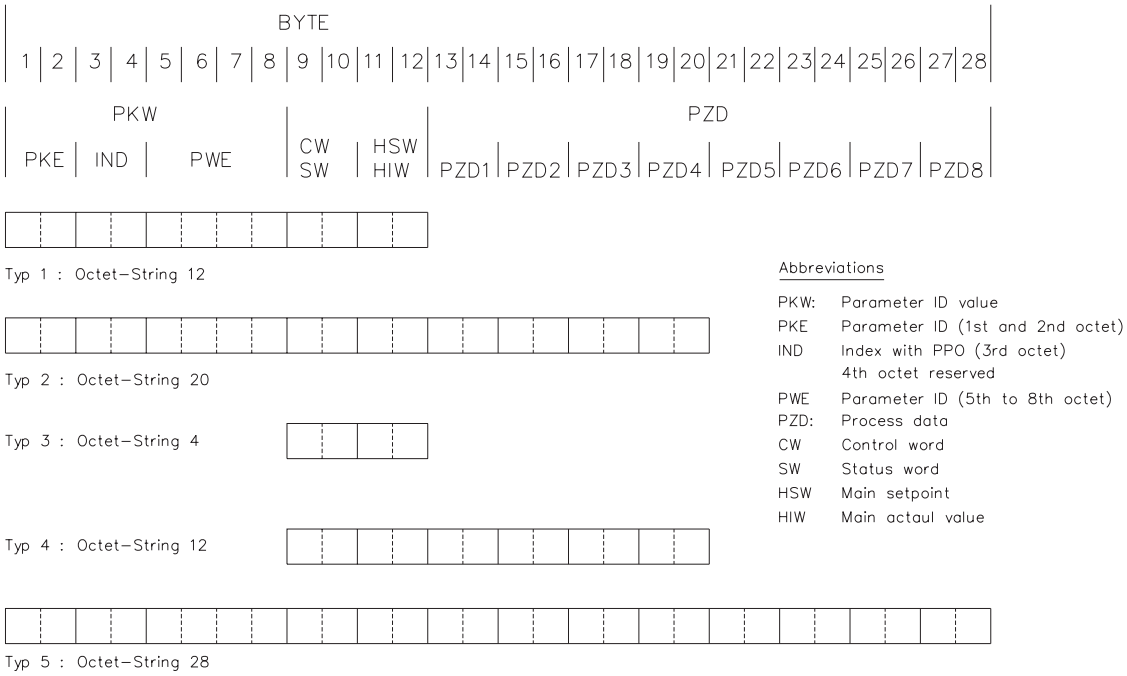
Warning :
If the final limit speed of the motor has to be changed, then all the position control and motion block parameters which were entered must be adjusted!
If Irms and / or Ipeak are changed after the position controller has been optimized, then Kp and Ff must be adjusted !

This page is deliberately left blank.

III Telegram structure

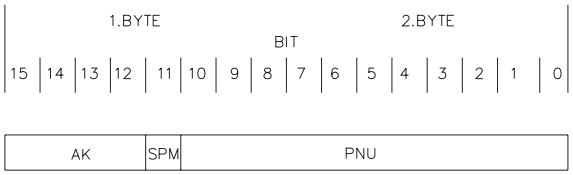
III.1 Parameter process data object (PPO) to PROFIBUS-profile VDI/VDE 3689

MSB flush left, LSB flush right



III.2 Specific digifas™ instrument profile

III.2.1 Parameter ID with PPO (PKE)



Abbreviations

AK	Order / response ID
SPM	Toggle-Bit für spontaneous message (not implemented at present)
PNU	Parameter number

Bold/italic printed lines in the tables are valid for digifas™ 7100/7200

Master —> Slave		Slave —> Master			Slave —> Master	
Task ID	Function	Re-sponse ID pos.	Re-sponse ID neg.		Re-sponse ID	Function
0	***no task***	***0***	***0***		***0***	***no task***
1	***request parameter value***	1, ***2***	***7***		1	transmit parameter value [W]
2	change parameter value [W]	1	7/8		***2***	***transmit parameter value [DW]***
3	***change parameter value [DW]***	***2***	***7***/8		3	transmit descriptive element
4	request descriptive element	3	7		4	transmit parameter value [A/W]
5	change descriptive element	3	7/8		5	transmit parameter value [A/DW]
6	request parameter value [A]	4, 5	7		6	transmit no. of array elements
7	change parameter value [A/W]	4	7/8		***7***	***task impossible (with error no.)***
8	change parameter value [A/DW]	5	7/8		8	no operator access for PKW interface
9	request no. of array elements	6	7		9	spontaneous message [W]
10	reserved				10	spontaneous message [DW]
11	reserved				11	spontaneous message [A/W]
12	reserved				12	spontaneous message [A/DW]
13	reserved				13	reserved
14	reserved				14	reserved
15	reserved				15	reserved

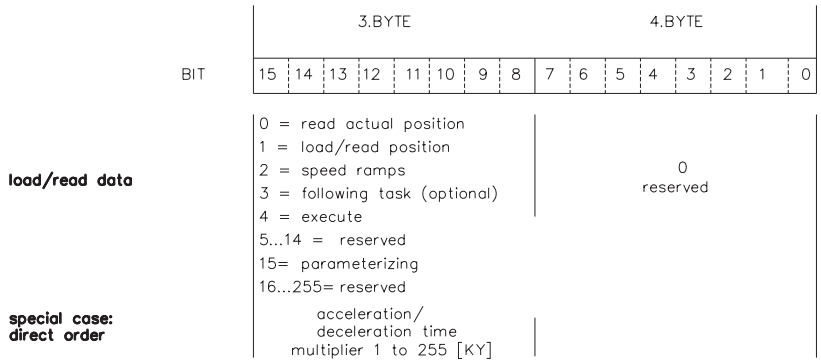
Profile-specific error nos. for response ID 7

If tasks cannot be performed, then the slave transmits the response ID 7 signifying “task impossible” and transfers the appropriate error no. in PWE (parameter value) byte 7-8 .

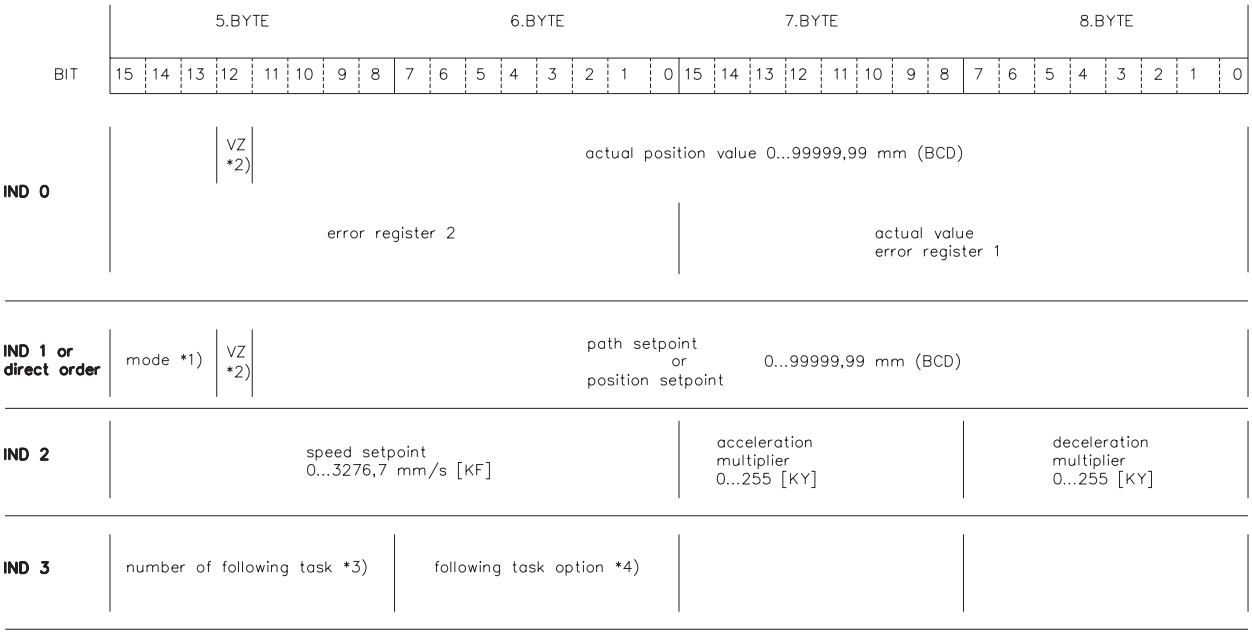
Error no.	Function
0	***invalid PNU***
1	***parameter value cannot be altered***
2	***passed lower or upper value limit***
3	***erroneous subindex (=index)***
4	no array
5	wrong data type
6	“set” not permitted (only resettable)
7	descriptive element cannot be changed
8	PPO-write requested in IR : not available
9	description data not available
10	access group
11	***no operator access***
12	key word missing
13	text not legible in cylical operation
14	name not legible in cyclical operation
15	no text array available
16	PPO-write is missing

Error no.	Function
17	***operating status prevents task***
18	***other errors***
19	date not legible in cyclical operation
20	***invalid command, axis not normalised***
21	**no task available**
22	**Bcc error in the motion task**
23	***response monitoring active***
24	—
25	***exceeded value limit***
26	***parameter value cannot be displayed internally***
27	***command can only be performed in disabled state***
28	***no EEPROM access possible***
29	***conversion caused by a change of resolution is in progress***
30	***impermissible number for next task***
31	command cannot be performed in this mode
32	ANALOG mode not possible
:	
100	

III.2.2 Subindex IND



III.2.3 Parameter value PWE



*1)

Bit 15	Bit 14	Bit 13	type
0	0	0	ABS
0	0	1	–
0	1	0	REL
0	1	1	REL1
1	0	0	–
1	0	1	–
1	1	0	REL2
1	1	1	–

see Section V.4.2

*2)
Bit 12 1=negative
0=positive

*3)
0=no following task
1...120 number of following task

*4)
Bit 2 1=no braking to target position
0=braking to target position

see Section V.4.8

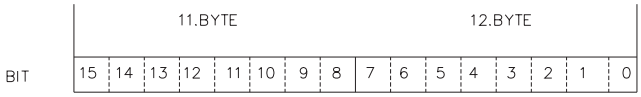
III.2.4 Control word CW digifas™ profile

Bit	Value	Meaning to VDI/VDE 3689	Meaning for digifas™ profile
0	1	ON	no significance
	0	OFF	no significance
1	1	operating condition	controller enable - AND combination with the ext. enable signal
	0	OFF 2	controller inhibit - AND combination with the ext. enable signal
2	1	operating condition	enable
	0	OFF 3	rapid stop : drive brakes with maximum deceleration to n = 0
3	1	operation enabled	if configured: free brake (not implemented)
	0	operation inhibited	if configured: do not free brake (not implemented)
4	1	operating condition	no significance
	0	inhibit ramp-up signal	no significance
5	1	stop ramp-up signal	no significance
	0	enable ramp-up signal	no significance
6	1	enable setpoint	enable motion task
	0	inhibit setpoint	pause
7	1	acknowledge	acknowledge contouring error/response monitoring (only when amplifier is inhibited/disabled)
	0	no significance	“acknowledge contouring error/response monitoring” inactive
8	1	jog 1 ON	jog 1 ON
	0	jog 1 OFF	jog 1 OFF
9	1	jog 2 ON	jog 2 ON
	0	jog 2 OFF	jog 2 OFF
10	1	control from AG	start motion task
	0	no control	stop motion task
11	1	no significance	start reference traverse, sets SW bit 12 to 0
	0	no significance	stop reference traverse
12	1	no significance	sets reference point
	0	no significance	“set reference point” inactive
13	1	no significance	no significance
	0	no significance	no significance
14	1	no significance	no significance
	0	no significance	no significance
15	1	no significance	no significance
	0	no significance	no significance

III.2.5 Status word SW digifas™ profile

Bit	Value	Meaning to VDI/VDE 3689	Meaning for digifas™ profile
0	1	ready to switch on	ready to switch on - status BTB-contact
	0	not ready to switch on	not ready to switch on - status BTB-contact
1	1	ready to switch on	controller enabled - AND combination with the ext. enable signal
	0	not ready to switch on	controller inhibited - AND combination with the ext. enable signal
2	1	operation enabled	n = 0 - drive not running
	0	operation inhibited	n ≠ 0 - drive running
3	1	fault	fault (see error register assignment IV.1.3.3)
	0	no fault	fault free
4	1	no OFF 2	signal from software limit switch 1
	0	OFF 2	no signal from software limit switch 1
5	1	no OFF 3	signal from software limit switch 2
	0	OFF 3	no signal from software limit switch 2
6	1	switch-on disabled	signal from hardware limit switch
	0	no switch-on disable	no signal from hardware limit switch
7	1	warning	warning: I^2t-limiting has responded
	0	no warning	no warning
8	1	SETPOINT/ACTUAL monitoring is within tolerance range	no contouring error - position setpoint/actual comparison is OK, allowing for the contouring-error window
	0	SETPOINT/ACTUAL monitoring is not within tolerance range	contouring error - position setpoint/actual comparison is not OK, allowing for the contouring-error window
9	1	control requested	extern control is enabled - PLC has access to all functions
	0	local operation	local operation - PLC has read-only access
10	1	reached f or n	in-position - drive stands at the setpoint position, allowing for the in-position window
	0	not reached f or n	drive is not at the setpoint position
11	1	no significance	drive is at the reference point, allowing for the in-position window
	0	no significance	drive is not at the reference point, allowing for the in-position window
12	1	no significance	reference is set (condition for position operation)
	0	no significance	no reference is set
13	1	no significance	“v” or “a” has been limited by the controller
	0	no significance	“v” or “a” has not been limited by the controller
14	1	no significance	traverse function activated
	0	no significance	no traverse function activated
15	1	no significance	error : bit command not executable because of operating status
	0	no significance	no error

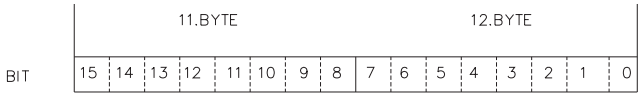
III.2.6 Main setpoint HSW



speed setpoint ± 3276,7 mm/s [KF]

Is used for reference traverse, direct order and jogging mode

III.2.7 Main actual value HIW



± 3276,7 mm/s [KF]

speed actual value

IV Description of the functions

IV.1 Movement functions

IV.1.1 Setup and general functions

IV.1.1.1 Cyclical transmission of a control word to the digifas™

In the cyclical exchange of telegrams, 16 control bits are sent with each telegram from the master station (PLC, CP5412 (A2)) to the digifas™ (see “Control word” Section III.2.4).

IV.1.1.2 Cyclical transmission of a status word from digifas™ to the master

In the cyclical exchange of telegrams, 16 status bits are sent with each telegram from the digifas™ to the master station (SPS, CP5412 (A2)) (see the table in Section III.2.5). It is also possible to make an active read-out of the status register in the digifas™, see Section IV.1.3.4 .

IV.1.1.3 Jog operation

Control bits 8 (Jog 1) and 9 (Jog 2) of the control word (CW) make it possible to run the drive in jog mode. The signed speed setpoint is taken from the main setpoint (HSW) (bytes 11 - 12) at the moment of starting. Jog 1 and 2 differ in that they have differing speed limit parameters in the digifas™.

Conditions:

rapid stop	CW	Bit 2 = 1	no contouring error	SW	Bit 8 = 1
standby	SW	Bit 0 = 1	external control	SW	Bit 9 = 1
controller enabled	SW	Bit 1 = 1	no motion function active	SW	Bit 14 = 0
fault	SW	Bit 3 = 0			

The drive runs as long as the appropriate bit (8/9) is 1.
This function is supported by the function-block package DIGI_DP1 through FB191 DIGIHAND or through the function-block library with the function **dp_tippen**

IV.1.1.4 Set reference point

When control bit 12 = 1 the current position is defined as the reference point. The status word bit 12 = 1, reference point set, is returned. The positioning functions are enabled.
The zero-point offset (NI Offset) is ineffective.

Conditions:

controller enabled	SW	bit 1 = 1	external control	SW	bit 9 = 1
speed = 0	SW	bit 2 = 1	no motion function active	SW	bit 14 = 0

This function is supported by the function-block package DIGI_DP1 through FB191 DIGIHAND or through the function-block library with the function **dp_ctrl_word** in combination with the function **dp_send_stw**.

Warning:
Take care that the position of the reference point permits the subsequent positioning operations. The software limit-switches which are parameterized in the digifas™ may be ineffective. The axis may travel up to the hardware limit-switch or the mechanical stop. There is a risk of damage.

IV.1.1.5 Reference traverse (Go to Reference)

The reference traverse is started by CW, bit 11 = 1. The start of the reference traverse is recognized by the positive edge of bit 11.
If bit 11 is reset to 0 before the reference point is reached (SW return message, bit 12 = 1), then the reference traverse will be cancelled.
A set reference point is a precondition for all positioning functions.
The reference point switch is connected to the I/O input (terminal X3/15) of the digifas™.
Depending on the type of reference traverse, you can shift the zero point of the motor shaft at will within a turn, and assign a reference offset to it, by using the parameter “Zero-point offset”.
The reference offset and the types of reference traverse are described in Chapter V.
After the reference traverse, the drive signals “InPosition” and thus enables the position controller.
The speed for the reference traverse is transmitted with the main setpoint, and is limited to 10% of the v_max (PNU 1076) which has been set. The sign is not evaluated.

A reference traverse must be made after the 25V auxiliary supply voltage has been switched on.
Take care that the position of the zero point for the machine (= reference point) permits the subsequent positioning operations. The software limit-switches which have been parameterized in the digifas™ may be ineffective. The axis could drive onto the hardware limit-switches or even the mechanical stop. There is a danger of damage.
If the reference point (machine zero point) is approached too rapidly and/or the moment of inertia is high enough, then the axis may overshoot this point and, in unfavourable cases, drive onto the hardware limit-switches or the mechanical stop, with the risk of damage .

Conditions:

rapid stop	CW	Bit 2 = 1	no contouring error	SW	Bit 8 = 1
standby	SW	Bit 0 = 1	external control	SW	Bit 9 = 1
controller enabled	SW	Bit 1 = 1	no motion function active	SW	Bit 14 = 0
fault	SW	Bit 3 = 0			

This function is supported by the function-block package DIGI_DP1 through FB191 DIGIHAND or through the function-block library with the function **dp_ref_fahrt**.

IV.2 Positioning functions

The following positioning functions can only be executed when the axis has been normalized (IV.1.1.4 or IV.1.1.5), otherwise the task will be rejected with error-no. 20.
The data are transmitted with the PKW-mechanism (PROFIBUS-profile VDI/VDE 3689).

IV.2.1 Direct order

In this kind of task, the telegram contains all the data which are necessary for the positioning procedure. The order-ID is fixed as task number 127 .

Telegram structure:

PKE/AK	3
PKE/PNU	127
IND	RAMP
PWE1 (Byte 5/6)	POSH
PWE2 (Byte 7/8)	POSL
CW/SW	control bits / status bits
HSW	VABS

where

POSH= Higher value portion of the position-path setpoint, interpreted as BCD.
bit 12 = sign 0 = +, 1 = —
bit 14 = mode: 1 = relative positioning, 0 = absolute positioning
if bit14=0 then bit 12 must = 0 (absolute positioning only for +)

POSL = Lower value portion of the position-path setpoint, interpreted as BCD.
Two decimal places.

VABS = traverse speed

RAMP= byte 3: Acceleration/Deceleration time multiplier (x 10ms)
Acceleration and deceleration are identical for a direct order

Conditions:

rapid stop	CW	Bit 2 = 1	motion task enable	SW	Bit 6 = 1
motion task START	CW	Bit 10 = 1	no contouring error	SW	Bit 8 = 1
standby	SW	Bit 0 = 1	external control	SW	Bit 9 = 1
controller enabled	SW	Bit 1 = 1	reference is set	SW	BIT12 = 1
fault	SW	Bit 3 = 0	no motion function active	SW	Bit 14 = 0

If the conditions are fulfilled, the order is executed immediately.
It can be interrupted by CW bit 6 = 0 (pause) and continued by CW bit 6 = 1.
If CW bit 10 = 0 (STOP) is set before the positioning task is completed (return message: InPosition CW bit 10 = 1), then the positioning task will be cancelled.

When the actual position has been reached and the positioning task is finished, then CW bit 10 = 1 (InPosition) is set. The “InPosition” remains available as long as the axis is within the “InPosition” window or until another task is started.

When identical relative motion tasks are started one after another, a STOP command must be inserted.
This is vital, to be able to distinguish between identical relative motion tasks.

If, in the amplifier, limit values which are parameterized for v_max / t_acc_min are exceeded / not achieved, then these values will be taken as the limits.
The limiting is signalled during the execution of the task by SW bit 13 = 1 .

This function is supported by the function-block packet DIGI_DP1 through FB 195 DIGIDIRE or by the function-block library with the function **dp_d_motion**.

IV.2.2 Starting a motion task from the digifas™ motion-task memory

In this kind of task, all the data which are necessary for the positioning task are in the motion-task memory of the digifas™. The parameter number specifies the motion-task memory in the digifas™.

Telegram structure:

PKE/AK	3
PKE/PNU	0, 121 to 128, >144 are not permitted 1 to 120 = motion-block memory no. (EEPROM, non-volatile) 129 to 144 = motion-block memory no. (RAM, volatile)
IND	4 = execute
PWE	no significance
CW/SW	control/status word
HSW	no significance

Conditions:

rapid stop	CW	Bit 2 = 1	fault	SW	Bit 3 = 0
motion task enable	CW	bit 6 = 1	no contouring error	SW	Bit 8 = 1
motion task START	CW	bit 10 = 1	external control	SW	Bit 9 = 1
standby	SW	Bit 0 = 1	reference is set	SW	Bit 12 = 1
controller enabled	SW	Bit 1 = 1	no motion function active	SW	Bit 14 = 0

If the conditions are fulfilled the order is executed immediately. It can be interrupted by CW bit 6 = 0 (pause) and continued by CW bit 6 = 1.
If CW bit 10 = 0 (STOP) is set before the positioning task is completed (return message: InPosition CW bit 10 = 1), then the positioning task will be cancelled.

When the actual position has been reached and the positioning task is finished, then CW bit 10 = 1 (InPosition) are set. The “InPosition” remains available as long as the axis is within the “InPosition” window or until another task is started.

Changing over from a current motion task to a new motion task :

You can change over from a motion task which is running to a new motion task. The start command intitiates the changeover.

The behaviour on a changeover depends on the type of ramp which has been set :

TRAPEZE : the new motion task starts immediately, at the present speed
SINE² : first the drive brakes down to zero speed,
and then the new motion task is started.

If identical relative motion tasks are started one after another, a STOP command must be inserted.
This is vital, to be able to distinguish between identical relative motion tasks.

If, in the amplifier, limit values which are parameterized for v_max / t_acc_min are exceeded / not achieved, then these values are the limits.
The limiting is reported during the execution of the task by SW bit13 = 1 .

This function is supported by the function-block packet DIGI_DP1 through FB 196 DIGISATZ or by the function-block library with the function **dp_start_fs_eeprom**.

IV.3 Data transmission functions

The following function data can be transfered between the PLC and the digifas™. Data transmission is also made with the aid of the PKW-mechanism. These functions are also available when the external control (BUS) in digifas™ is not enabled.

IV.3.1 Load data block

This function transfers motion blocks from the PLC to the digifas™.

Telegram structure:

PKE/AK	3	
PKE/PNU	0, 121 to 128,>144 are not permitted 1 to 120 = motion-block memory no. (EEPROM, non-volatile) 129 to 144 = motion-block memory no. (RAM, volatile)	
IND	1 = load position values 2 = load speed setpoint and ramps 3 = load next task + options	
PWE	path/position setpoint speed setpoint acceleration time multiplier deceleration time multiplier next motion task (byte 5) options (byte 6)	with IND = 1 with IND = 2 with IND = 2 with IND = 2 with IND = 3 (optional) with IND = 3 (optional)
HSW	no significance	

The transmission of a motion task is made by using 2 successive PKW tasks, with subindex 1 and 2. If the following task is also to be defined (optional), then the PKW task with subindex 3 must also be transmitted.
Please observe the sequence :

1.- motion task with a following task IND1, IND3, IND2
2.- motion task without following task IND1, IND2

Conditions:

Jog 1 : CW bit 8 = 0 Jog 2 : CW bit 9 = 0 Reference traverse : CW bit 11 = 0

After a “load data” task has been performed, the “read data block” function and the start of a motion task are inhibited for several milliseconds, because of the EEPROM write procedure. Any attempt to transfer a task during this period will be rejected with “task terminated on error”.
While motion tasks are running, the motion tasks 129 to 144 (in RAM) can be redefined and transmitted, but motion tasks 1 to 120 (in EEPROM) cannot.

Explanation of PWE bytes 5 / 6 with IND = 3

Byte 5 : Following tasks can be predefined under 1 to 120 (EEPROM) and 129 to 144 (RAM) . A “0” means that no following task has been defined. In this case, byte 6 will not be evaluated.

Byte 6 : If bit 2 = 0, the preset braking ramp is used to brake to the target position, before the next task is started.

If bit 2 = 1, there is no braking to the target position, but the current speed will be adjusted to the speed for the following task, using the appropriate ramp.
Bit 2 is only relevant if the ramp type is set to “trapeze” (trapezoid form).
If the ramp type is set to “sine²”, braking is always to the target position.

This function is supported by the function-block package DIGI_DP1, using FB 197 DIGILADE, or through the function-block library, using the function **dp_w_fahrsatz**.

IV.3.2 Read data block

This function transfers motion blocks from the digifas™ to the PLC.

Telegram structure:

	request	reply
PKE/AK	1	2
PKE/PNU	0, 121 to 128, >144 not permitted 1 to 120 (EEPROM, non-volatile) 129 to 144 (RAM, volatile)	motion-block memory no.
IND	1= read position values 2= read speed setpoint and ramp setpoints 3= read next task+options (byte 5+6)	1 = PWE contains position setpoints, only positive sign is allowed for absolute positioning. 2 = PWE contains speed and ramp setpoints
PWE	no significance	path/position setpoint with IND = 1 speed setpoint with IND = 2 acceleration time multiplier with IND = 2 deceleration time multiplier with IND = 2 next motion task (byte 5) with IND = 3 options (byte 6) with IND = 3
HSW	no significance	no significance

The transmission of a motion task is made by using 2 successive PKW tasks, with subindex 1 and 2. If the following task is also to be defined (optional), then the PKW task with subindex 3 must also be transmitted.
Please observe the sequence :

1.- motion task with following task IND1, IND3, IND2
2.- motion task without following task IND1, IND2

Conditions:

jog 1 : CW bit 8 = 0
jog 2 : CW bit 9 = 0
reference traverse : CW bit 11 = 0

This function is supported by the function-block packet DIGI_DP1 through FB 198 DIGILESE or by the function-block library with the function **dp_r_fahrsatz**.

IV.3.2.1 Read actual values

Cyclical actual-value request

This PKW task switches on the reading of an actual value. The actual value will now be transmitted with each cyclical telegram - until a new PKW task is transferred.

Telegram structure:

	request	reply
PKE/AK	1	2
PKE/PNU	actual value ID 1 = position 2 words BCD 3...6 not used 7 = speed 8 = contouring error 2 words BCD 9 = serial no. (amplifier) 2 words BCD 10= serial no. (interface) 2 words BCD 11...30 not used 31 = error register 2 words KM	as transmitted
IND	0 = read	0
PWE	no significance	actual value (data with length of 1 word placed in PWE bytes 7-8)
HSW	no significance	no significance

If the error register is required (read actual values PNU = 31), 2 error registers (each 16 bits) are transferred to PWE. The bit assignment is as follows (0 = no error, 1 = error):

Error register 1 (bytes 7/8)		Error register 2 (bytes 5/6)	
Bit 15	Motor cable fault (from version 7P20)	Bit 15	not used
Bit 14	Resolver error	Bit 14	not used
Bit 13	Aux. supply fault	Bit 13	not used
Bit 12	Output stage fault	Bit 12	not used
Bit 11	Brake fault	Bit 11	not used
Bit 10	Output stage mains fault	Bit 10	not used
Bit 9	Overvoltage	Bit 9	not used
Bit 8	Undervoltage	Bit 8	not used
Bit 7	Short to earth	Bit 7	not used
Bit 6	DPR-error	Bit 6	not used
Bit 5	EEPROM-error	Bit 5	not used
Bit 4	Ballast load exceeded	Bit 4	not used
Bit 3	Response monitoring active	Bit 3	not used
Bit 2	Motor temperature exceeded	Bit 2	not used
Bit 1	Ambient temperature exceeded	Bit 1	not used
Bit 0	Heat sink temperature exceeded	Bit 0	not used

This function is supported by the function-block packet DIGI_DP1 through FB 199 DIGIIST or by the function-block library with the function **dp_r_istwert**

Bits 0-2 and 6-15 in error register 1 result in a switch-off of the amplifier (red LED) and can only be cancelled by switching off the 25V auxiliary supply.
If bit 3 is set, then no commands can be executed until bit 7 is acknowledged by CW.

Actual-value request with parameter number

Telegram structure :

	request	response
PKE/AK	1	2 (7 in the event of an error)
PKE/PNU	actual value (PNU : see IV.2.3.2)	as transmitted
IND	15	15
PWE	no significance	actual value (data format : see IV.2.3.2)
HSW	no significance	no significance

IV.3.2.2 Read status register (PNU 1118)

This command is used for an active read-out of the current status information for the servo-amplifier. The status register has a size of 32 bits.

Telegram structure :

	request	response
PKE/AK	1	2 (7 in the event of a fault)
PKE/PNU	1118 (see table)	as transmitted
IND	15	15
PWE	no significance	32-bit status register (PWE 5 LSB)
HSW	no significance	no significance

Amplifier ID (bits 24-26, 28 of the status register)

digifas™ 71xx (bit 28=1)		digifas™ 72yy (bit 28=0)	
bit 26 | 25 | 24	rated current / A	bit 26 | 25 | 24	rated current / A
0 | 0 | 0	—	0 | 0 | 0	yy=01
0 | 0 | 1	xx=50	0 | 0 | 1	yy=02
0 | 1 | 0	xx=33	0 | 1 | 0	yy=04
0 | 1 | 1	xx=03	0 | 1 | 1	yy=06
1 | 0 | 0	xx=05	1 | 0 | 0	—
1 | 0 | 1	xx=08	1 | 0 | 1	—
1 | 1 | 0	xx=12	1 | 1 | 0	—
1 | 1 | 1	xx=16	1 | 1 | 1	—

Description of the status register (SR) for digifas™ :

Bit-no.	Status	Description
0	1	amplifier ready to switch on
	0	amplifier not ready
1	1	amplifier enabled (HW and SW-Bus)
	0	amplifier disabled
2	1	r.p.m. = 0
	0	r.p.m. ≠ 0
3	1	fault recognised (see fault register)
	0	no fault
4	1	software limit switch 1 activated
	0	software limit switch 1 not activated
5	1	software limit switch 2 activated
	0	software limit switch 2 not activated
6	1	hardware limit switch activated
	0	hardware limit switch not activated
7	1	warning : I^2t limit has been activated
	0	no warning
8	1	no contouring error
	0	contouring error
9	1	control from BUS
	0	control from PC
10	1	In set position
	0	not in set position
11	1	position is at reference point
	0	position is not at reference point
12	1	reference point is set
	0	reference point is not set
13	1	speed (or acceleration) limit is active
	0	speed (or acceleration) limit is not active
14	1	ballast load exceeded
	0	ballast load not exceeded
15	1	digital setpoint provision
	0	analog setpoint provision
16	1	quick stop active
	0	quick stop not active
17	1	brake applied
	0	brake released
18	1	pause active
	0	pause not active
19	1	jog mode active
	0	jog mode not active
20	1	motion task running
	0	no motion task in progress
21	1	reference traverse in progress
	0	no reference traverse in progress
22	1	EEPROM access denied
	0	EEPROM access permitted
23	1	amplifier enable is on (via BUS)
	0	amplifier enable is off
24-26	1	output stage ID
	0	output stage iD
27	1	with LCA
	0	without LCA
28	1	amplifier ID (digifas™ 7100)
	0	amplifier ID (digifas™ 7200)
29	1	no brake
	0	with brake
30	1	left hardware limit switch (evaluate only if bit 6 is set)
	0	right hardware limit switch (evaluate only if bit 6 is set)
31	1	data not converted
	0	data converted (resolution has been changed)

IV.4 Setting the amplifier parameters

IV.4.1 Data format

The data format is described in the following sections. All the parameter numbers (PNUs) are provided with an offset of 1000. Each command or command variable is recognised with the aid of the PNU. All PNUs which are used for setting the parameters for the servo-amplifier can only be reached by Subindex 15.

The data for the PNU-variables are placed flush right in the parameter value (PWE) :

1-byte data (byte)	PWE 8
2-byte data (word)	PWE 7-8 (PWE 8 LSB)
4-byte data (double-word)	PWE 5-8 (PWE 8 LSB)

If a command cannot be carried out, then the error is signalled by the response ID AK = 7 and an error number is produced. The error numbers are described in Section III.2.1 .
A more detailed description of the parameters can be found in the operating instructions for the operator software BS7200.

IV.4.2 Write / read an amplifier parameter

Write (AK = 3, IND = 15) or read (AK = 1, IND = 15) an amplifier parameter

To write or read an amplifier parameter, which is identified by the parameter number (PNU), to the *volatile* memory of the digifas™. The parameters which are stored in the digifas™ can be transferred to the *non-volatile* memory by using the command “Save amplifier parameters to EEPROM” (see also Section IV.2.4).

Telegram structure :

	request	response
PKE/AK	1 (read) / 3 (write)	2 / 7 on error
PKE/PNU	see table	as transmitted
IND	15	15
PWE	with AK = 3 data type : see IV.2.3.2 with AK = 1 data type has no meaning	with AK = 3 mirrored PWE of the request with AK = 1 data type : see IV.2.3.2
HSW	no significance	no significance

IV.4.3 Summary of the extended parameter numbers

All the extended parameter numbers are arranged numerically, with a brief description, in the table in Section IV.4.3.2 .
The parameters which are described can only be used with subindex 15 . The parameter value (PWE) is irrelevant for those parameter numbers for which no data type is given.

IV.4.3.1 Symbolic labels

Some of the PNUs is Section IV.4.3.2 are marked by * . These are symbolic labels, to which the parameter values in the table below are assigned.

		Parameter values										
Label	**PNU_D**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Analog/digital setpoint provision	1048	position	n analog	I analog	n digital	I digital	—	—	—	—	—	—
I/O	1055	reset	1:1	int. off	I²t	ballast	reference	Ipeak x%	ROD-SSI	mains_BTB	set/8	InPos
Brake	1056	none	with	mains_BTB	—	—	—	—	—	—	—	—
Synchronization correction.	1062	off	on	—	—	—	—	—	—	—	—	—
Limit-switch	1080	off	on	stop	—	—	—	—	—	—	—	—
DC monitor	1081	tacho	current	s_fehl	I_soll	—	—	—	—	—	—	—
No. of motor poles	1086	2	4	6	8	10	12	—	—	—	—	—
No. of resolver poles	1091	2	—	—	—	—	—	—	—	—	—	—
Language	1092	German	English	French	—	—	—	—	—	—	—	—
Count direction	1097	negative	positive	—	—	—	—	—	—	—	—	—
Ballast select	1098	internal	external	—	—	—	—	—	—	—	—	—
Axis type	1101	rotary	linear	—	—	—	—	—	—	—	—	—
Control from	1102	BUS	PC	—	—	—	—	—	—	—	—	—
Ramp type	1106	trapeze	sine²	—	—	—	—	—	—	—	—	—
Reference traverse	1107	1-	1+	2-	2+	3-	3+	4-	4+	5-	5+	—

IV.4.3.2 List of the parameter numbers

PNU_D	Data type	Access	Brief description	Documented in manual :
1000-1020	—	—	reserved	
1021	—	w/a	save amplifier parameter to EEPROM	PROFIBUS
1022-1030	—	—	reserved	
1031	32-Bit Int.	r	actual position, weighting 2	
1032	32-Bit Int.	r	actual speed, weighting 1	BS7200
1033	32-Bit Int.	r	actual contouring error, weighting 2	
1034	32-Bit Float	r	actual current	BS7200
1035	32-Bit Float	r	actual speed	BS7200
1036	16-Bit Int.	r	actual angular position	BS7200
1037	16-Bit Int.	r	actual heat sink temperature	BS7200
1038	16-Bit Int	r	actual ambient temperature	BS7200
1039	16-Bit Int.	r	actual DC-link voltage	BS7200
1040	16-Bit Int.	r	actual ballast power	BS7200
1041	16-Bit Int.	r	actual I^2t utilization	BS7200
1042	32-Bit Float	r	actual operating time	BS7200
1043	32-Bit Int.	r	current serial no. (Hardware)	
1044	4 Char	r	current software version : control	
1045	4 Char	r	current software version : PROFIBUS interface	
1046	4 Char	r	current amplifier ID	PROFIBUS
1047	32-Bit Int.	r	current interface no.	
1048*	8-Bit Char	r/w/a/d	setpoint provision mode	PROFIBUS
1049	16-Bit Int.	r/w	digital setpoint	PROFIBUS
1050	—	—	reserved	
1051	32-Bit Float	r/w	Irms - effective current	BS7200
1052	32-Bit Float	r/w	Ipeak - peak current (pulse current)	BS7200
1053	8-Bit Char	r/w	I^2t signal	BS7200
1054	—	—	reserved Ivor - current lead	
1055*	8-Bit Char	r/w	function of the I/O terminal	BS7200
1056*	8-Bit Char	r/w	brake selection	BS7200
1057	32-Bit Float	r/w	Kp gain factor of the current controller	BS7200
1058	32-Bit Float	r/w	Tn integral-action time of the current controller	BS7200
1059	—	—	reserved I^2t time	
1060	16-Bit Int.	r/w	K_E voltage constant of the motor	PROFIBUS
1061	—	—	reserve	
1062*	—	—	reserved	
1063	32-Bit Float	r/w	L motor inductance	PROFIBUS
1064	16-Bit Int.	r/w	Kp gain factor of the speed controller	BS7200
1065	32-Bit Float	r/w	Tn integral-action time of the speed controller	BS7200
1066	—	—	reserve	
1067	32-Bit Float	r/w	PID-T2 second time constant for speed controller	BS7200
1068	16-Bit Int.	r/w	final limit speed for tachometer feedback	BS7200
1069	16-Bit Int.	r/w	initial Phi	BS7200
1070	16-Bit Int.	r/w	final Phi	BS7200
1071	32-Bit Float	r/w	tachometer time constant	BS7200
1072	32-Bit Float	r/w	Kp gain factor of the position controller	PROFIBUS
1073	32-Bit Float	r/w	Ff feed-forward factor of the position controller	PROFIBUS
1074	16-Bit Int.	r/w	t-emergency max. braking deceleration	PROFIBUS
1075	16-Bit Int.	r/w	t-acc-min min. acceleration time	PROFIBUS
1076	32-Bit Float	r/w	v-max maximum speed	PROFIBUS
1077	32-Bit Float	r/w	contouring error	PROFIBUS
1078	32-Bit Float	r/w	In-Position window	PROFIBUS
1079	32-Bit Float	r/w/d	resolution (adjust increments to SI-units)	PROFIBUS

PNU_D	Data type	Access	Brief description	Documented in manual :
1080*	8-Bit Char	r/w	limit-switch selection on / off / stop	BS7200
1081*	8-Bit Char	r/w	DC monitor	BS7200
1082	32-Bit Float	r/w/d	resolution (adjust SI-units to increments)	PROFIBUS
1083	—	—	reserved	
1084	—	—	reserve	
1085	—	—	reserve	
1086*	8-Bit Char	r/w/d	no. of motor poles	BS7200
1087	—	—	reserve	
1088	—	—	reserve	
1089	—	—	reserved limit value : heat sink temperature	
1090	—	—	reserved limit value : ambient temperature	
1091*	8-Bit Char	r/w	no. of resolver poles fixed at 2	BS7200
1092*	8-Bit Char	r/w	language selection	BS7200
1093	32-Bit Float	r/w	setpoint offset	BS7200
1094	16-Bit Int.	r/w	rising setpoint ramp	BS7200
1095	16-Bit Int.	r/w	falling setpoint ramp	BS7200
1096	16-Bit Int.	r/w	ballast load	BS7200
1097*	8-Bit Char	r/w	count direction for position controller	PROFIBUS
1098*	8-Bit Char	r/w	select ballast internal/external	BS7200
1099	32-Bit Float	r/w	software limit-switch 1	PROFIBUS
1100	32-Bit Float	r/w	software limit-switch 2	PROFIBUS
1101*	8-Bit Char	r/w	axis type (linear / rotary)	PROFIBUS
1102*	8-Bit Char	r	control from	PROFIBUS
1103	16-Bit Int.	r/w	response monitoring	PROFIBUS
1104	32-Bit Float	r/w	zero-point offset	PROFIBUS
1105	—	—	reserved	
1106*	8-Bit Char	r/w	ramp type trapeze/sine²	PROFIBUS
1107*	8-Bit Char	r/w	type of reference traverse	PROFIBUS
1108	32-Bit Float	r/w	reference offset	PROFIBUS
1109	16-Bit Int.	r/w	jog speed 1	PROFIBUS
1110	16-Bit Int.	r/w	jog speed 2	PROFIBUS
1111	8-Bit Char	r	index for error statistics	PROFIBUS
1112	32-Bit Int.	r	read error statistics	PROFIBUS
1113	16-Bit Int.	r	Teach In	PROFIBUS
1114	16-Bit Int.	r/w	Ipeak x%	BS7200
1115-1117	—	—	reserve	
1118	32-Bit Int.	r/a	read status register	PROFIBUS
1119-1122	—	—	reserved	
1123	—	—	reserve	
1124-1127	—	—	reserved	

Abbreviations in the “Access” column

The “Access” column shows which access rights (e.g. read / write) are available via the bus, and under what circumstances (e.g. disabled).

Abbreviation	Description
a	Command can also be performed while the response monitoring is active. (only relevant for write access, read acces is always possible)
d	Command is only possible when output stage is switched off (disabled). The “disable” must be made with CW bit1 .
w	write access
r	read access
Int.	Integer

IV.4.4 Save the amplifier parameters to the EEPROM (PNU 1021)

This command can be used to transfer the amplifier parameters (see Section IV.2.3.2) from the volatile memory of the digifas™ to the non-volatile memory (EEPROM).

Telegram structure

	Request	Response
PKE/AK	3	2 / 7 in the event of a fault
PKE/PNU	1021	as transmitted
IND	15	15

IV.4.5 Mode changeover (PNU 1048)

The servo-amplifer can be switched between 5 modes. For digital speed and torque control (current control) enter the setpoint for the speed or current control via the bus (PNU 1049). Motion tasks can **not** be started in modes 1 to 4.

Telegram structure

	Request	Response
PKE/AK	3	2 / 7 in the event of a fault
PKE/PNU	1048	as transmitted
IND	15	15
PWE	mode	no significance

Mode	Description	Comment
0	position control	transmission and start of motion tasks
1(res)	analog speed control	not possible for standard PROFIBUS instrument
2(res)	analog current control	not possible for standard PROFIBUS instrument
3	digital speed control	parameter provision (n_{nom}) via PROFIBUS, position control inactive
4	digital current control	parameter provision (I_{nom}) via PROFIBUS, position control inactive

Never change the mode while the motor is rotating !

While the amplifier is enabled, a change of mode is only permitted when the speed is 0. Set the speed setpoint to 0 before changing over.

IV.4.6 Digital setpoint (PNU 1049)

If mode 3 is selected with PNU 1048, then the setpoint is provided digitally.

max. negative setpoint : 9999H max. positive setpoint : 6666H

The command is executed immediately. If the command is repeated several times with different setpoints it is **not** necessary to insert STOP commands.

Telegram structure

	Request	Response
PKE/AK	3	2 / 7 in the event of a fault
PKE/PNU	1049	as transmitted
IND	15	15
PWE	digital setpoint	no significance

When using this function, an excessive value for KE will result in instability.
The motor can accelerate out of control (run away) if the setpoint is very small.

IV.4.7 Teach In (PNU 1113)

The Teach In function is used to define new motion tasks, based on motion tasks which are already available. When the Teach In command is received, the interface program in the amplifier reads the source motion task from the motion task memory, enters the current position as the target position, designates the motion task as absolute, and saves it under the target task position.

Telegram structure

	Request		Response
PKE/AK	3		2 / 7 in the event of a fault
PKE/PNU	1113		as transmitted
IND	15		15
PWE	1 ... 120 (source motion task) 1 ... 120 (target motion task)	byte 5 byte 6	no significance

IV.5 Sample telegrams

IV.5.1 Transmit motion task

Motion-task memory no.	: **11**	000 0000 1011
Mode	: **relative**	x1x
Sign	: **positive**	0
Speed	: 12 m/min=**200.0** mm/s	0000 0111 1101 0000
Acceleration time	: 0.5 s = **50** x 10ms	0011 0010
Braking time	: 0.8 s = **80** x 10ms	0101 0000
Path setpoint	: **250.80** mm	0000 0000 0010 0101 0000 1000 0000

Telegram 1:

Byte 1	2	3	4	5	6	7	8	9	10	11	12
00110000	00001011	00000001	00000000	x1x00000	00000010	01010000	10000000	xxxx0x00	xxxxxxxx	xxxxxxxx	xxxxxxxx
PKE		IND		PWE				CW		HSW	
AK	SPM	Motion task no.	Subindex	reserved	Mod	VZ	Path setpoint	Control word		Main setpoint	

Telegram 2:

Byte 1	2	3	4	5	6	7	8	9	10	11	12
00110000	00001011	00000010	00000000	00000111	11010000	00110010	01010000	xxxx0x00	xxxxxxxx	xxxxxxxx	xxxxxxxx
PKE		IND		PWE				CW		HSW	
AK	SPM	Motion task no.	Subindex	reserved	Speed setpoint		Accel.	Brake	Control word		Main setpoint

IV.5.2 Start reference traverse

Main setpoint : **15,0** mm/s 0000 0000 1001 0110

Byte 1	2	3	4	5	6	7	8	9	10	11	12
00000000	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxx1xxx	x1xx111x	00000000	10010110
PKE		IND		PWE				CW		HSW	
AK	SPM	Motion task no.	Subindex	reserved	Mod	VZ	Path setpoint	Control word		Main setpoint	

IV.5.3 Start direct order

motion task memory no.	: **127**	000 0111 1111
mode	: **absolut**e	x0x
sign	: **positive**	**0**
main setpoint	**: 200,5** mm/s	0000 0111 1101 0101
ramp	: 0,52 s = **52** x 10ms	0011 0100
path setpoint	: **250,84** mm	0000 0000 0010 0101 0000 1000 0100

Byte 1	2	3	4	5	6	7	8	9	10	11	12
00110000	01111111	00110100	00000000	x0x00000	00000010	01010000	10000100	xxxxx1xx	x1xx111x	00000111	11010101
PKE		IND/RAMP		PWE				CW		HSW	
AK	SPM	Motion task no.	Ramp	reserved	Mod	VZ	Path setpoint	Ccontrol word		Main setpoint	

IV.5.4 Parameterizing of the amplifier, v_max

parameter no.	: **1076**	10 0011 0100
parameter value	: **350,0** mm/s	0100 0011 1010 1111 0000 0000 0000 0000
subindex	: **15**	0000 1111

Byte 1	2	3	4	5	6	7	8	9	10	11	12
00110010	00110100	00001111	00000000	01000011	10101111	00000000	00000000	xxx0x0xx	0xxxxxxx	xxxxxxxx	xxxxxxxx
PKE		IND		PWE				CW		HSW	
AK	SPM	Parameter no.	Subindex	reserved	Parameter value			Control word		Main setpoint	

V Parameter description

The digital servo-amplifiers of the digifas™ 7100 / 7200 series must be adapted to the requirements of your machine. The parameters for both types of controller are set with the aid of the operator software BS7200 or via the PROFIBUS.

This section only describes the portion of the BS7200 operator software which is relevant for the PROFIBUS CONNECT interface module. The use of the software and the parameters for current/speed control are described in the BS7200 operating manual.

V.1 BS7200 : SERVICE menu page, fieldbus monitor

The item “FIELDBUS MONITOR” appears on the SERVICE menu page. This covers a screen page which displays the data words in both the transmit and receive directions, as seen from the bus master. This page is helpful when searching for faults and in commissioning the bus communication.

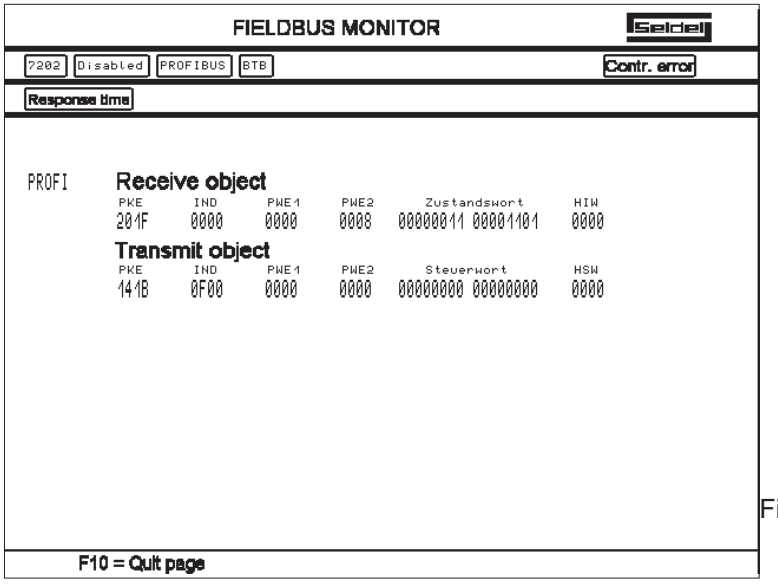


Fig. : a snapshot of the fieldbus monitor

Line 1 : Header line

Line 2 : Status line, as described in the BS7200 operating manual

PROFIBUS lines : **Receive object** : the last bus-object received by the master

here PKE : response to request “read error register”

PWE : response monitor is active

Transmit object : the last bus-object transmitted by the master

here PKE : request for the parameter “I_{rms}”

IND : load/read an amplifier parameter

V.2 BS7200 : CONNECT menu page

The menu item “Connect” appears in the menu line of the operator software. You can find all the parameters which are relevant for setting up the position controller in this menu item, apart from KE and L.

The parameters KE (PNU 1060) and L (PNU 1063) can only be set via the PROFIBUS.

The following **actual values** are displayed online on the CONNECT menu page:

s_act	current load position	(0 to 99 999 999.999 mm)
s_fehl	current load contouring error	(0 to 99.999 mm)
v_act	current load speed	(0 to 9 999.999 mm/s)

V.3 Description of CONNECT parameters

The parameters can be printed out and saved.

V.3.1 Kp, P-gain (PNU 1072)

Determines the proportional gain of the position controller.
Setting range: 0 to 8

Effects : value too low — too much lag, drive too soft
value too high — drive oscillates

V.3.2 Ff, feed-forward factor (PNU 1073)

Determines the speed feed-forward of the position controller. The feed-forward serves to ease the task of the proportional-controller. The better the Ff factor is adjusted, the better the utilization of the dynamic range of the P-controller. The most favourable setting (usually around 1.0) depends on factors which are external to the drive, such as friction, dynamic resistance and stiffness.
Setting range: 0 ... 2

Effects : value too low — the dynamic range of the P-controller is limited. Drive is lagging
value too high — the dynamic range of the P-controller is limited. Drive is oversteering

V.3.3 Control from (PNU 1102)

Determines which instrument is to be used for parameterization.

Parameter value	
0	BUS
1	PC

Section BS7200 : BUS, PC

V.3.4 Mode (PNU 1048)

The servo-amplifier can be changed over between 5 different modes (see also Section IV.2.5). For digital speed and torque control (current control) enter the setpoint for the speed or current control via the bus (PNU 1049, see Section IV.2.6). Modes 1 and 2 are not possible with the standard PROFIBUS instrument.

Parameter value	
0	position control
1	n-analog (reserved)
2	I-analog (reserved)
3	n-digital
4	I-digital

Never change the mode while the motor is rotating !

While the amplifier is enabled, a change of mode is only permitted when the speed is 0.
Set the speed setpoint to 0 before changing over.

BS7200 selection : position, (n-analog, I-analog,) n-digital, I-digital

V.3.5 Axis type (PNU 1101)

The selection of the axis type decides whether the axis is driven as a linear or a rotary axis. According to whether you choose a linear or a rotary axis, differences result in the treatment of the software limit-switches.

Parameter value	
0	rotary
1	linear

Selection BS7200: rotary/linear

Linear

A linear axis is an axis with a **limited** range of travel. The implementation of the gearing can be freely chosen, e.g recirculating ball spindle, toothed belt, or a gearbox with a crank arm. The linear axis can travel both absolutely and relatively over the range defined between the software limit-switches. A linear axis can be a:

feed drive, lifting table, positioning drive

Rotary

A rotary axis is an axis with an **unlimited** range of travel. Software limit-switches are meaningless in this context. A rotary axis always performs a relative movement, even when the tasks are input as absolute. The actual position is set at 0 for each new start. A rotary axis can be an:

axle drive, rotary table, winder, conveyor belt (endless belt), roller mill drive

V.3.6 Count direction (PNU 1097)

Determines the count direction for positional values.

Parameter value	
0	negative
1	positive

Selection BS7200 : positive / negative

Effects:

positive — with positive rotation (clockwise, looking at the end of the shaft) increasing positional values

negative — with positive rotation (clockwise, looking at the end of the shaft) decreasing positional values

In both cases, the actual position counts upwards!

V.3.7 Resolution (PNU 1079 / 1082)

The resolution establishes a relationship between the built-in measuring system and the position of the load. **The resolution determines how far the load will travel for one revolution of the motor shaft.** The calculated resolution takes into account all the conversion and gearing which lies between the motor and the load.
Non-linearities in the mechanism (backlash, elasticity etc.) are not accounted for.
The **absolute** positioning accuracy, considering inaccuracies and temperature effects in the measurement circuitry of the resolver, is ±25 angular minutes.
Setting range: 0.01 ... 999.9 mm/rev.

e.g. no of revolutions of the motor : i = 10 revs. , distance travelled for i motor revs. : s = 50mm

$$\textbf{Resolution = s / i} = \frac{\mathbf{50}\,\boldsymbol{mm}}{\mathbf{10}\,\boldsymbol{revs}} = \mathbf{5}\,\boldsymbol{mm}\,/\,\boldsymbol{rev}$$

The theoretically attainable positioning accuracy **ds** can now be calculated as follows:

$$ds = \frac{Resolution}{4096\,incr./rev} = \frac{5\,mm\,/\,rev}{4096\,incr./rev} = 0.0012207\,mm\,/\,incr.$$

Effects : Value too low — the desired physical values will not be reached
Value too high — the desired physical values will be exceeded

If you alter the resolution, it is vital that you check all the parameter in the CONNECT menu page (operator software BS7200) and in the MOTION BLOCKS, as to whether they are within the permitted min/max limits. If not, adjust the parameters !
After a change, the new parameter set is only contained in the working memory of the amplifiers. In order to store it permanently it is necessary to execute the function "Save to EEPROM" in the "Management" menu page.

After changing the resolution, certain parameters and up to 120 motion tasks will be converted by the digifas™. The conversion procedure runs in the background, and may require several seconds. Bit 31 of the status word is set while the conversion is running. Please note the following points for the period of the conversion:

- During the period of the conversion there will be a delay of several seconds in responding to request telegrams. Provide a longer timeout.
- The servo-amplifer must be disabled during the conversion period.
  If you attempt to enable the amplifer (CW bit 1), then SW bit 15 will be set (bit command : cannot be executed because of operating status). The SW bit 15 can only be cancelled by setting CW bit 7 (acknowledge error).
- Saving to the EEPROM (PNU 1021) and writing (saving) of motion blocks to the motion-block memory is not possible. Any attempt to save during the conversion will be answered by an error (AK = 7, error number 29).
- Since the newly calculated parameters are only present in volatile memory, you should save the parameters to the EEPROM as soon as the conversion is finished.

If the supply voltage fails during the conversion, or the servo-amplifier is switched off without saving the converted parameters, then, all the motion tasks and parameters will be re-calculated with the old resolution (the resolution which is in the EEPROM) as soon as it is switched on again.

Program behaviour of the operator software after entry of the resolution parameter value

A distinction is made between internal parameters and menu values. Internal parameters are the values which the program uses internally to operate the position controller.
Menu values are the actual (entered) parameters in the menu pages.

Case 1: You re-enter **the same** value for the resolution which was already in the field.
The programm recalculates the **internal** parameters. The menu values remain unchanged

Case 2: You enter a **different** value for the resolution.
In this case the correspondence between menu values and internal parameters must be re-established. There are two basic possibilities for this:

a.- The menu values are adjusted, the internal parameters remain unchanged. This makes sense when, for instance, the reason for the change of resolution is that the load is mechanically correctly placed, but the actual position is wrongly displayed. (Answer query with “Y”)

b.- The menu values remain unchanged, the internal parameters are adjusted. This makes sense when the reason for the change of resolution is that the load is wrongly placed, but the setpoint entry cannot be changed. (Answer the query with “N”)

V.3.8 v_max, maximum speed (PNU 1076)

This parameter is used to adjust the maximum traversing speed to the limits of the driven machine. The upper setting limit is calculated according to the final limit speed which is selected for the drive (BS7200 operating manual, speed controller, max. 6000 rpm).
The value which is entered serves as a limit value for the entry “v_nom” in den motion tasks. During commissioning you can limit the speed with v_max (without changing the settings for the motion blocks).
A smaller value of v_max will override the v_nom of the motion tasks.

Effects : Value too low — maximum speed cannot be implemented
Value too high — the mechanism of the driven machine may be damaged

V.3.9 Ramp type (PNU 1106)

Determines which type of acceleration or braking ramp should be used in carrying out a motion task.

Parameter value	
0	trapeze
1	sine²

Selection BS7200 : trapeze / sine²

trapeze

The drive is linearly accelerated up to the target speed with constant acceleration (acceleration time from the motion task).

sine²

The drive is accelerated up to the target speed within the acceleration time (motion task), using an acceleration ramp without steps. The resulting speed curve has a sine²-characteristic.

V.3.10 t_acc_min, maximum accelaration (PNU 1075)

A drive is always so dimensioned that it can deliver more power than required by the application. With this parameter you can fix a value for the maximum mechanical acceleration which cannot be exceeded by the drive. The setting is effective as a limit value for the acceleration and deceleration time.
Setting range: 10 to 2550 ms

Effects:	value too low	—	machinery is heavily stressed and could be damaged
	value too high	—	the required acceleration is not achieved

V.3.11 t_emergency, maximum braking deceleration (PNU 1074)

Determines the limit for the deceleration on braking. In an emergency situation the drive will be braked to a stop within the emergency braking time, provided that adequate electrical energy is still available. In this case, the braking time can be smaller than the minimum braking/ acceleration time t_accel_min. Setting range: 10 to 2550 ms.

Effects :	value too low	—	the mechanism of the machinery and/or the drive could be damaged
	value too high	—	the drive does not brake quickly enough

V.3.12 Type of reference traverse (PNU 1107)

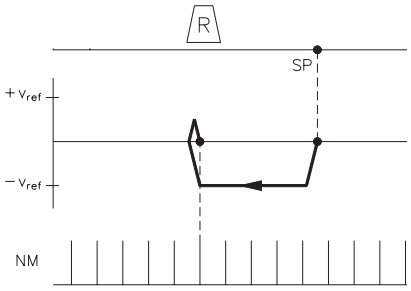
You can select which type of reference traverse should be performed.

Parameter value		Parameter value	
0	ref. traverse 1 negative	1	ref. traverse 1 positive
2	ref. traverse 2 negative	3	ref. traverse 2 positive
4	ref. traverse 3 negative	5	ref. traverse 3 positive
6	ref. traverse 4 negative	7	ref. traverse 4 positive
8	ref. traverse 5 negative	9	ref. traverse 5 positive

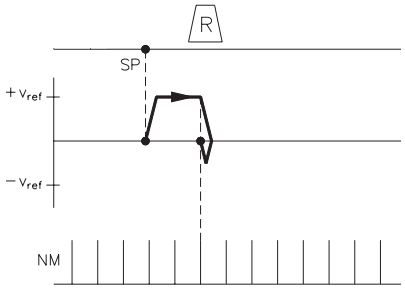
Selection BS7200 : 1-, 1+, 2-, 2+, 3-, 3+, 4-, 4+, 5-, 5+

Reference traverse 1 (1-, 1+) Go to reference switch with zero-mark recognition
In this case a reference traverse is also possible without hardware limit-switches.
The precondition is one of the initial situations shown below :

1 negative (positive count direction)



1 negative (negative count direction)



Reference traverse 1 negative is thus compatible to the reference traverse of the older software versions (prior to 6A40) if the reference offset is set to 0.
The reference point is always set to the first zero-crossing of the resolver (zero mark) after recognition of the reference switching edge/transition.
A 2-pole resolver has exactly one zero-crossing per turn, so the positioning of the zero mark within one revolution of the motor is unambiguous. If the switching edge of the reference switch is close to the zero-crossing point of the resolver, then the zero-mark positioning may vary by one motor turn.

Reference traverse 2 (2-, 2+) Go to hardware limit-switch with zero-mark recognition
The reference point is set to the first zero-crossing of the resolver (zero mark) outside the limit-switch.
The hardware limit-switch must remain activated until standstill is reached.

Reference traverse 3 (3-, 3+) Go to reference switch without zero-mark recognition
The reference point is set to the switching edge of the reference switch.

Reference traverse 4 (4-, 4+) Go to hardware limit-switch without zero-mark recognition
The reference point is set to the switching edge of the hardware limit-switch.
The hardware limit-switch must remain activated until standstill is reached.

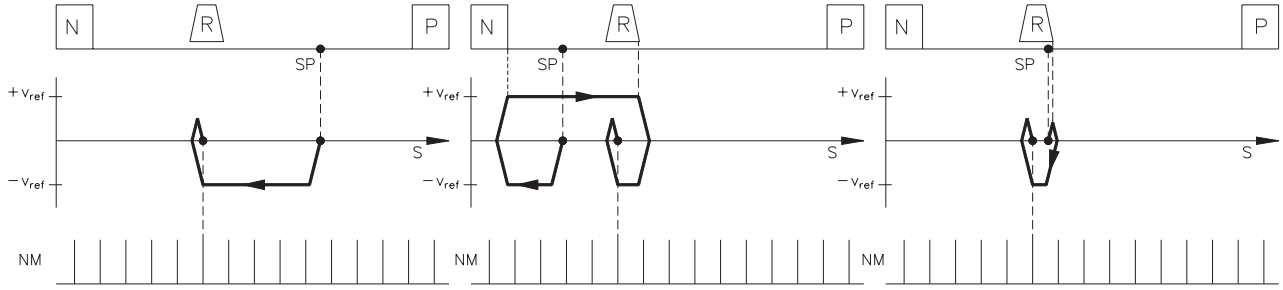
Reference traverse 5 (5-, 5+) Go to next resolver zero-mark
The reference point is set to the next zero mark of the resolver.

On the following pages you can find the paths traversed during the various reference traverses for every possible initial situation (positive count direction).

Legend					
N	Limit switch NSTOP	P	Limit switch PSTOP	SP	Initial position
R	Reference switch	v_{ref}	Set speed value	NM	Zero mark of the resolver

Sequence for reference traverse 1 negative

(with reference switch, traverse direction negative, 3 initial situations, positive count direction)



Initial point positive from reference switch | Initial point negative from reference switch | Initial point on the reference switch

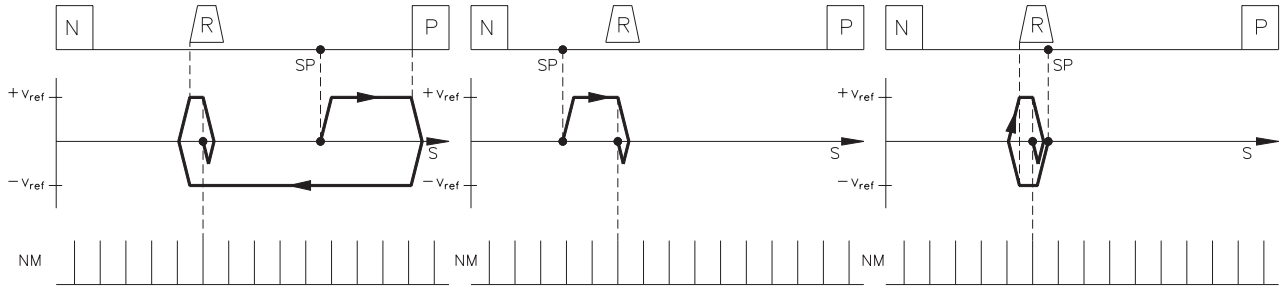
Attention !

Before starting the reference traverse, check the safety of the system to ensure that the load can also be moved if the limit switches are disconnected or defective.

The STOP limit switch function must be activated to achieve the full reference traversing functionality.

Sequence for reference traverse 1 positive

(with reference switch, traverse direction positive, 3 initial situations, positive count direction)



Initial point positive from reference switch | Initial point negative from reference switch | Initial point on the reference switch

Attention !

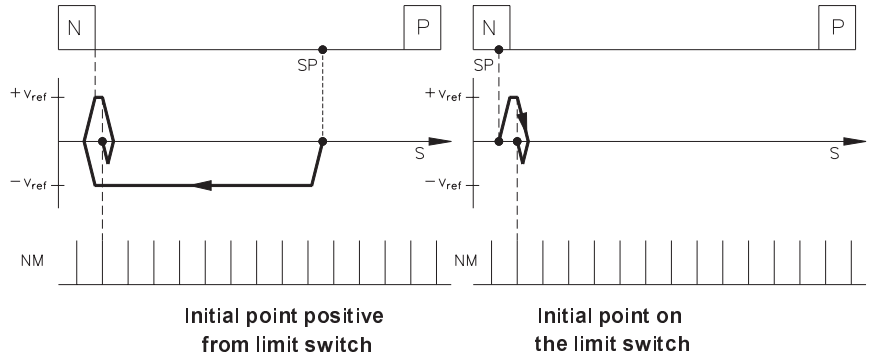
Before starting the reference traverse, check the safety of the system to ensure that the load can also be moved if the limit switches are disconnected or defective.

The STOP limit switch function must be activated to achieve the full reference traversing functionality.

Legend					
N	Limit switch NSTOP	P	Limit switch PSTOP	SP	Initial position
R	Reference switch	v_{ref}	Set speed value	NM	Zero mark of the resolver

Sequence for reference traverse 2 negative

(without reference switch, traverse direction negative, 2 initial situations, positive count direction)



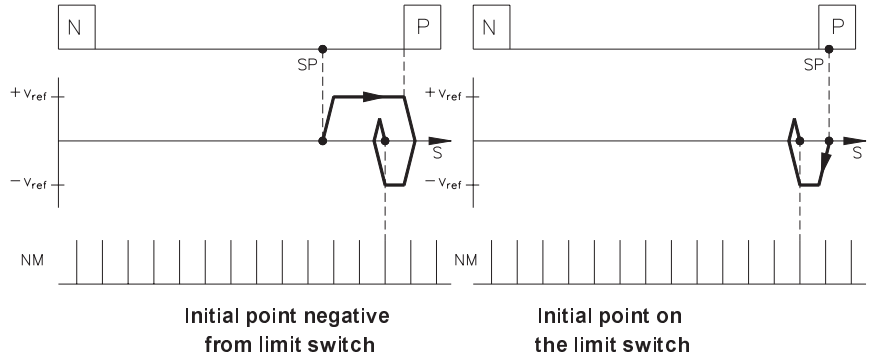
Attention !

Hardware limit switches must be present and connected.

The STOP limit switch function must be activated.

Sequence for reference traverse 2 positive

(without reference switch, traverse direction positive, 2 initial situations, positive count direction)



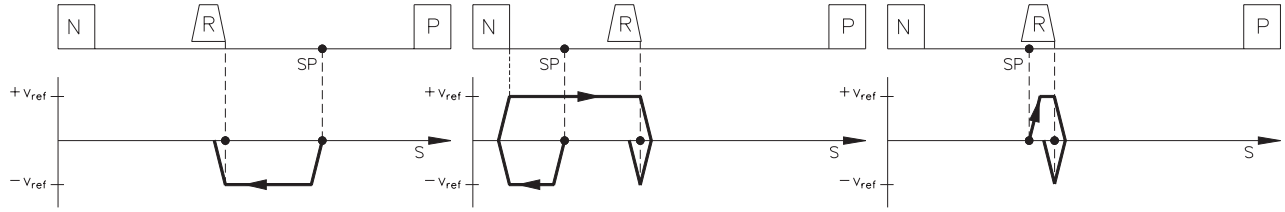
Attention !

Hardware limit switches must be present and connected.

The STOP limit switch function must be activated.

Legend					
N	Limit switch NSTOP	P	Limit switch PSTOP	SP	Initial position
R	Reference switch	v_{ref}	Set speed value		

Reference traverse 3- (with reference switch, negative traverse direction, 3 initial situations, positive count direction, without zero mark)



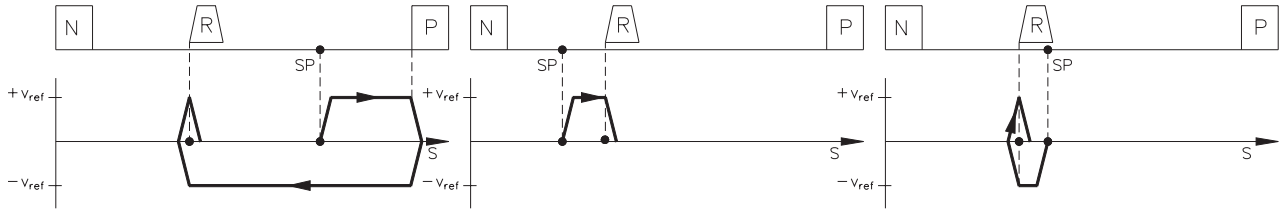
Initial point positive from reference switch | Initial point negative from reference switch | Initial point on the reference switch

Attention !

Before starting the reference traverse, check the safety of the system to ensure that the load can also be moved if the limit switches are disconnected or defective.

The STOP limit switch function must be activated to achieve the full reference traversing functionality.

Reference traverse 3+ (with reference switch, positive traverse direction, 3 initial situations, positive count direction, without zero mark)



Initial point positive from reference switch | Initial point negative from reference switch | Initial point on the reference switch

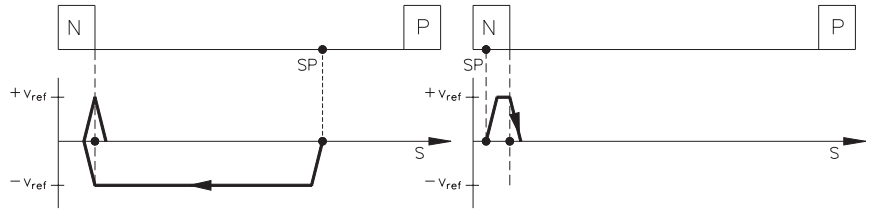
Attention !

Before starting the reference traverse, check the safety of the system to ensure that the load can also be moved if the limit switches are disconnected or defective.

The STOP limit switch function must be activated to achieve the full reference traversing functionality.

Legend					
N	Limit switch NSTOP	P	Limit switch PSTOP	SP	Initial position
R	Reference switch	v_{ref}	Set speed value		

Reference traverse 4- (without reference switch, negative traverse direction, 2 initial situations, positive count direction, without zero mark)



Initial point positive from limit switch

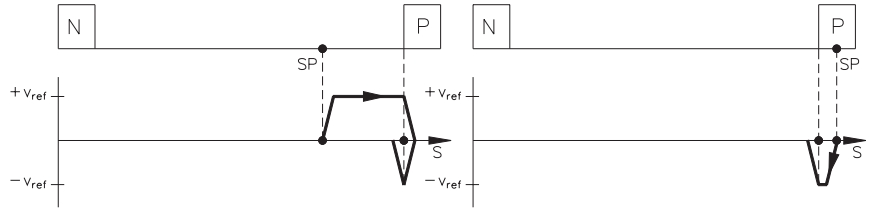
Initial point on the limit switch

Attention !

Hardware limit switches must be present and connected.

The STOP limit switch function must be activated.

Reference traverse 4+ (without reference switch, positive traverse direction, 2 initial situations, positive count direction, without zero mark)



Initial point negative from limit switch

Initial point on the limit switch

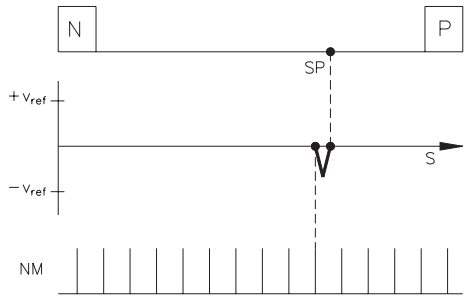
Attention !

Hardware limit switches must be present and connected.

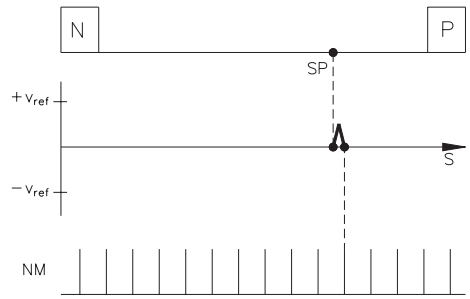
The STOP limit switch function must be activated.

Legend					
N	Limit switch NSTOP	P	Limit switch PSTOP	SP	Initial position
R	Reference switch	vref	Set speed value		

Reference traverse 5- (without reference switch, positive traverse direction, positive count direction, with zero mark)



Reference traverse 5+ (without reference switch, positive traverse direction, positive count direction, with zero mark)



V.3.13 Zero point offset (PNU 1104)

This entry is used to shift the mechanical zero point of the axis within one turn. The smallest amount by which the zero point can be shifted depends on the resolution which is set.
This parameter is only relevant for reference traverse types 1-/1+/2-/2+
Setting range: 0 ... resolution

V.3.14 Reference offset (PNU 1108)

Reference offset is used to define the reference point to be a position other then the zero position. Physically, nothing is changed by the reference offset, but the offset value is used as a reference value within the positioning control. A traverse to the reference position will thus not stop at the zero position, but at the position determined by the reference offset value.
The reference offset must be set before starting the reference traverse.
A change of the reference offset value will only take effect for the next reference traverse.
Setting range: -20% resolution ... +maximum path entry
In this case : resolution = numerical value of the preset resolution in mm
maximum path entry = 32767 * resolution < 999.999,99 mm.
If the resolution is larger than 30,52 mm/turn, then the max. possible entry is limited to 999.999,99 mm.

After changing the reference offset, and a fresh reference traverse, check that the software limit-switches and target positions in the motion blocks lie within the permitted and safe ranges.
The positional values will not automatically be adjusted if the reference point is physically moved or provided with an offset.

V.3.15 Limit-switch 2 (PNU 1100)

The software limit switch 2 is part of the monitoring functions of the position controller. It is only active in linear mode. It monitors whether the actual position is larger than the preset value, and if this is so it blocks the direction of rotation (positive count direction).
Setting range: 0 ... +maximum path entry
(maximum path entry = 32767 * resolution < 999.999,99 mm)

Effects : Value too low — if smaller than limit-switch 1, movement is impossible
Value too high — mechanical stop is reached

V.3.16 Limit-switch 1 (PNU 1099)

The software limit switch 1 is part of the monitoring functions of the position controller. It is only active in linear mode. It monitors whether the actual position is larger than the preset value.
If this is so the drive is braked in time t_emerg and remains at standstill under tension.
The direction of rotation (negative count direction) is now blocked. You can only leave the limit-switch 1 in the positive count direction.
Setting range: -20 % resolution ... maximum path entry

Effects: Value too low — mechanical stop is reached
Value too high — if larger than limit-switch 2 , movement is impossible

V.3.17 InPosition (PNU 1078)

Sets up the InPosition window. Determines up to which distance from the setpoint position the “InPosition” signal should be output.
Setting range: 0 ... 10 % of resolution

Effects :	Value too low	—	positioning time increases, no InPosition signal
	Value too high	—	axis approaches the target jerkily

Residual path processing

Rotary and linear axes possess the characteristic of “residual path processing” for relative tasks. After a movement, the drive stops at the target position, but the motor axis can stop 1/4096 of a turn away from the target position, as a result of the control technology. This deviation is taken into account at the start of the next relative task, so that the deviations are not added together to make a larger error.
The residual path processing refers only to deviations InPositioning. Approximation errors (max. 0.5/4096 turn) in the calculation of the target positions cannot be compensated for. This means that traversing in incremental coordinates with relative motion tasks can lead to in the accumulation of small positional deviations.
Depending on the accuracy which is required, you should therefore either completely avoid using incremental coordinates, or use an absolute motion task to move back to the start position.

V.3.18 Contouring error (PNU 1077)

The contouring error is the maximum difference between the setpoint position and the actual position which is allowed to occur during traversing. The contouring error entry is interpreted as a +/- window. If the system leaves this window the position controller generates a fault signal and brakes the drive with the emergency deceleration. Setting range: 0 ... 49 % of resolution.

Effects :	Value too low	—	the acceleration process is broken off
	Value too high	—	contouring error is not recognised

V.3.19 Motion task

This entry is used to select the motion task page. Task numbers can be entered from 1 to 120 and from 129 to 144. The motion task which is selected will then apear in the entry line when the motion task page is built up.
Setting range : 1 ... 120, 129 ... 144

V.3.20 Response monitoring (PNU 1103)

The response monitoring is one of the safety functions of digifas™. If the digifas™ is not access- ed from the bus within the preset response-monitoring time, then the response monitoring ensures that, if the bus communication is defective, any previously started function is terminated and the drive is brought to a standstill.

The time which must be set depends on the number of bus participants and the access frequency of the bus master.

Effects:	Value too low	—	the digifas™ will not be able to do anything
	value too high	—	the safety function will be reduced

V.3.21 Station address

The setting can only be made with the operator software BS7200.
This entry determines the station address for the attached PROFIBUS interface.
BS 7200 setting range : 3 ... 124

If the station address is changed, it must first of all be saved, and then the 25 V auxiliary supply voltage must be switched off and on again. Only then is the new address valid.

Effect : wrong address — the bus communication is disturbed

V.3.22 Jog speed 1 (PNU 1109)

The entry determines the limit speed for jog mode 1.
Setting range : 0 ... v_max

Effects : Value too low — the speed coming from the bus will be limited
Value too high — there is no limiting

V.3.23 Jog speed 2 (PNU 1110)

The entry determines the limit speed for jog mode 2.
Setting range : 0 ... v_max

Effects : Value too low — the speed coming from the bus will be limited
Value too high — there is no limiting

V.4 Description of the motion block parameter BS7200

The “Motion task” page displays the motion tasks which are stored in the servo-amplifier. The weighting which has been set can be seen from the number of places after the decimal point.

NOTE:

Because of the internal representation of the controller parameters, certain combinations of values for speed, and braking / acceleration time cannot be displayed.

In such cases the braking / acceleration time will automatically be adjusted by the controller. This adjustment is indicated by an asterisk next to the motion task.

If a motion task is altered, you must transmit it to the volatile memory of the amplifier by operating the function key F6.
Afterwards, make a permanent save by using the function “Save to EEPROM” (PNU 1021) on the “Management” menu page.

There is an entry line for each motion task, in which the following parameters can be edited :

V.4.1 No.

Here you can select which motion task is to be created or altered. The motion task numbers are binary-coded decimal numbers, with bit 0 as LSB and bit 3 as MSB.
Entry range : 1 ... 120
Motion task “0" is only used for going to the reference (type = ABS, s_nom = 0)

V.4.2 Type

This selection determines whether the motion task is to be interpreted as a relative or absolute task.
Selection : ABS / REL / REL1 / REL2
ABS : a movement to an absolute target position, referred to the reference point.
REL : if the load is within the InPosition window - relative to last target position
if the load is not within the InPosition window - relative to actual position at start
REL1 : relative to last target (in connection with motion-block changeover : e.g. summing mode)
REL2 : relative to the actual position at the start
(in connection with motion-block changeover : e.g print registration control)

In the operator software, the transmission of an absolute task to the active memory of the amplifier is blocked for the ROTARY type of axis.

If you are using a sequence of relative motion tasks, please note that each turn has an internal resolution of 16 bit (0 ... 65535). If the path cannot be precisely described with this resolution, then approximation errors may occur.

V.4.3 s_nom

This parameter detemines the path to be traversed, in mm.
Setting range : Absolute 0 ... maximum path
Relative –max ... +max

V.4.4 v_nom

This parameter determines the traversing speed in mm/s.
Setting range : 0 ... v_max
If v_max is reduced to a value which is less than v_nom at a later time, the position controller will use the smaller value.

V.4.5 t_acc

This parameter determines the time for acceleration to v_set.

Setting range : 5 or $t\_acc\_min \cdot \frac{v\_nom}{v\_max}$ ... 2550 ms

V.4.6 t_brake

This parameter determines the braking time from v_nom to zero.

Setting range : 5 or $t\_acc\_min \cdot \frac{v\_nom}{v\_max}$ ... 2550 ms

The longest possible time can be calculated in advance, with the minumum deceleration of 1 m/s², according to the following formula:
e.g v_nom = 500 mm/s a = 1 m/s² —> t_brake = t_accel = **v_nom / a** = 0.5 s = 500 ms

V.4.7 Next no.

The number of the following task, which should be started automatically when the current task is finished.

The InPosition signal will only be made available when the last motion task has been processed (i.e. there is no following task). Since the program stepping is dependent on the internal setpoint generator, increasing the size of the InPosition window does not produce any acceleration.

Entry range : 1 ... 120 (for BS7200, an empty field means that there is no following task)

V.4.8 Pause

Determines whether there is a pause between the current task and the next task.

With pause (PWE byte 6, bit 2 = 0)
The drive is braked down the predefined ramp to a standstill at the target position, before the following task is started.

Without pause (PWE byte 6, bit 2 = 1)
The drive is not braked to the target position. The speed is adjusted to the speed required for the following task, using the appropriate acceleration or deceleration ramp.

Selection : with / without

If the ramp type is set to sine² , then the drive is always braked to the target position. The setting for the pause (PWE byte 6, bit 2) is then irrelevant.

V.5 Additional motor parameters

V.5.1 K_E, voltage constant of the motor (PNU 1060)

This parameter makes an optimum adjustment of the speed-dependent feed-forward voltage of the current controller to the e.m.f. of the motor which is used. The maximum value which you can set is the voltage constant K_E [V/1000 r.p.m.] of the motor as given in the motor manual. **This parameter is adjusted by the motor-dependent default values which we supply with the BS7200 operator software, and should not be altered.**

Effect : Value too low : high motor speed cause oversteering of the current controller. In some cases, the final limit speed may not be reached.

Value too high : very low gain in the current controller or a low setting for the pulse current I_{peak} leads to positive feedback. The motor can run away.

This parameter cannot be set by the operator software.

V.5.2 L, motor inductance (PNU 1063)

This parameter makes an optimum adjustment of the speed-dependent feed-forward voltage of the current controller to the inductance of the motor which is used. The value which is set for L corresponds to the phase-phase- value of the motor inductance L [mh] as in the motor manual. **This parameter is adjusted by the motor-dependent default values which we supply with the BS7200 operator software, and should not be altered.**

Effect : Value too low : high motor speed cause oversteering of the current controller. The available voltage is not fully utilized.

Value too high : the motor current is increased unnecessarily, and the thermal utilization of the motor is bad.

This parameter cannot be set by the operator software.

V.6 Error log

The type of error and the time when it occurred are stored in the amplifier. The following parameters can be used to read out the error log from the amplifier memory.
A description of the errors can be found in the description of the BS7200 operator software.

V.6.1 Error index (PNU 1111)

Parameter (8-bit char, write / read), integral value from 0 to 17.
The components of the error statistics can be read out in association with the error index.

V.6.2 Error statistics (PNU1112)

Error statistics depend on the error index. The component of the error statistics which is to be read must be identified by the error index.

Depending on the error index, the components of the error statistics show different data formats.

The following table designates the individual components of the error statistics according to the error index. The column for error recognition provides the error index for the error which has been stored.

From error index 6 on, the type of error is assigned to the corresponding error recognition.

Error index	Data format	Error recognition	Interpretation
0	16-Bit Integer	1...12	ID of the last error
1	32-Bit Float	—	time of the last error
2	16-Bit Integer	1...12	ID of the 2nd-last error
3	32-Bit Float	—	time of the 2nd-last error
4	16-Bit Integer	1...12	ID of the 3rd-last error
5	32-Bit Float	—	time of the 3rd-last error
no. of occurrences of the error (error frequency)			
6	16-Bit Integer	1	undervoltage
7	16-Bit Integer	2	overvoltage
8	16-Bit Integer	3	output stage mains supply fault
9	16-Bit Integer	4	brake fault
10	16-Bit Integer	5	output stage fault
11	16-Bit Integer	6	aux. supply fault
12	16-Bit Integer	7	resolver error
13	16-Bit Integer	8	short to ground
14	16-Bit Integer	9	heat sink temperature
15	16-Bit Integer	10	ambient temperature
16	16-Bit Integer	11	motor temperature
17	16-Bit Integer	12	motor cable

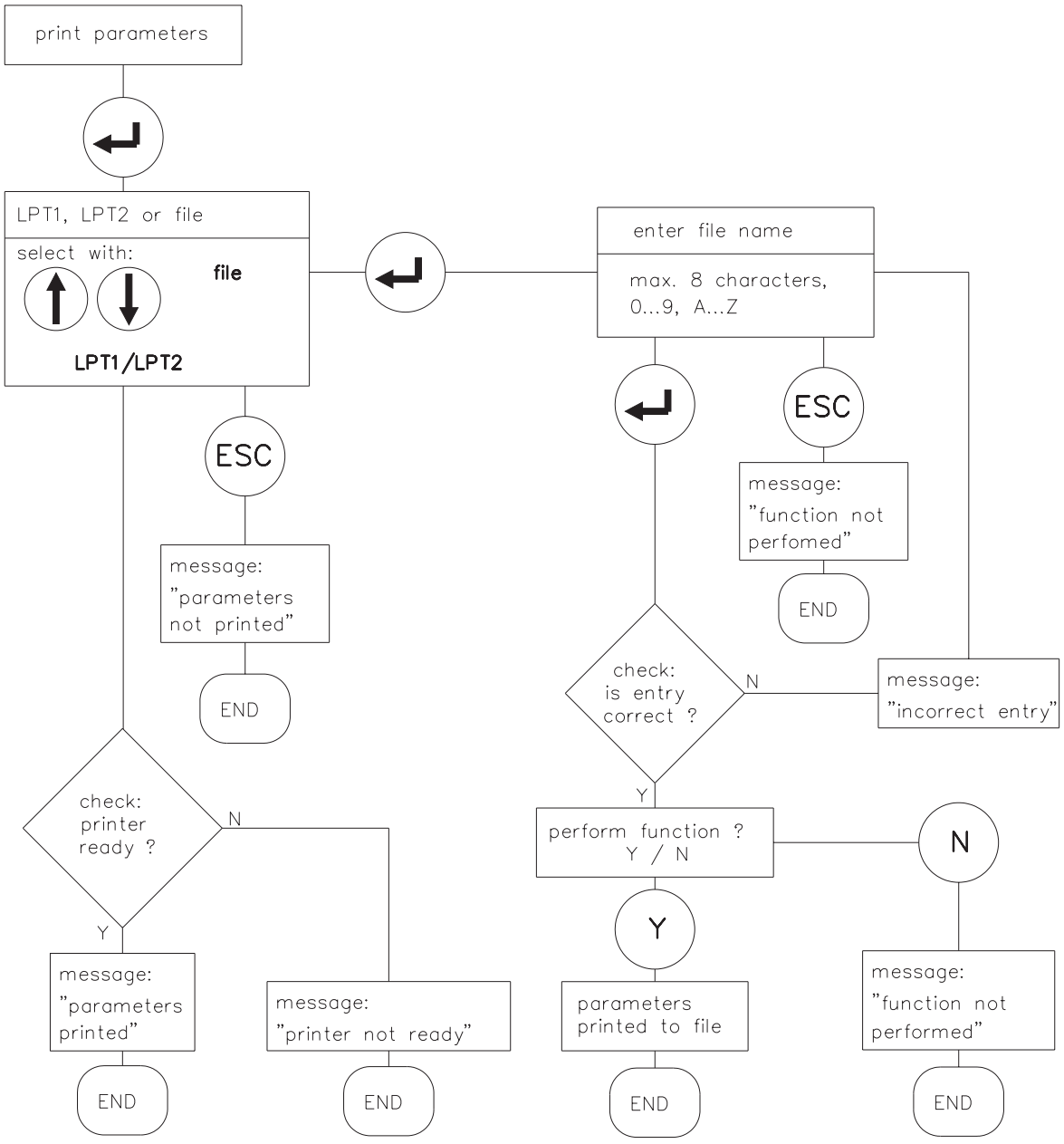
V.7 Print parameters

V.7.1 Print amplifier parameters

See BS7200 operating manual, Section III.1.7

V.7.2 Print motion-block parameters

You can use the function “Print parameters” on the MANAGEMENT menu page of the BS7200 operator software to print out the motion blocks which are stored in the amplifier.
Condition : parameter “Offline edit” = OFF
You can select between printout to the LPT1or LPT2 port, or to a file. The functional sequence is shown below.
Entry : Return



VI Appendix

VI.1 Function-block package DIGI_DP1

The function-block package operates together with the function blocks FB21, FB22 from the block package DVA_S5 Version 2.0 for L2-DP communication with drives to the PROFIBUS profile.

Variables and parameters		Kind	Type	Format
FSNR	motion block number	E	W	KH
POSH	position setpoint, higher value portion	E/A	W	KH(BCD)
POSL	position setpoint, lower value portion	E/A	W	KH(BCD)
ISTH	actual value, higher value portion	A	W	KH,KF,KM
ISTL	actual value, lower value portion	A	W	KH,KF,KM
VABS	speed setpoint	E/A	W	KF
RAMP	accel. and deceleration multiplier	E/A	W	KY
	FB195 only	E	BY	KY
SW	status word	A	W	KM
TIP1	jog command 1	E	BI	
TIP2	jog command 2	E	BI	
FREF	start reference traverse	E	BI	
STRT	start task	E	BI	
SEND	start data send	E	BI	
LESE	start data read	E	BI	
FERT	data transmission complete	A	BI	
KFEH	communication faulty	A	BI	
OFEH	task completed without error	A	BI	
MFEH	task completed with error	A	BI	
STAT	status word FB190- FB199	intern.	W	KM
IKEN	actual value ID	E	BY	KH
WAHR	actual values are valid	A	BI	

Note:

You must call FB190 as the first block. From the blocks FB191 to FB199 it is only necessary to call those FBs where the function is actually required.
The call for all FBs is made absolutely.
Between calling FB190 and calling FBs 191 to 199 it is forbidden to call any program block which uses the lubrication marker MW200 - MW254 !
For alarm processing, save the lubrication marker area and reload it !
Separate function-block calls, beginning with FB190, are required for each drive. Only one task per drive can be started at the same time. You can start the next task when the previous one has been cancelled or completed and BUSY=0.

If data words are declared to the FBs as parameters, then these are deposited in the parameterized net-data block (FB190 NUDB). Take care that none of the areas which are used by the DVA_S5 FB package are overwritten.
For the operation of the function-blocks of DVA_S5 please refer to the appropriate Siemens manual. For the structure of the bus system and the parameterization of the IM308B please refer to the Siemens manual COM ET200.

System requirements: IM308B implementation level 5 (DP Standard) or IM308C
COM ET 200 Version 4
FB package DVA_S5 Version 2.0 (for CPU945 from Version 3.0)

VI.1.1 FB190 DIGISTAT

The FB190 provides the connection between the net-data block and the DIGIFDP1 FBs for the particular axis which is parameterized. Additional funcitons are: transmission of the CW bit 0-7, provision of the SW from digifas™ and display of the communication status.

Tech. Data:

block name:	DIGISTAT	
markers used:	MW200-MW254	
access to:	DB net-data	
processing times:	CPU	max
	941B/942B	2.18 ms
	943B	2.13 ms
	944B	0.09 ms
	928B	0.09 ms

Function of the inputs/outputs of FB190:

Name	Type/kind	Comment
NUDB	KY	left byte: number of the net-data DB right byte: start DW of net-data for the corresponding drive
STB1	KM/E	control bits 0-7 to digifas™
ZSW	KM/A	status information bit 0-15 from digifas™
KFEH	BI/A	1 = communication error: slave station out of action repeat counter overflow
BUSY	BI/A	1 = task being processed
OFEH	BI/A	1 = PKW task completed without error
MFEH	BI/A	1 = PKW task completed with error
STAT	KM	marker word for status FB190 - FB199

VI.1.2 FB 191 DIGIHAND

FB191 provides the setup functions: jog, reference traverse, and set reference point.

Tech. data:

block name:	DIGIHAND		
markers used:	MW200-MW254		
processing times:	CPU	idle	max
	941B/942B	0.83	1.44 ms
	943B	0.82	1.41 ms
	944B	0.03	0.06 ms
	928B	0.03	0.06 ms

Function of the inputs/outputs of FB191:

Name	Type/kind	Comment
TIP1	BI/E	1 = drive travels with ± VABS in jog mode (limit value: jog 1)
TIP2	BI/E	1 = as TIP1 but with limit parameter jog 2 in digifas™
FREF	BI/E	1 = drive travels with VABS to the reference point. The direction is taken from the digifas™ parameter set. 0 = 0-signal before completion aborts the reference traverse.
SREF	BI/E	1 = the actual position of the drive becomes the reference point.
VABS	KF/E	± traverse speed. The sign is only relevant for TIP1/2 + = in the direction of increasing positional values

VI.1.3 FB195 DIGIDIRE

Transmits a complete motion task to the digifas™ and causes immediate execution.
This function can only be performed with a normalised axis (reference point available).

Tech. data:

block name:	DIGIDIRE		
markers used:	MW200-MW254		
processing times:	CPU	idle	max
	941B/942B	0.67	2.8 ms
	943B	0.67	2.62 ms
	944B	0.03	0.12 ms
	928B	0.03	0.12 ms

Function of the inputs/outputs of FB195:

Name	Type/kind	Comment
POSH	KH-BI/E	higher-value portion of the pos./path setpoint, interpreted as BCD
		bit 12 = sign 0 = +, 1 = —
		bit 14 = mode: 0 = absolute positioning
		1 = relative positioning
POSL	KH/E	lower-value portion of the pos./path setpoint, interpreted as BCD, two decimal places
VABS	KF/E	traverse speed
RAMP	KY/E	acceleration / deceleration time multiplier (byte)
STRT	BI/E	start command
		1 = task is transmitted to digifas™ and started
		0 = 0-signal before completion (INPOSITION) aborts the motion task
STAT	KM	same marker word as parameterized in FB190

VI.1.4 FB196 DIGISATZ

Selects a motion task from the motion-block memory of the digifas™ and starts to carry it out.
The motion task is performed using the data which are stored in the digifas™ .
This function can only be performed with a normalised axis (reference point available).

Tech. data:

block name:	DIGISATZ		
markers used:	MW200-MW254		
processing times:	CPU	idle	max
	941B/942B	0.67	1.89 ms
	943B	0.67	1.85 ms
	944B	0.03	0.09 ms
	928B	0.03	0.10 ms

Function of the inputs/outputs of FB196:

Name	Type/kind	Comment
FSNR	KH/E	number of the motion block in the digifas™ motion-block memory
STRT	BI/E	1 = start selected motion block
		0 = 0-signal before completion (INPOSITION) aborts the motion task
STAT	KM	same marker word as parameterized in FB190

VI.1.5 FB197 DIGILADE

FB197 loads data from the PLC memory into the motion-block memory of the digifas™ .
The transmission is made in two successive telegrams and is followed by an 0-order.

Tech. data:

block name: DIGILADE
markers used: MW200-MW254
processing times:

CPU	idle	max
941B/942B	0.18	2.24 ms
943B	0.18	2.18 ms
944B	0.01	0.11 ms
928B	0.01	0.12 ms

Function of the inputs/outputs of FB197:

Name	Type/kind	Comment
FSNR	KH/E	number of the motion-block memory in digifas™
POSH	KH-BI/E	higher-value portion of the pos./path setpoint, interpreted as BCD bit 12 = sign 0 = +, 1 = — bit 14 = mode: 0 = absolute positioning 1 = relative positioning
POSL	KH/E	lower-value portion of the pos./path setpoint, interpreted as BCD, two decimal places
VABS	KF/E	traverse speed (sign has no significance)
RAMP	KY/E	left byte: acceleration multiplier right byte: deceleration multiplier
SEND	BI/E	1 = send command data
FOLG	BY/E	Number of next task 0 = no following task
BRMS	BI/E	Transition between motion blocks: 0 = with braking 1 = transition with the parameterized ramp of the next task (only for ramp type = trapeze)
STAT	KM	same marker word as parameterized in FB190
STEP	BY/E	internal status byte for task processing
FERT	BI/A	confirmation: transmission completed

VI.1.6 FB198 DIGILESE

FB198 reads data from the motion-block memory of the digifas™ and transmits it to the PLC memory. The transmission is made in two successive telegrams and is followed by an 0-order.

Tech. data:

block name: DIGILESE
markers used: MW200-MW254
processing times:

CPU	idle	max
941B/942B	0.18	1.99 ms
943B	0.18	1.75 ms
944B	0.01	0.11 ms
928B	0.01	0.11 ms

Function of the inputs/outputs FB198:

Name	Type/kind	Comment
FSNR	KH/E	number of the motion-block memory in digifas™
POSH	KH-BI/A	higher-value portion of the pos./path setpoint, interpreted as BCD bit 12 = sign 0 = +, 1 = – bit 14 = mode: 0 = absolute positioning 1 = relative positioning
POSL	KH/A	lower-value portion of the pos./path setpoint, interpreted as BCD, two decimal places
VABS	KF/A	traverse speed (sign has no significance)
RAMP	KY/A	left byte: acceleration multiplier right byte: deceleration multiplier
MFOL	BI/E	read next task number 0 = without 1 = with
FOLG	BY/A	next task number 0 = no following task
BRMS	BI/A	Transition between motion blocks : 0 = with braking 1 = transition with the parameterized ramp of the next task (only for ramp type = trapezoidal)
LESE	BI/E	1 = read command data
STAT	KM	same marker word as parameterized in FB190
STEP	BY/E	internal status byte for task processing
FERT	BI/A	confirmation : transmission completed

VI.1.7 FB199 DIGI-IST

FB199 switches on the cyclical transmission of an actual value or the error register from the digifas™ to the PLC. The actual value is transmitted repeatedly until a new task is transferred.

Techn. data :

block name: DIGI-IST
markers used : MW200-MW254
processing times :

CPU	idle	max
941B/942B	0.81	2.83 ms
943B	0.80	2.76 ms
944B	0.03	0.13 ms
928B	0.03	0.13 ms

Function of the inputs/outputs of FB199:

Name	Type/kind	Comment		
ISTH	/A	higher-value portion of the actual value, error register 2		
		If actual -value transmission is off, output is ISTH = 0000		
ISTL	/A	lower-value portion of the actual value, error register 1		
		If actual -value transmission is off, output is ISTL = FFFF		
LESE	BI/E	1 = switch on the actual-value transmission		
IKEN	BY/E	actual-value ID		
		1 = actual position	2 words	/KH BCD
		2 = contouring error	1 word	/KF(ISTL)
		3 = not used		
		4 = angle of rotation	1 word	/KF(ISTL)
		5 - 30 not used		
		31= error register	2 words	/KM
WAHR	BI/A	1 = output values are valid		
		0 = output values are invalid		
STAT	KM	same marker word as parameterized in FB190		

VI.1.8 FB 200 DIGIPARA

FB 200 is used for reading and writing parameter values, according to the documented list of parameter numbers.

Techn. data :

block name :	DIGIPARA		
markers used :	MW200-MW254		
processing times :	CPU	idle	max.
	941B/942B		
	943B		
	944B		
	928B		

Function of the inputs/outputs of FB 200:

Name	Type/kind	Comment
PANR	W/E	parameter number according to the table in Section IV.2.3.2
PTYP	BY/E	parameter type 0 - The values in PAHI and PALO are accepted without any change 1 - there is a conversion between the PLC-specific KG floating-point format and the IEEE-854 floating-point format im digifas™ (only relevant for 32-bit float numbers)
REWR	BI/E	read/write flag : 0 - read a parameter 1 - write a parameter
STRT	BI/E	1 - start a read/write task 0 - cancels the task
AKTV	BI/A	active display of task, 1 = active
FERT	BI/A	confirmation : 1 = parameter processed
FEHL	BI/A	error display : 1 = parameter could not be read or written
PAHI	W/A (E)	higher-value portion of the parameter value
PALO	W/A (E)	lower-value portion of the parameter values
STAT	W/A (E)	same marker word as parameterized in FB 190

VI.2 Function-block library (FB)

The function blocks are described in the function-block library (FB) for the PROFIBUS DP-master interface SINEC CP5412 (A2)

The function blocks are available as source-code (in the high-level language **C**), and so, if required, they can be optimized and adapted, integrated directly into an application, or used as examples for the connection to the SINEC DP-programming interface.

In order to be able to use the function-block package, the CP5412 (A2) must be in the productive phase, i.e. the initialization for the productive phase must be made directly via the SINEC DP-programming interface. The same is valid for control telegrams (sync-/unsync mode, freeze-/unfreeze mode, clear mode) and other DP functions (see the manual for the SINEC DP-programming interface).

The detailed instructions for commissioning, initalization, loading databases, can be found in the corresponding SINEC manuals.

VI.2.1 Further documentation / manuals

PROFIBUS DP-master operator software for digifas™ 7100/7200 Mat.No.: 85577
SINEC Product Information S79200-A0737-X-02-7437 revision 06/95
SINEC Installation Instructions S79200-A0737-X-01-7419
SINEC Introduction to CP5412 (A2), C79000-G8900-C068 version 01
SINEC DP-master operation, project-planning with COML DP, C79000-G8900-C069 ver.01
SINEC DP-programming interface C79000-G8900-C071 version 01

VI.2.2 System requirements

— digifas™ 7100/7200 with PROFIBUS CONNECT interface
— bus connection and bus medium: DP standard (EN 50170)
— transmission speed max.: 1.5 Mbit/s
— master station with PROFIBUS bus interface (SINEC CP5412 (A2))
— software SINEC DP-5412/MS-DOS, WIN V1.00 or higher

VI.2.3 Installation of the function-block library

The function-block library is available on a diskette as a self-extracting file **“proflib.exe”.**
Proflib.exe is copied to the working directory defined by the user and then started.
This executable file unpacks all the relevant files (see also the section with the description of the files for the function-block library).

In addition to the *include* and *source* files, an instrument specification file for the digifas™ is also included (**digifas.gsd**). This file is required for the project-planning (Siemens project-planning tool COML DP) of the communication procedure. Copy it into the appropriate directory (see the manual for SINEC DP-master operation, project-planning with COML DP).

Also included is a database in the form of a sample text file (digifas.txt) for a PROFIBUS DP-bus system. This text file can be called from within COML DP. The database contains the project-planning only for digifas-slaves and a DP-master.

A master program **prof7200.exe** (see the description of the DP-master operator software for digifas™) and a demo-program **profdemo.exe** (see the section on the demo-program) are also provided.

VI.2.4 Demo-program “profdemo.exe”

The demo program is intended to demonstrate the application of the function-blocks in the library, and for this reason it is available as source-code (cpdemo01.cpp). This program carries out the initialization of the CP 5412 (A2) and three motion functions (reference traverse, jog mode, motion task) in a predetermined sequence. The production program accesses a digifas-slave with the station address **3**.

When **profdemo.exe** is run, the initialization messages of the CP appear (see also the Siemens manuals).

If the initialization has been successfully concluded, the reference traverse is started and the output of the position (negative position output) and the status word will follow.

If the initialization was unsuccessful, an error number will be output which corresponds to the return value of the DP-programming interface function “dpn_init” (see the appropriate Siemens manuals) and the demo-program is terminated. In this case, the correct installation of the Siemens communications module (CP 5412 (A2) hard- and software) is the first thing to check.

Pressing any key will stop the reference traverse. The momentary position is set as a reference point (position = 0.00 mm), jog mode is started, and the position and the status word are shown in the display.

Pressing a key again will stop the jog mode and start a defined motion task. The motion task is concluded by a “Drive InPosition” message, and you can quit the demo-program by pressing a key again.

If the axis type for the servo-amplifier is set to “linear”, then the positions of the software limit-switches must be considered. The settings for the software limit-switches and the type of axis can be altered in the operator software (BS7200) on the CONNECT page, and stored on the MANAGEMENT page.

VI.2.5 Description of the transfer structures of the function blocks

This section describes the transfer structures which are used when the function blocks are called. These structures can include both statement- and return elements. Detailed information on these elements can be found below, in the section “Description of the function blocks for the master connection CP5412 (A2)”.

The transfer structures are constructed in such a way that not only management parameters (station address, timeouts for the receive routine, status word) for the function blocks are made available, but also necessary parameters or variables for the controller.

If the application requires additional information which is not included in the transfer structures (e.g. status of the DP-slave, status of the DP-master, event messages, diagnosis information), then the transfer structures can be expanded, or this information can be called up directly via the SINEC DP-interface.

All the transfer structures are contained in the *include* file user_st.h

VI.2.5.1 “Verwaltung” structure

The *verwaltung* structure is a sub-structure which is embedded in the following structures.

VERWALTUNG structure :

```
struct VERWALTUNG  {
        unsigned char stationsnr;
        float tout;
        unsigned short int zustandsreg;
};
```

Structure elements

stationsnr

The structure element *stationsnr* includes the identification of the slave-station (digifas ™) which is to be addressed, and must be initialized before every DP-function call. The address for the
digifas™ interface module is set through the BS7200 operator software on the CONNECT page.

tout

The structure element *tout* is used for function blocks which have to wait for a response from the digifas™ (e.g. requesting an actual value). The waiting time (time-out) is parameterized in *tout*, in seconds (for example, *tout* = 0.001 sec). The finest resolution is 1 ms. *Tout* is required because there are no interrupt functions available for the reception of response telegrams. This means that it is necessary to poll the data area of the DP-master in the input buffer of the station concerned. The waiting/polling time which must be parameterized depends on the number of network participants, the processing time of the interface module, the maximum number of parameterized telegram repeats, and the minimum TSDR-time. For this reason, it is not possible to present fixed times here for the individual function blocks.

zustandsreg

The structure element *zustandsreg* includes the current status register (status word SW) of the digifas™. The SW is updated by every DP-function.

VI.2.5.2 “Datensatz” structure

The *datensatz* structure includes both statement- and return elements. These are used for loading and reading data blocks (motion blocks). The loading of data blocks can either be made as a direct motion task, or stored as a motion task in the motion-block memory of the digifas™ (max. 120 motion blocks).

Datensatz structure :

```
struct datensatz {
        struct VERWALTUNG verw;
        unsigned char ds_nummer;
        unsigned char art;
        long s_soll;
        short unsigned int v_soll;
        unsigned char a_beschl;
        unsigned char a_brems;
};
```

Structure elements

ds_nummer

This element represents the *datensatz* (= data-block) number. The data-block number gives the position of the data block which is to be loaded or read in the motion-block memory.
Position numbers must be selected in the range from 1 to 120. The data-block number is irrelevant for direct motion tasks.

art

designates the motion task as absolute (initialization of the structure element with 0) or relative (initialization of the structure element with 1).

s_soll

represents a weighted positional value (weighting factor = 2) as a 32-bit integer number.
Example : Position value = 123**.45** mm => s_soll = 123**45**
Position value = 12345 mm => s_soll = 12345**00**
The maximum entry range for the position value is 0 ... ±99999.99 mm.

v_soll

gives the speed setpoint (weighting factor = 1) as an unsigned 16-bit integer number.
Example : Speed setpoint = 123**.4** mm/s => v_soll = 123**4**
Speed setpoint = 1234 mm/s => v_soll = 123400
The speed is defined in the range of 0 ... 3276.7 mm/s. If a negative speed is to be given, then the controller performs a magnitude calculation.

a_beschl

This structure element presents the acceleration multiplier (acceleration ramp) as an unsigned 8-bit number. The resolution is 10ms per increment. The acceleration multiplier is defined in the range from 0 to 255. This corresponds to the time range 0 ... 2550ms.

a_brems

This structure element presents the deceleration multiplier (deceleration ramp) as an unsigned 8-bit number. The resolution is 10ms per increment. The deceleration multiplier is defined in the range from 0 to 255. This correponds to the time range 0 ... 2550 ms.

VI.2.5.3 “Istwerte” structure

The *istwerte* structure (Istwerte = actual values) can be used to read out the following process data or conditions (error or status register). All the positions and speed which are to be read are weighted values.
The weighting factor for speeds is ‘1’, and for positions or contouring errors it is ‘2’.

Istwerte structure:

```
struct istwerte  {
        struct VERWALTUNG verw;
        unsigned char iwert_ken;
        long s_ist;
        short int v_ist;
        unsigned short int inc_winkel;
        short int s_fehler;
        unsigned long fehlerreg;
};
```

Structure elements

iwert_ken

The structure element *iwert_ken* provides the actual-value IDs which designate the actual value (process value) which is to be read. The following actual-value IDs have been defined:

Actual-value ID	Description	Data format
1	Position Speed	32-Bit Integer (weighted) 16-Bit Integer (weighted)
2	Contouring error	16-Bit Integer (weighted)
3	not used	
4	Rot. angle	16-Bit Integer
5...30	not used	
31	Error register	32-Bit Integer

s_ist
The structure element *s_ist* contains the weighted actual position (weighting factor = 2).
Example : s_ist = 123**45** mm => actual position = 123**.45**
The maximum range which can be represented for the actual position is 0 ... ±99999.99 mm.

v_ist
The structure element *v_ist* contains the weighted actual speed (weighting factor = 1).
Example : v_ist = 123**4** mm/s => actual speed = 123**.4**
The maximum range which can be represented for the actual speed is 0 ... ±3276.7 mm.

inc_winkel
The structure element *inc_winkel* contains the present angle of rotation in increments.
One turn (360°) is resolved into 4096 increments.
Example : inc_winkel = 1000 => present angle of rotation = 88°

s_fehler
The structure element *s_fehler* contains the weighted contouring error (weighting factor = 2).
Example : s_fehler = 1**23** => contouring error = 1.23 mm.
The maximum range which can be represented for the contouring error is 0 ... ±327.67 mm.

fehlerreg
The structure element *fehlerreg* contains the current error register of the digifas™.

VI.2.5.4 Structure for starting a motion block from the motion-block memory

In order to be able to use this structure, the motion block data which is required must be available in the motion-block memory.

Structure fs_start:

```
struct fs_start  {
        struct VERWALTUNG verw;
        unsigned char fs_nr;
};
```

Structure elements

fs_nr

The structure element *fs_nr* contains the motion-block number for the motion-block which is to be started. The motion-block number can be defined in the range 1 ... 120.

VI.2.5.5 Structure for setting-up functions

This structure initialises the speed setpoints for setting-up functions (reference traverse, jog operation).

Structure v_soll:

```
struct v_soll  {
        struct VERWALTUNG verw;
        short int geschw;
};
```

Structure elements

geschw

The structure element *geschw* defines the weighted speed setpoint for the particular setting-up function (weighting factor = 1).

Example : Speed = 123**.4** mm/s => geschw = 123**4**
The maximum entry range for the speed setpoint is
0 ... ±3276.7 mm/s.

VI.2.5.6 Structure for transferring parameters, commands and actual values

This structure can be used to initialise parameters for a read or write access, commands (e.g. save to EEPROM) or actual values. As opposed to the “Istwerte” structure already described above, this structure can be used to request additional actual values.

Structure v_soll:

```
struct para {
        struct VERWALTUNG verw;
        unsigned int para_nummer;
        unsigned char datatyp ;
        unsigned char rw_acc;
                void * datum;
};
```

Structure elements

para_nummer:

The parameter number (multiplexer) is used to designate the parameter to be requested, the actual value or the command.
The possible parameter numbers can be found in Section IV.2.3.2. Every parameter number is provided with an offset of 1000. (e.g. parameter no. 21 _ 1021, save parameter to EEPROM).

datatyp:

The structure element *datatyp* is used to determine the data type which is pointed to by the *datum* pointer. The following data types are possible (see also file USER_ST.H):

Designator	Data type
DTYP_DWORD	32- Bit Integer
DTYP_WORD	16- Bit Integer
DTYP_FLOAT	32- Bit Float
DTYP_CHAR	4 x 8-Bit characters
DTYP_BYTE	8- Bit Integer

rw_acc :

The structure element *rw_acc* is used to define a read or write access. Commands are basically performed with a write access.
The following designators have been defined (see also file TE_STRUCT.H):

Designator	Description
AK_READ	read access
AK_CHANG	write access

datum :

This structure element is used to initialise the variable. The data type of the variable must match the data type which is defined for the *datatyp* structure element.

VI.2.6 Overview of the error IDs

The return values of the DP-function blocks are error IDs, which can either originate at the Sinec DP-interface or be generated by the digifas™. Error messages from the SINEC DP-interface are only passed on by the DP-function blocks, i.e. the definitions of constants of the SINEC DP-interface (DPN_ACCESS_ERROR, DPN_APPL_LIMIT_ERROR...) can still be utilised by the user.
An exception is the SINEC-DP message “No error in processing the function call” (DPN_NO_ERROR) : since the return value ‘0’ is already used as an error message by the controller, the correct execution of the particular function is indicated by a ‘-1’.
Explanations of the error messages of the SINEC DP-interface can be found in the manual for the SINEC-DP programming interface.
The possible error messages are listed below.

Return value	Description
-1	no error in processing the function calls _ transfer successful
0	impermissible parameter number *
1	parameter value cannot be altered *
2	passed lower or upper value limit *
3	error in subindex (=Index) *
4-10	not used *
11	no operator access *
12-16	not used *
17	job not executable because of operating state *
18	other error *
19	not used *
20	impermissible command, axis not normalized *
21	job not available *
22	BCC-error in motion task *
23	time-out (defined in structure element *t_out*) in receive function
24-100	not used
101	erroneous actual-value ID (only relevant in the DP-function *dp_r_istwert*)
102-127	not used
128-149	error messages from the SINEC DP-interface (see SINEC DP-interface manual)
>149	not used

VI.2.7 Description of the function blocks for the CP5412 (A2)

All the DP function-blocks are described in detail in this section. The structure which is to be used for the particular function call is divided according to statement and return parameters and the appropriate parameters are marked (**bold print**). In this way you can not only recognize which parameters must be filled before the function call, but also which parameters are updated or altered by the function.

The correct execution or transfer of the functions can be evaluated by means of the return values of the DP-function, so that a verification of the transfer is achieved.

Position information is transferred in BCD-format. Function blocks which transmit or receive position information convert the data into BCD-format themselves. The user only has to enter weighted 32-bit integer values into the particular structure element.

Requirements :

The function-block package for the CP5412 (A2) has been tested with the aid of the master program PROF_72.EXE from Seidel. The functions of the FBs have been compiled with the Microsoft-Compiler 8.0 (Microsoft ™ C/C++ Optimizing Compiler Version 8.0) and bound into the master program by using the Microsoft-Linker Version 8.00 (Microsoft ™ Segmented Executable Linker Version 5.50).

VI.2.7.1 Setting-up and general functions

This section describes functions which are not only necessary for the commissioning of the digifas™, but can also be used for control purposes while the application is running.

VI.2.7.1.1 Transmitting a control command (control word CW) to the digifas™

Function call :

short int dp_send_stw(struct VERWALTUNG adm)

Description :

This function transmits the control word only. The control word is defined with the help of the auxiliary function *dp_ctrl_word* (see Section “Auxiliary function for the control word”).
For example, the following control commands can be defined: controller enable ON/OFF, contouring error / acknowledge response monitoring, START/STOP motion task, ...
This function does not carry out a plausibility check of the control flags which are have been set by the user.
Jog operation or a reference traverse cannot be started with this function. There are separate DP-functions for these operations.

Stated parameters

adm.**stationsnr**
adm.**tout**
adm.zustandsreg

Return parameters

adm.stationsnr
adm.tout
adm.**zustandsreg**

Note:

digifas ™ responses or answers to control commands can be read from the status word or register and the error register, which must be explicitly read out. Control commands should remain present until a confirmation has been made by the relevant register.

Example :

Control command “Acknowledge response monitoring”

(Bit 7 of control word) => confirmation by bit 3 in the error register

The control word is transmitted with every DP-function. Nevertheless, it may happen that a control word needs to be implemented in the meantime (“Rapid stop”), without transmitting any user data.

The auxiliary function *dp_ctrl_word* is deliberately not included in this function, since it should be possible to set several control bits with a single function call, without a lot of effort (“Rapid stop” and “Acknowledge response monitoring”).

VI.2.7.1.2 Jog mode

The drive can be moved in jode mode with the aid of this function.

Function call :

dp_tippen(struct v_set v_tippen)

Description :

The signed speed setpoint is passed to the function as the set parameter. The function uses control bit 8 (Jog 1) for its execution.

Requirements :

see Section IV.1.3

The necessary condition can be fulfilled by the function *dp_send_stw* .
Jog operation is terminated by cancelling control bit 8 (Jog 1 OFF) with *dp_send_stw* .

Stated parameters :

v_tippen.**verw**
v_tippen.**geschw**

Return parameters :

This function does not deliver any return parameters. The digifas ™ responses can be read from the status word or register and the error register, which must be explicitly read out.
For example, the activation of the motion functions can be read in the status word SW (motion function active : bit 14 = 1).

VI.2.7.1.3 Reference traverse

The reference traverse is started with the aid of this setting-up function. Further information about the reference traverse can be found in the PROFIBUS CONNECT manual.

Function call :

short int dp_ref_fahrt(struct v_soll v_ref)

Description :

This function sets bit 11 in the control word CW to 1 and passes the reference traverse speed to the main setpoint. If bit 11 is cancelled before the reference point is reached, then the reference traverse is abandoned. The reference traverse is finished when the reference point switch has been reached. This is signalled by bit 12 = 1 in the status word SW, and bit 11 in CW can then be cancelled.

Requirements :

see Section IV.1.1.5

Stated parameters :

v_ref.**verw**
v_ref.**geschw**

Return parameters :

This function does not deliver any return parameters. The digifas ™ responses can be read from the status word or register.

VI.2.7.2 Position functions

The following position functions can only be carried out when the axis has been normalized, otherwise the task will be rejected, with the error message “Impermissible command, axis not mormalized”.

VI.2.7.2.1 Direct motion task

The direct motion task is transferred to the drive, and can be started directly by an explicit setting of CW bit 10. Direct motion tasks cannot be stored in the motion block memory of digifas ™.

Function call :

short int dp_d_fahrsatz(struct datensatz *fs)

Description :

The function *dp_d_fahrsatz* describes the complete motion task in a 12-byte telegram. The acceleration and deceleration ramps are identical for direct motion tasks, so they are defined just by the set parameter *a_beschl*.

Requirements :

See Section IV.1.2.1

Set (statement) parameters :

fs_**verw**
fs_ds_nummer
fs_**art**
fs_**s_soll**
fs_**v_soll**
fs_**a_beschl**
fs_a_brems

Return parameters :

This function does not deliver any return parameters. The responses of the digifas ™ can be read from the status word or register and the error register, which must be explicitly read out. For example, the activation of the motion function can be read from the status word SW (active motion function : bit 14 = 1).

VI.2.7.2.2 Starting a motion task from the digifas™ motion-block memory

In order to start a motion task from the motion-block memory of the controller, this task must have been previously defined and saved.

Function call :
short int dp_start_fs_eeprom(struct fs_start fs_eeprom)

Description :
The function *dp_start_fs_eeprom* starts a motion block from the motion-block memory. The relevant control flags (no rapid stop, no pause, motion task START) are set accordingly.

Requirements : See Section IV.1.2.2

Set parameters :
fs_eeprom.**verw**
fs_eeprom.**fs_nr**

Return parameter :
This function does not deliver any return parameters. The responses of the digifas ™ can be read from the status word or register and the error register, which must be explicitly read out. For example, the activation of the motion function can be read from the status word SW (active motion function : bit 14 = 1).

VI.2.7.3 Data transfer functions

The following DP-functions have nothing to do with the positioning process, but transfer data between the DP-master and the digifas ™.

VI.2.7.3.1 Load data block

This function can be used to load motion tasks from the DP-master into the motion-block memory (EEPROM) of the digifas ™.

Function call :
short int dp_w_fahrsatz(struct datensatz *fs)

Description :
The function *dp_w_fahrsatz* transfers the motion block which is to be loaded with the aid of two telegrams, each with 12 bytes of user data, and stores this in the motion-block memory of the amplifier.
The successful transmission of each telegram is verified by a reply telegram, which is generated by the interface of the amplifier.

Set parameters :
fs_**verw**
fs_**ds_nummer**
fs_**art**
fs_**s_soll**
fs_**v_soll**
fs_**a_beschl**
fs_**a_brems**

Return parameters :
The correct execution can be evaluated by means of the return values of the DP-function, so that a verification of the transmission is achieved.

VI.2.7.3.2 Read data block

This function can be used to read out motion tasks from the motion-block memory (EEPROM) of the digifas™.

Function call :

short int dp_r_fahrsatz(stuct datensatz *.fs)

Description :

The function *dp_r_fahrsatz* reads a motion block with the aid of two telegrams, each with 12 bytes of user data, from the motion-block memory of the controller.
The successful transmission of each telegram is verified by a reply telegram, which is generated by the interface of the controller.

Set parameters :	**Return parameters :**
fs_**verw**	fs_verw
fs_**ds_nummer**	fs_ds_nummer
fs_art	fs_**art**
fs_s_soll	fs_**s_soll**
fs_v_soll	fs_**v_soll**
fs_a_beschl	fs_**a_beschl**
fs_a_brems	fs_**a_brems**

VI.2.7.3.3 Read out actual values and the error register

This function can be used to read out the actual position value, contouring error, rotational angle, and the error register.

Function call :

short int dp_r_istwert(struct istwerte * i_wert)

Description :

The function *dp_r_istwert* reads out the actual values described above from the controller, depending on the actual-value ID. The section “Description of the transfer functions of the function blocks” describes which actual values can be read with which actual-value IDs.

Set parameters :	**Return parameters :**
i_wert_**verw**	i_wert_verw
i_wert_**iwert_ken**	i_wert_iwert
i_wert_s_ist	i_ken i_wert_**s_ist**
i_wert_v_ist	i_wert_**v_ist**
i_wert_inc_winkel	i_wert_**inc_winkel**
i_wert_s_fehler	i_wert_**s_fehler**
i_wert_fehlerreg	i_wert_**fehlerreg**
i_wert_zustandsreg	i_wert_**zustandsreg**

VI.2.7.3.4 Read / write parameters, execute commands, request actual values

Function :

short int dp_rw_param(struct para *pa)

Description :

Depending on the parameter number (multiplexer), the function *dp_rw_param* can be used to read or write a parameter, carry out a command (save to EEPROM) or request actual values.

Set parameters :

pa_**verw**
pa_**para_nummer**
pa_**datatyp**
pa_**rw_acc**
*pa_**datum**

Return parameters :

pa_verw
pa_para_nummer
pa_datatyp
pa_rw_acc
*pa_**datum**

VI.2.7.4 Auxiliary function for the control word SW

This help function is already implemented in several DP-function which show command characteristics (*dp_start_fs_eeprom, dp_ref_fahrt, dp_tippen*), but can also be called separately in order to modify the SW.

Function call :

char dp_ctrl_word (unsigned char arg_number, unsigned short int first_mask,...)

Description :

The function *dp_ctrl_word* works with a dynamic call-parameter list. The first call parameter indicates the number of following call parameters. For instance, the following control commands can be passed to the function simultaneously : ackowledgement of the response monitoring (bit 7 = 1), motion task STOP (bit 10 = 0), set reference point (bit 12 =1).
The function call for the three control commands then looks like this :

dp_ctrl_word(3,CW_QUITERROR, CW_FSSTOP, CW_SETREF,);

The sequence of the control commands within the parameter does not matter. The plausibility of the commands must be checked by the user.

Set parameters : (see also the *include* file USER_ST.H)

Constant definition	Description	Constant definition	Description
CW_ENABLE	controller enable (logic AND with the enable signal)	CW_TIP1ON	jog 1 ON[2),3)]
CW_DISABLE	controller disable	CW_TIP1OFF	jog 1 OFF
CW_SHALTOFF	enable (rapid stop) [4)]	CW_TIP2ON	jog 2 ON[2),3)]
CW_SHALTON	rapid stop	CW_TIP2OFF	jog 2 OFF
CW_BRREALESE	if in project: release brake	CW_FSSTART	motion task START[2),3)]
CW_BRNOTREL	if in project: do not release brake	CW_FSSTOP	motion task STOP
CW_NOISTOP	enable motion task	CW_REFSTART	ref. traverse START[2)]
CW_ISTOP	pause	CW_REFSTOP	ref. traverse STOP
CW_QUITERROR	contouring error / acknowledge response monitoring[1)]	CW_SETREF	set reference point
CW_RQUITERROR	cancel CW_QUITERROR flag	SRW_RSETREF	cancel ref. point flag

[1)] Servo-amplifier **must be disabled.**
[2)] Bit 14 (task running) in the status word must **not** be set.
[3)] Reference point **must** be set.
[4)] This bit **must** be set to be able to carry out a motion task.

Note :

The success of the transmission is **not** checked by this function. Care must be taken that the number of control commands (first call parameter) matches the number of control commands which has actually been entered, since the checking of the call-parameter list can only be carried out in a limited form.

The function *dp_ctrl_word* has the following return values :

Return values	Descriptions
0	function performed correctly
1	the number of arguments must be more than 0
2	erroneous or undefined bit

Note :

To initiate control functions, this function should be called before an interface function, because in every DP-function the control word is passed to the communication process.

VI.2.8 Description of the files for the function-block library

The function-block library is put together from a number of files (*Sources,Includes*). The files are described briefly below.

DP_SINEC.CPP

This file includes transmit and receive routines which are called exclusively by the DP-function blocks. The transmit and receive routines form the interface to the “SINEC DP-interface”. In the transmit routine (*send_dp_telegramm*) the parameter process-data object (PPO) is assembled for the digifas™ and passed to the communication process.
The receive routine (*rec_dp_telegramm*) reads in and evaluates the entry data from the DP-master. There is a receive loop for this purpose, which is terminated by the time-out which has been predetermined by the user. In this loop, the received objects are filtered by subindex, parameter number and response ID, so that the corresponding answer to a “Request” (e.g a request for an actual value) can be evaluated.

DP_DIGFKT.CPP

This source file includes all the DP-function blocks for the digifas™, which have already been described above.

DP_HFKT.CPP

This source file includes a function (*dp_ctrl_word*), which modifies the control word. The function is used by the DP-function blocks, but it can also be called directly by the user.

BCD_HEX.CPP

This source file includes conversion functions which convert a signed 32-bit number (long integer) into BCD-format (*hex2bcd*) or a BCD- number into a signed 32-bit number (*bcd2hex*). These conversion functions are necessary because 32-bit integer numbers (positions) are transmitted in the BCD-format.

USER_ST.H

This *include* file contains definitions of constants for the control word and the declarations of the transfer structures (see section “Description of the transfer structures of the function blocks”).

DP_DIGFKT.H

This *include* file contains function declarations for the DP-function blocks.

DP_SINEC.H

This *include* file contains the structure declaration *dpsinec*, which is used to assemble the PPO.

TE_SRUCT.H

This include file contains constant definitions for the assembly of the PP and the evaluation of the status word.

VI.3 Sample parameterization der ET200

Parameters for a sample system with an IM308B(C) module and a digifas™ .

VI.3.1 Parameterization of the master module

VI.3.1.1 Bus parameters

Station number IM308B(C)	1	1 or 2 (only for second master station with IM308B(C))
PG on Bus	N	if Y, PG received station number 0 and had higher priority
seecond IM308B(C)	N	see station number; Y, only when second master with IM308B(C)is available
further active stations	N	for parameterization of other stations, for digifas™ : configuration only in network
baud rate	500kbaud	9.6/19.2/93.75/187.5/500/1500 kBaud
bus profile	DP standard	

VI.3.1.2 AG parameters

CPU type	B	A:S5-135/150U; B:S5-115U(CPU941,942,943); C:S5-115U(CPU944); D:S5-115U(CPU945); E:S5-155U
response monitoring	J	Y or N; if Y, slave moves to safe state after the preset time, all outputs are disabled
QVZ on station failure	J	Y or N; if Y, CPU enters RUN mode only when all the slaves have properly acknowledged
diagnosis	J	IM308B : Y or N; if Y, the IM308B provides a diagnosis word for the CPU **IM308C : N, since no diagnosis is provided**
diagnosis address	252	
tiling	N	Y or N; if Y, enter the no. and kind of tiles
ready signal		only relevant for tiled addressing, is only set to Y by the last IM308B(C)
run-up delay	20s	20-1200s, use only if QVZ = Y
response monitoring	0.1s	system recommendation
minimum cyle time of the system	2ms	

VI.3.2 Configuration of digifas™ - PROFIBUS as a slave

Station number : 3 Range: P
Station type : DIGIFAS_1.SPC Station designation : Test drive
Minimum cycle time: 2 ms

Position	Channel	Input / Symbol	Output / Symbol
0	4AX	EW 128	AW 128
		EW 130	AW 130
		EW 132	AW 132
		EW 134	AW 134
1	2AX	EW 136	AW 136
		EW 138	AW 138

The station type DIGIFAS_1.SPC and the 4AX und 2AX modules are provided through the ET200 parameter file SPCDI1TD.200, included in the delivery.
Further information can be found in the ET200 manual from Siemens.

VI.4 Sample program

VI.4.1 General notes

The following sample program is intended to demonstrate how – with very little programming effort – a digifas™ servo drive can be remotely controlled by PROFIBUS CONNECT.

The program is deliberately written without any additional interlocks or monitors.

The formal operands have been deliberately so chosen that the control signals can be generated by, for instance, an input simulator. Using the PC function “Control variable”, the motion-block data can be placed in marker words and status signals can be displayed.

The significance of the variables and parameters can be found in the commentary block for OB1 or the FB-DOKU.TXT file on the accompanying diskette.

The hardware and software requirements correspond to the system requirements described in Section I.4 .

Linear addressing (LP) was selected in DB100 (see page VI-28).

VI.4.2 Listing for sample program

NETWORK 1 *0000*

FB 190

DIGISTAT

101,1	NUDB	ZSW	MW 190
EB 8	STB1	KFEH	M 10.0
		BUSY	M 10.1
		OFEH	M 10.2
		MFEH	M 10.3
		STAT	MW 50

NETWORK 2 *000B* *MANUAL CONTROL OF DIGIFAS*

FB 191

DIGIHAND

E 9.0	TIP1
E 9.1	TIP2
E 9.2	FREF
M 0.0	SREF
MW 124	VABS

NETWORK 3 *0013* *START DIRECT TASK*

FB 195

DIGIDIRE

MW 120	POSH	STAT	MW 50
MW 122	POSL		
MW 124	VABS		
MB 126	RAMP		
E 9.5	STRT		

NETWORK 4 *001C* *PROCESS MOTION BLOCK*

FB 196

DIGISATZ

MW 100	FSNR	STAT	MW 50
E 9.7	STRT		

NETWORK 5 *0022* *LOAD MOTION BLOCK*

FB 197

DIGILADE

MW 100	FSNR	STAT	MW 50
MW 120	POSH	STEP	MB 130
MW 122	POSL	FERT	M 11.0
MW 124	VABS		
MW 126	RAMP		
MB 128	FOLG		
M 129.0	BRMS		
E 9.3	SEND		

NETWORK 6 *002D* *READ BACK MOTION BLOCK*

FB 198

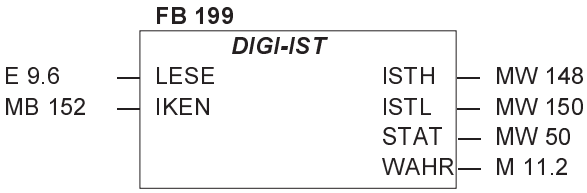
DIGILESE

MW 100	FSNR	POSH	MW 140
E 10.0	MFOL	POSL	MW 142
E 9.4	LESE	VABS	MW 144
		RAMP	MW 146
		FOLG	MB 154
		BRMS	M 155.0
		STAT	MW 50
		STEP	MB 156
		FERT	M 11.1

NETWORK 7 *0038* *SWITCH ON ACTUAL-VALUE READING*

FB 199

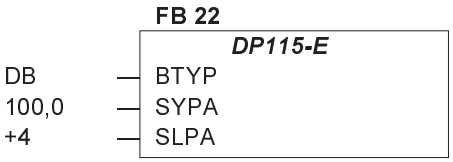
	DIGI-IST		
E 9.6	LESE	ISTH	MW 148
MB 152	IKEN	ISTL	MW 150
		STAT	MW 50
		WAHR	M 11.2



NETWORK 8 *0047* *COMMUNICATION: RECEIVE*

FB 22

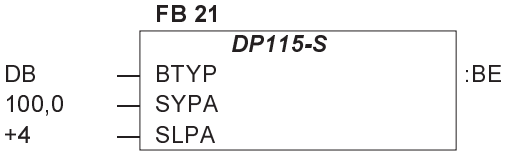
	DP115-E
DB	BTYP
100,0	SYPA
+4	SLPA



NETZWORK 9 *0041* *COMMUNICATION: TRANSMIT*

FB 21

	DP115-S	
DB	BTYP	:BE
100,0	SYPA	
+4	SLPA	



DB100		***DB parameter sets***	
0:	KH = 0100;	diagnosis per standard ?	
1:	KC ='LP';	system param.: addressing mode	
2:	KY = 001,000;	with diagnosis, base tile (page frame) for tiling	
3:	KF = +00001;	repeats counter	
4:	KF = +00001;	slave 1 : PPO-type	
5:	KF = +00003;	station number	
6:	KY = 000,101;	DB/DX , DB user data	
7:	KF = +00001;	address KCW	
8:	KY = 000,128;	start tile, start byte	
9:			

DB101		***DB user data PPO 1***	
0:	KH = 0000;	reserved	
1:	KM = 00000000 00000000;	slave 1 : user control word 1 KCW	
2:	KM = 00000000 00000000;	control word 2 (internal)	
3:	KM = 00000000 00000000;	user Anzw	
4:	KF = +00000;	repeats counter (internal)	
5:	KY = 000,000;	Pafe 1 / Pafe 2	
6:	KH = 0000;	PKW internal :	PKE
7:	KH = 0000;		IND
8:	KH = 0000;		PWE 1
9:	KH = 0000;		PWE 2
10:	KH = 0000;	transmit port :	PKE
11:	KH = 0000;		IND
12:	KH = 0000;		PWE 1
13:	KH = 0000;		PWE 2
14:	KH = 0000;		CW
15:	KH = 0000;		HSW
16:	KH = 0000;	receive port :	PKE
17:	KH = 0000;		IND
18:	KH = 0000;		PWE 1
19:	KH = 0000;		PWE 2
20:	KH = 0000;		ZSW
21:	KH = 0000;		HIW
22:			

VI.5 Parameter form for parameter CONNECT

Display text	Dim	Min.	Max.	Default	Actual value
Kp	—	0	8	**1**	
Ff	—	0	2	**1**	
Control from	—	BUS	PC	**BUS**	
Mode	—	position	n-analog/I-analog/ n-digit/I-digit	**posi-tion**	
Axis type	—	round	linear	**round**	
Count direction	—	positive	negative	**pos.**	
Resolution	mm/turn	0.01	999.99	**10**	
v_max	mm/s	0	rated speed conversion	**250**	
Ramp type	—	trapeze	sine²	**trap.**	
t_acc_min	ms	10	2550	**100**	
t_emergency	ms	10	2550	**50**	
Reference traverse	—	1+/1-/2+/2-/3+/3-	4+/4-/5+/5-	**1-**	
Zero point offset	mm	0	resolution	**0**	
Reference offset	mm	-20% resolution	max. path*	**0**	
Limit-switch 2	mm	0	max. path*	**300**	
Limit-switch 1	mm	-20% resolution	max. path*	**-2**	
In Position	mm	0	10% resolution	**0.1**	
Contouring errorr	mm	0	49% resolution	**2**	
Response monitoring	ms	0	5000	**5000**	
Station address	—	3	124	**3**	
Jog 1	mm/s	0	v_max	**50**	
Jog 2	mm/s	0	v_max	**100**	

* max. path = 32767 * resolution < 999 999.99 mm

Customer Cabinet no. Instrument no.

Place, Date Signature

VI.6 Index

	Text	Page
A	abbreviations	I-2
	amplifier ID	IV-8
	axis type	V-3
B	BS7200: CONNECT menu page	V-1
	BS7200: SERVICE menu page	V-1
	bus cable	I-4
	bus medium	I-2
C	commissioning	II-5
	CONNECT parameter form	VI-29
	connection diagram	II-2
	connection methods	II-1
	contouring error	V-14
	control from	V-2
	count direction	V-3
D	data format for parameters	IV-10
	digital speed setpoint	IV-14
	direct order	IV-3
E	error index	V-19
	error log	V-19
	error statistics	V-19
	extended PNU	IV-11
F	FB	VI-8
	FB installation	VI-8
	FB system requirements	VI-8
	FB auxiliary function for SW	VI-23
	FB control command	VI-17
	FB direct motion task	VI-19
	FB error IDs	VI-15
	FB files	VI-24
	FB function blocks	VI-16
	FB jog mode	VI-18
	FB load data block	VI-20
	FB parameter schreiben	VI-22
	FB read actual values	VI-21
	FB read data block	VI-21
	FB read error register	VI-21
	FB reference traverse	VI-18
	FB start motion task	VI-20
	FB structure “Datensatz”	VI-11
	FB structure “Istwert”	VI-12
	FB structure “Verwaltung”	VI-10
	FB structure to start motion block	VI-13
	FB structure: setting-up functions	VI-13
	FB transfer structure	VI-14
	FB190 DIGISTAT	VI-2
	FB191 DIGIHAND	VI-2
	FB195 DIGIDIRE	VI-3
	FB196 DIGISATZ	VI-3
	FB197 DIGILADE	VI-4
	FB198 DIGILESE	VI-5
	FB199 DIGI-IST	VI-6
	FB200 DIGIPARA	VI-7
	features	I-2
	Ff, feed-forward factor	V-2
	fieldbus monitor	V-1
	following task	IV-5
	frontal view	I-3
	function-block library (FB)	VI-8
	function-block package DIGI_DP1	VI-1
H	HIW	III-6
	HSW	III-6
I	IND	III-3
	index	VI-30
	InPosition	V-14
	installation	II-1
J	jog 1	V-15
	jog 2	V-15
	jog mode	IV-1
K	KE, motor voltage constant	V-18
	Kp position controller	V-2
L	L, motor inductance	V-18
	limit-switch 1	V-13
	limit-switch 2	V-13
	linear axis	II-6
	list of parameter nos.	IV-12
	load data blcok	IV-5
M	mode	IV-14
	motion task	V-14
	motion-block parameters, print	V-20
N	next no.	V-17
	no.	V-16
	numerical format	I-4
P	parameterization of the ET200	VI-25
	parameterizing the CP5412(A2)	II-3
	parameterizing the IM308B	II-3
	pause (next task)	V-17
	pin assignments	I-3
	PKE	III-2
	PLC-programming	II-4
	PPO	III-1
	profdemo.exe	VI-9
	PWE	III-3
R	ramp type	V-5
	read actual values	IV-7
	read data block	IV-6
	read status register	IV-8
	reference offset	V-13
	reference traverse	IV-2
	residual path processing	V-14
	resolution	V-4
	response monitoring	V-14
	rotary axis	II-8

	Text	Page
S	s_nom	V-16
	safety instructions	1-E
	sample program	VI-26
	sample telegrams	IV-15
	save amplifier parameters	IV-14
	set reference point	IV-1
	setting the amplifier parameters	IV-10
	sine²	V-5
	SINEC DP programming interface	II-4
	standard function blocks	II-4
	start a motion task	IV-4
	station address	V-15
	status register (SR)	IV-9
	STW	III-4
	symbolic labels	IV-11
	system requirements	I-2
T	t_acc	V-17
	t_acc_min	V-6
	t_brake	V-17
	t_emergency, t_acc_emergency	V-6
	Teach In	IV-15
	transmission procedure	I-2
	transmission STW to amplifier	IV-1
	transmission ZSW to master	IV-1
	type	V-16
	type of reference traverse	V-7
U	Use as directed	I-1
V	v_max	V-5
	v_nom	V-17
W	write/read amplifer parameters	IV-10
Z	zero point offset	V-13
	ZSW	III-5

Vertrieb und Service / Sales and Service / Agence et Services

Bundesrepublik Deutschland/
Germany/Allemagne

Seidel Servo Drives GmbH
Verkaufsniederlassung Nord
Dasselsbrucher Str. 49a
D-29227 Celle
Tel.: +49(0)5141 - 98 10 40
Fax: +49(0)5141 - 98 10 41

Seidel Servo Drives GmbH
Verkaufsniederlassung West
Wacholderstr. 40-42
D-40489 Düsseldorf
Tel.: +49(0)203 - 99 79 - 180
Fax: +49(0)203 - 99 79 - 118

Seidel Servo Drives GmbH
Verkaufsniederlassung Süd-West
Bruchsaler Str. 3
D-76646 Bruchsal-Untergrombach
Tel.: +49(0)7257 - 9 23 07
Fax: +49(0)7257 - 9 23 08

Seidel Servo Drives GmbH
Verkaufsniederlassung Süd-Ost
Viehmarktstr. 7a
D-82256 Fürstenfeldbruck
Tel.: +49(0)8141 - 34 90 94
Fax: +49(0)8141 - 34 90 95

Servo-Dyn
Technik GmbH
Münzgasse 10
D-01067 Dresden
Tel.: +49(0)351 - 49 05 793
Fax: +49(0)351 - 49 05 794

Dänemark/
Denmark/Danemark

DIGIMATIC
Ormhöjgaardvej 12-14
DK-8700 Horsens
Tel.: +45 - 76 26 12 00
Fax: +45 - 76 26 12 12

Finnland/
Finland/Finlande

Drivematic OY
Hevosenkenkä 4
FIN-28430 Pori
Tel.: +358 - 2 - 52 99 600
Fax: +358 - 2 - 52 99 610

Frankreich/
France/France

Seidel Servo Drives GmbH
Parc technologique St.Jacques
2 rue Pierre et Marie Curie
F-54320 Maxéville
Tel.: +33(0)3 83 95 44 80
Fax: +33(0)3 83 95 44 81

Großbritannien/
Great Britain/Royaume-Uni

Kollmorgen
PO Box 147, KEIGHLEY
West Yorkshire, BD21 3XE
Tel.: +44(0)15 35 - 60 76 88
Fax: +44(0)15 35 - 68 05 20
Heason Technologies Group
Claremont Lodge
Fontwell Avenue
Eastergate Chichester PO20 6RY
Tel.: +44(0)12 43 - 54 54 00
Fax: +44(0)12 43 - 54 45 90

Italien/
Italy/Italie

M.C.A. s.r.l.
Via f. Turati 21
I-20016 Pero (Mi)
Tel.: +39(0)02 - 33 91 04 50
Fax: +39(0)02 - 33 90 85 8

Niederlande/
Netherlands/Pays-Bas

Dynamic Drives
Jan van der Heydenstraat 24a
NL-2665 JA Bleiswijk
Tel.: +31(0)10 - 52 15 490
Fax: +31(0)10 - 52 18 994

Schweden/
Sweden/Suéde

S D T AB
SE-25467 Helsingborg
Tel.: +46(0)42 - 380 800
Fax: +46(0)42 - 380 813
Stockholm
SE-12030 Stockholm
Tel.: +46(0)8 - 640 77 30
Fax: +46(0)8 - 641 09 15
Göteborg
SE-42671 Västra Frölunda
Tel.: +46(0)31 - 69 62 60
Fax: +46(0)31 - 69 62 69

Schweiz/
Switzerland/Suisse

Seidel Servo Drives GmbH
Eggbühlstr. 14
CH-8050 Zürich
Tel.: +41(0)1 - 300 29 65
Fax: +41(0)1 - 300 29 66

Spanien/
Spain/Espagne

Comercial BROTOMATIC, S.L.
San Miguel de Acha, 2 Pab.3
E-01010 Vitoria
Tel.: +34 945 - 24 94 11
Fax: +34 945 - 22 78 32

Systempartner / System partners / Partenaires du syst me

Bundesrepublik Deutschland/
Germany/Allemagne

Werner P. Hermes
Ingenieurbüro
Turmstr. 23
40750 Langenfeld
Tel.: +49(0)212 - 65 10 55
Fax :+49(0)212 - 65 10 57

EAT GmbH
Elektronische Antriebstechnik
Abrichstr. 19
79108 Freiburg
Tel: +49(0)761 - 13 03 50
Fax:+49(0)761 - 13 03 555

IBK Ingenieurbüro Keßler GmbH
Dachtmisser Str. 10
21394 Kirchgellersen
Tel: +49(0)4135 - 12 88
Fax:+49(0)4135 - 14 33

Großbritannien/
Great Britain/Royaume-Uni

Motor Technology Ltd.
Unit 1
Chadkirk Industrial Estate
Otterspool Road
Romiley, Stockport
GB-Cheshire SK6 3LE
Tel.: +44(0)161 - 42 73 641
Fax : +44(0)161 - 42 71 306

Schweiz/Switzerland/Suisse

Bobry Servo Electronic AG
Zentralstr. 6
CH-6030 Ebikon
Tel.: +41(0)41- 440 - 77 22
Fax : +41(0)41 - 440 - 69 43

Frankreich/France/France

Transtechnik Servomécanismes
Z.A. Ahuy Suzon
17, Rue des Grandes Varennes
F-21121 AHUY
Tel.: +33(0)3 - 80 55 69 41
Fax: +33(0)3 - 80 53 93 63

Niederlande/
Netherlands/Pays-Bas

Kiwiet
Ingenieurbüro
Helenaveenseweg 35
NL-5985 NK Panningen (Grashoek)
Tel.: +31(0)77 - 30 76 661
Fax: +31(0)77 - 30 76 646

Italien/Italy/Italie

Servo Tecnica
Viale Lombardia 20
I-20095 Cusano Milanino (MI)
Tel.: +39(0)02 - 66 42 01
Fax: +39(0)02 - 66 40 10 20

Australien/Australia/Australie

FCR Motion Technology PTY. Ltd.
23 Mac Arthurs Road
Altona North, 3025
Melbourne/Australia
Tel.: +61 393 99 15 11
Fax: +61 393 99 14 31

Seidel Servo Drives GmbH

Hausanschrift

Wacholderstr. 40-42
D - 40489 Düsseldorf
Tel.: +49(0)203 - 99 79 - 0
Fax: +49(0)203 - 99 79 - 155
Internet : http://www.seidelservodrives.de

Postanschrift

Postfach 34 01 61
D-40440 Düsseldorf

Kollmorgen

201 Rock Road
Radford, VA 24141
Tel.: +1 540 - 639 - 24 95
Fax: +1 540 - 731 - 08 47
Internet : http://www.kollmorgen.com