

CAN CONNECT

Interface module integrated in
digifas™ 7100-CAN
and
digifas™ 7200-CAN



Previous editions

Edition	Comments
11 / 96	First edition, valid from software versions 6L40/6A40/6C40
08 / 97	New types of reference traverse, following task, new relative movements,error log, valid from SW 7C40

**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.: 83105

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	Drawing	Page
Contents		A
Safety notes		D
Directives and standards		E
CE-conformance		E
I General		
I.1 About this manual		I-1
I.2 "Use as directed" for the CAN CONNECT interface module		I-1
I.3 Abbreviations used in this manual.		I-2
I.4 Features of CAN CONNECT.		I-2
I.5 Numerical format.		I-3
I.6 Bus cable.		I-3
I.7 Response times to commands		I-4
I.7.1 Response time for a change of resolution		I-4
I.7.2 Response time for starting a direct motion task		I-4
I.7.3 Response time for changing the weighting factors 'Position' and 'Speed'		I-4
I.7.4 Response time for changing the final limit speed (tacho feedback).		I-4
I.8 Frontal view: digifas™ 7103 - CAN...7116-CAN	- A.4.020.4/4	I-5
I.9 Frontal view: digifas™ 7133-CAN...7150-CAN	- A.4.020.4/5	I-6
I.10 Frontal view: digifas™ 72xx - CAN	- A.4.020.4/3	I-7
II Installation / commissioning		
II.1 Assembly, installation.		II-1
II.1.1 Connection methods		II-1
II.1.2 Connection diagram CAN CONNECT.	- A.4.020.1/2	II-2
II.1.3 Pin assignments	- A.4.020.4/2	II-3
II.1.4 Code switch for station address	- A.4.020.4/6	II-4
II.2 Commissioning.		II-5
II.2.1 Linear axis.		II-6
II.2.1.1 Commissioning the position controller for a linear axis.		II-6
II.2.1.2 Notes on optimising the linear axis.		II-7
II.2.2 Rotary axis		II-8
II.2.2.1 Commissioning the position controller for a rotary axis		II-8
II.2.2.2 Notes on optimising the rotary axis.		II-9
III Software protocol		
III.1 General description of CAN.		III-1
III.1.1 Format of a Communication Object (COB)		III-1
III.1.2 Construction of the COB identifier		III-1
III.2 Definition of the Communication Objects		III-2
III.2.1 The Fault object		III-2
III.2.2 The Control object		III-4
III.2.2.1 Multiplexer for Control and Broadcast objects		III-5
III.2.3 The Status object		III-8
III.2.4 The Broadcast object		III-10
III.2.5 The Direct motion-task object		III-12

Contents	Drawing	Page
IV Description of the functions		
IV.1 Control functions		IV-1
IV.1.1 STOP function		IV-1
IV.1.2 CONTROLLER ENABLE ON/OFF		IV-1
IV.1.3 QUICK STOP ON/OFF		IV-1
IV.1.4 RELEASE/APPLY BRAKE function (in preparation)		IV-2
IV.1.5 PAUSE ON/OFF		IV-2
IV.1.6 ACKNOWLEDGE CONTOURING ERROR/RESPONSE MONITORING		IV-2
IV.1.7 BROADCAST ON/OFF function		IV-2
IV.1.8 Setting the status register mask		IV-3
IV.1.9 Activate baud rate recognition		IV-3
IV.1.10 ANALOG / DIGITAL setpoint entry		IV-4
IV.2 Digital speed and torque control		IV-5
IV.2.1 DIGITAL setpoint for the speed controller		IV-5
IV.2.2 DIGITAL setpoint for the torque controller		IV-5
IV.3 Description of the position and motion functions		IV-6
IV.3.1 Jog mode		IV-6
IV.3.2 Set reference point		IV-6
IV.3.3 Go to reference (reference traverse)		IV-7
IV.3.4 Start a motion task		IV-7
IV.3.5 Start a direct-motion task (ABSOLUTE / RELATIVE)		IV-8
IV.4 Actual value functions		IV-9
IV.5 Data transfer functions		IV-10
IV.5.1 Read/write a parameter of the local motion task		IV-10
IV.5.1.1 Read/write the position value		IV-10
IV.5.1.2 Read/write the speed value		IV-10
IV.5.1.3 Read/write the acceleration/braking ramp		IV-10
IV.5.1.4 Read/write the type of motion task absolute/relative		IV-11
IV.5.1.5 Save the local motion task to the EEPROM		IV-12
IV.5.2 Load a motion task from the EEPROM to the local buffer		IV-12
IV.5.3 Read the BAUD RATE		IV-12
IV.5.4 Read/write a controller parameter		IV-12
IV.5.5 Save the controller parameters to the EEPROM		IV-13
IV.5.6 Read/write the POSITION weighting factor		IV-13
IV.5.7 Read/write the SPEED weighting factor		IV-13
IV.5.8 Teach In		IV-14
IV.6 K_E , voltage constant of the motor		IV-14
IV.7 L , motor inductance		IV-14
IV.8 Fault signals		IV-15
IV.9 Error log		IV-16
IV.9.1 Error index		IV-16
IV.9.2 Error statistics		IV-16

Contents	Drawing	Page
V Operator Software		
V.1	General	V-1
V.2	Fieldbus monitor	V-2
V.3	Parameter description CONNECT menu page	V-3
V.3.1	Kp, P-gain	V-3
V.3.2	Ff, feed-forward factor	V-3
V.3.3	t_accel_min, maximum acceleration	V-3
V.3.4	v_max, maximum speed	V-3
V.3.5	t_emerg, maximum deceleration	V-3
V.3.6	Resolution	V-4
V.3.7	Count direction	V-5
V.3.8	Contouring error	V-5
V.3.9	In Position	V-5
V.3.10	Zero point offset	V-5
V.3.11	Limit switch 1	V-6
V.3.12	Limit switch 2	V-6
V.3.13	Axis type	- A.4.020,4/2 V-6
V.3.14	Controlled by	V-7
V.3.15	Response monitoring	V-7
V.3.16	Baud rate	V-7
V.3.17	Ramp type	V-7
V.3.18	Reference offset	V-8
V.3.19	Type of reference traverse	- A.4.024,3/8,10,13 V-9
V.3.20	Mode	V-14
VI Appendix		
VI.1	User notes and examples	VI-1
VI.1.1	Communication setup	VI-1
VI.1.1.1	Commissioning the CAN-bus master	VI-1
VI.1.1.2	Connection test master ↔ digifas™	VI-1
VI.1.2	Examples of various functions	VI-2
VI.1.2.1	Weighting factors	VI-2
VI.1.2.2	Reference traverse (“go to reference”)	VI-2
VI.1.2.3	Jog mode	VI-2
VI.1.2.4	Writing and performing a motion block	VI-3
VI.1.2.5	Direct motion tasks / examples of response times	VI-4
VI.1.2.6	Broadcast operation	VI-5
VI.1.2.7	Fault evaluation	VI-6
VI.2	Parameter form for CONNECT (operator software BS7200)	VI-7
VI.3	Index	VI-8

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 BGV A2
- ◆ 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 instructions 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/EEC
EC Low Voltage Directive	73/23/EEC

This page has been deliberately left blank

I General

I.1 About this manual

This manual describes the wiring, commissioning, range of functions and software-protocol of the CAN 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.
CAN Application Layer (CAL) for Industrial Applications	CiA e.V.	
CAN Specification Version 2.0	Philips Semiconductors	
ISO 11898 ...Controller area network (CAN) for high-speed communication		

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



Wiring	:	Electrically trained personnel
Programming	:	Software developers, CAN-BUS project engineers

We offer training and familiarisation courses on request.

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

The CAN-CONNECT interface module is permanently built into digital servo amplifiers of the digifas™ 7100-CAN and digifas™ 7200-CAN series.

These servo-amplifiers 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-CAN and digifas™ 7200-CAN series of servo amplifiers are **exclusively** intended for the closed-loop drive of brushless synchronous servomotors from the 6SM series of servomotors.

The CAN CONNECT interface module serves only to connect the servo amplifier to a master which has a CAN BUS 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.



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
Baud	Bit / s	FR	Fault/error Register
BCC	Checksum	ID	Identifier
CAL	Can Application Layer	i.V.	In preparation
CAN	Controller Area Network	MUX	Multiplexer
CMS	Can based Message Specification	P	proportional
COB	Communication Object	RTR	Remote Transmission request
EEPROM	Electrically Erasable/Programmable ROM	SR	Status Register

I.4 Features of CAN CONNECT

When working with the position controller integrated into the digifas™ 7100/7200 digital servo amplifier, the following functions are available:

Setup and general functions:

- go to reference point set reference point
- jogging, with variable speed
- motion with a digital setpoint

Positioning functions:

- execution of a motion task from the motion block store of the servo amplifier
- execution of a direct motion task

Data transfer functions:

- transmit a motion task to the motion block store of the servo amplifier
A motion task consists of the following elements:
 - » position setpoint (absolute task) or path setpoint (relative task)
 - » speed setpoint
 - » acceleration time, braking time
 - » motion block number
 - » type of task (absolute / relative)
 - » following task (with or without pause)
- read a motion task from the motion block store of the servo amplifier
- read actual values
- read the fault register
- read the status register
- read / write control parameters

System requirements:

- Servo amplifier digifas™ 71xx/72xx with CAN CONNECT interface module
- Master station with a CAN-BUS interface (e.g. PC with CAN interface)

Transmission procedure:

- Bus connection and bus medium : CAN-Standard ISO 11898 (CAN - high speed)
- transmission rate: max. 1MBit/s
possible settings of the servo amplifier: Auto, 20, 50, 100, 125, 250, 500, 1000 kbaud

I.5 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.6 Bus cable

In accordance with ISO 11898, you should use a bus cable with a characteristic impedance of 120 Ω. The usable cable length for reliable transmission reduces as the transmission rate increases. The following values which we measured can be used as a guide. They should not, however, be interpreted as limiting values:

Cable data:	characteristic impedance	100-120 Ω
	cable capacitance	max. 60 nF/km
	lead resistance (loop)	159.8 Ω/km

Cable length, depending on the transmission rate

Transmission rate / kbaud	max. cable length / m
1000	20
500	70
250	115

Longer transmission distances may be achieved with a lower cable capacitance (max. 30 nF/km) and a lower lead resistance (loop, 115 Ω/km).

(characteristic impedance $150 \pm 5\Omega \Rightarrow$ terminating resistor $150 \pm 5\Omega$).

For EMC reasons, the SubD connector housing must fulfil the following requirements:

- metal or metallised housing
- provision for connecting the cable shielding in the housing, with a large contact area.

I.7 Response times to commands

In general it may be said that a response time T_R (Timeout) between 0.4 ms and 2.5 ms is to be expected from transmission to reception of a Communication Object (command) by the master station, depending on the command and the transmission rate.

This response time takes into account the transmission time T_{tr} (across the bus in transmit and receive directions at 1Mbaud) and the processing time T_{pr} in the digifas™, i.e. $T_R = T_{tr} + T_{pr}$.

In extreme cases T_R may be > 2.5 ms. These special situations are described below.

I.7.1 Response time for a change of resolution

The resolution can be changed by Mux 79 (adapt SI-units to "Increments") or by Mux. 82 (adapt "Increments" to SI-units). With the aid of these commands, all the motion tasks (max. 120) which are defined in the motion task memory of the digifas™ are loaded from the EEPROM, converted, and then rewritten to the EEPROM. The following equation is a rough guide for the response time:

$$T_R < 25 \text{ ms} + n \cdot 320 \text{ ms} \quad (n = 0 \dots 120, \text{ number of the defined motion tasks})$$

During T_R , bit 31 of the status register remains set (see Chapter III.2.3). Since a long response time may be expected with 120 defined motion tasks, and every change of resolution causes a fresh operation in the EEPROM (the operational life of the EEPROM depends, among other things, on the no. of write operations), the resolution should only be changed for commissioning.

The conversion calculation is only performed for motion tasks where the speed is not zero. You can thus save calculating time by setting the speed to 0 mm/s for unused motion tasks.

I.7.2 Response time for starting a direct motion task

In order to start a direct motion task (motion task no. 0), the direct motion task must first be defined in the servo amplifier. There are two ways of doing this:

- 1.- Define and start each component, individually transmitted with different Mux :
motion task type, position, speed, ramps, start command

$$T_R < 8 \text{ ms}$$

- 2.- Direct motion task object (see Chapter III.2.5), combined with the commands :
"Start a direct motion task absolute/relative" (Mux 121/122)

$$T_R < 4 \text{ ms}$$

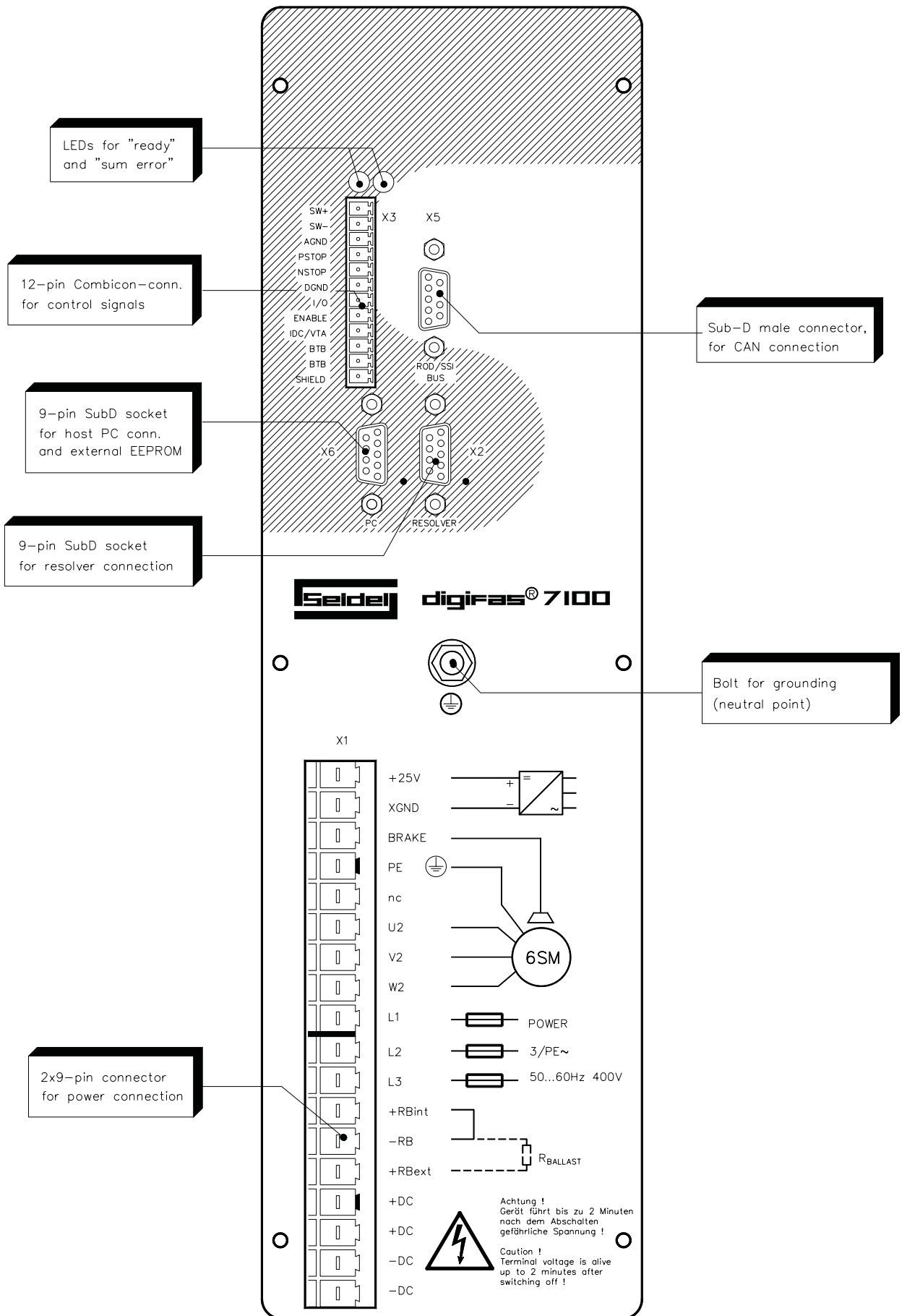
I.7.3 Response time for changing the weighting factors 'Position' and 'Speed'

When changing the weighting factors (Mux 126 and Mux 127) T_R is < 25 ms, since various calculations are necessary when this parameter is changed. For this reason the weighting factors should only be changed during commissioning.

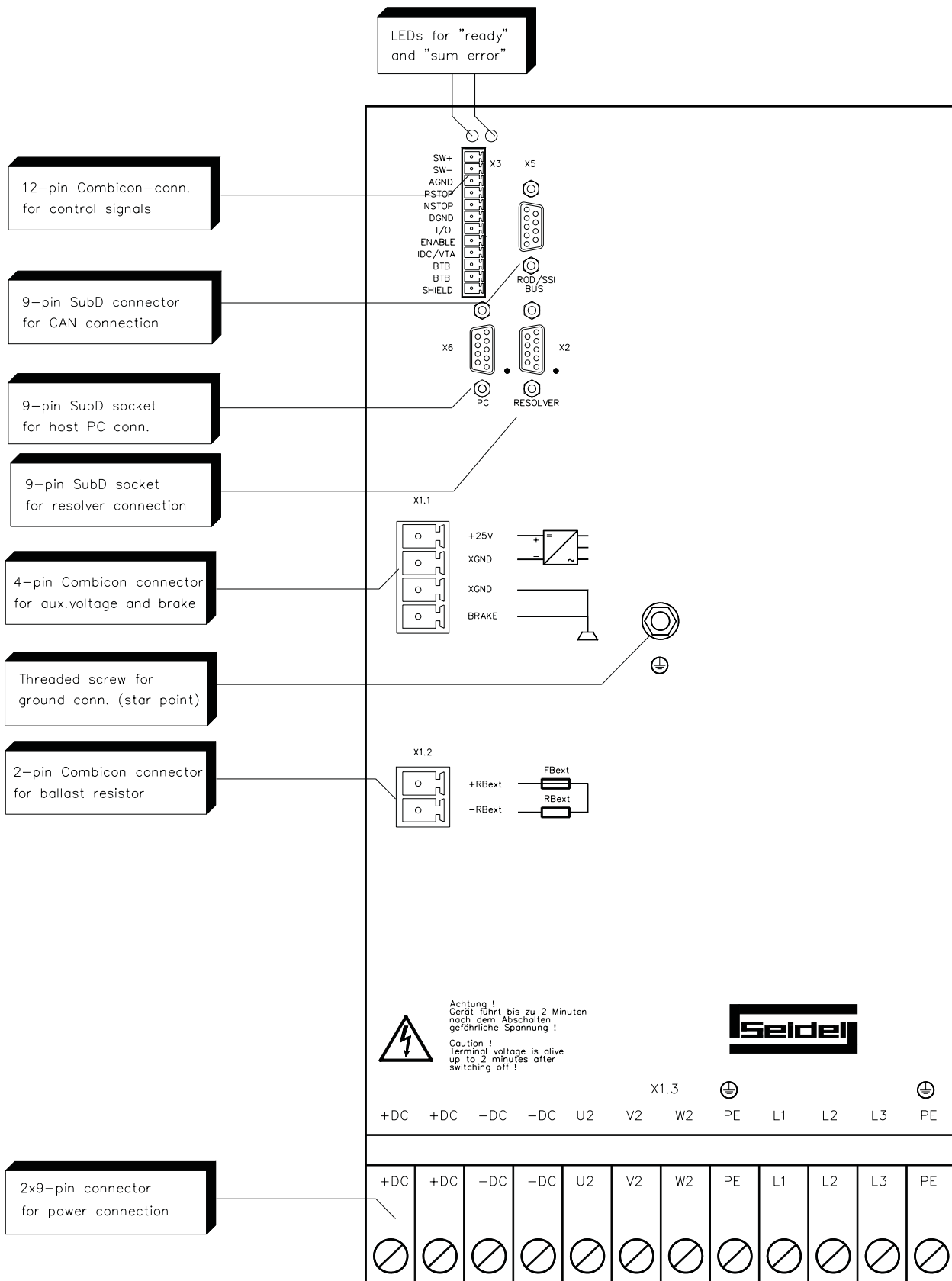
I.7.4 Response time for changing the final limit speed (tacho feedback)

When the final limit speed is changed (Mux 68) T_R is < 10 ms, since various calculations are necessary when this parameter is changed.

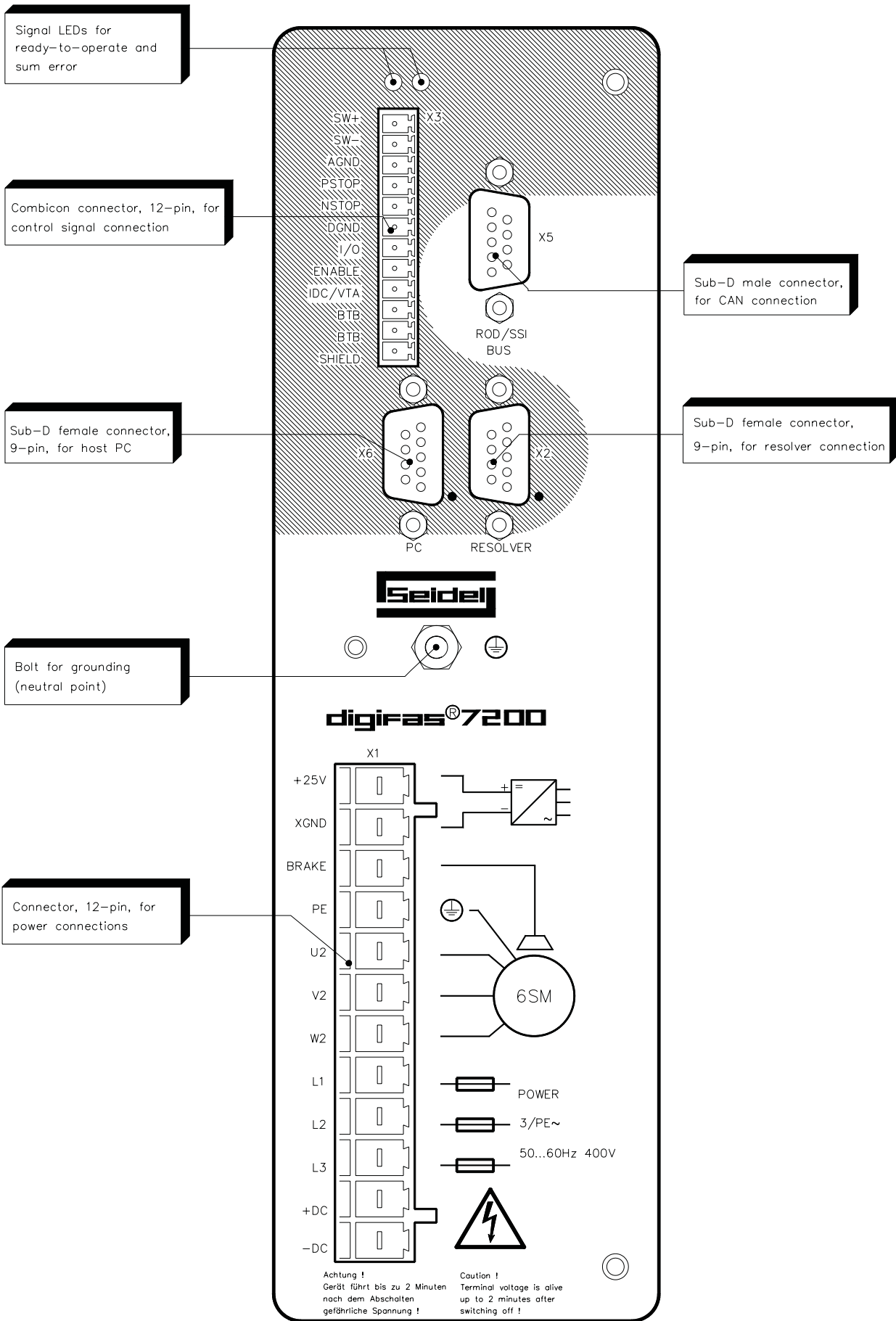
I.8 Frontal view: digifas™ 7103 - CAN...7116-CAN



I.9 Frontal view: digifas™ 7133-CAN...7150-CAN



I.10 Frontal view: digifas™ 72xx - CAN



This page has been deliberately left blank

II Installation / commissioning

II.1 Assembly, installation



Only install and wire up the equipment in a de-energized condition, i.e. neither the mains supply voltage nor the 25 V auxiliary supply nor the operating voltage of any other connected equipment may be switched on.

Take care that the switchgear cabinet is safely disconnected (lock-out, warning signs etc.). The individual voltages are switched on for the first time during commissioning.

Never disconnect the electrical connections to the servo amplifier while it is live. This could result in destruction of the electronics.

Residual charge in the capacitors can still have a dangerous level several minutes after switching off. Measure the voltage in the DC-link circuit and wait until the voltage has fallen below 40 V.

Even when the motor is not rotating, control and power leads can still be live.

Set up the station address for the servo amplifier on the CAN-bus (Chapter II.1.4). The station address must only be set while the equipment is de-energized.

Assemble the servo amplifier as described in the installation instructions for digifas™ 7100 or digifas™ 7200 respectively. Observe all the safety notes in the installation instructions which belong to the servo amplifier. Follow all the notes on mounting position, ambient conditions, wiring and fuse/overload protection.

To connect up the control, use the connection diagram in Chapter II.1.2 of this manual. The connections for the mains supply and the motor, as well as advice on correct system assembly for EMC, can be found in the installation instructions for the particular servo amplifier which is used.

The wiring for the setpoint input and the connection of the positioning interface in accordance with the wiring diagram in the installation instructions does not apply.

II.1.1 Connection methods

Mains, motor conn. : see Installation instructions for digifas™ 7100 resp. digifas™ 7200

Analog setpoints : no function

Digital control signals : see connection diagram in Chapter II.1.2, cable cross-section 0.5mm²

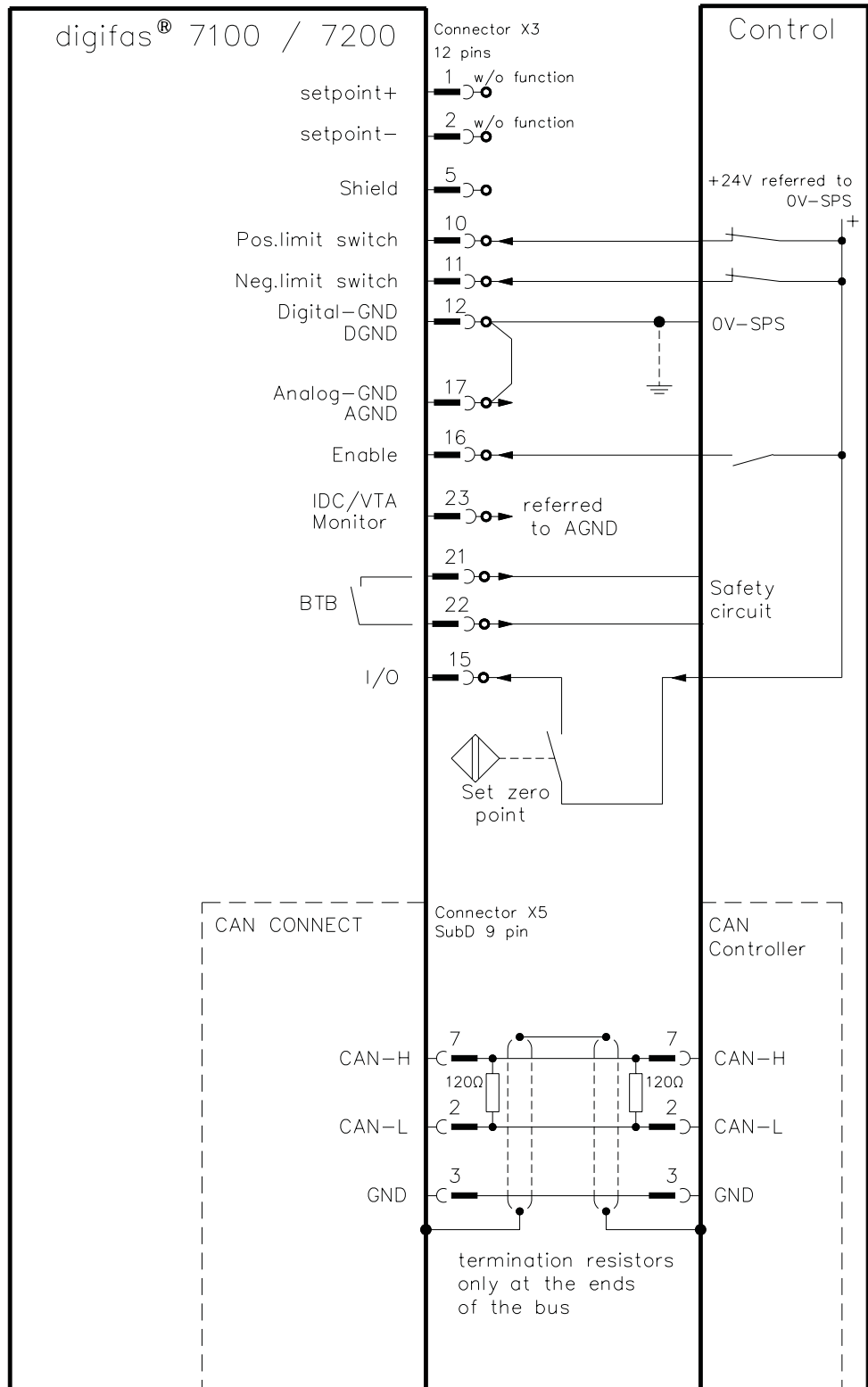
CAN-connection : see connection diagram in Chapter II.1.2

- use a bus cable specified to ISO11898
- for the cables which we supply, the following cable lengths are permissible, depending on the transmission rate:

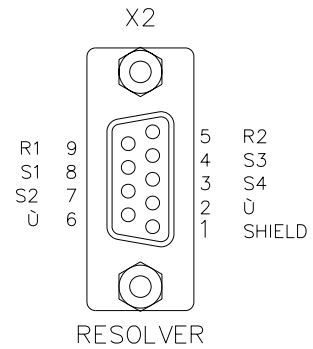
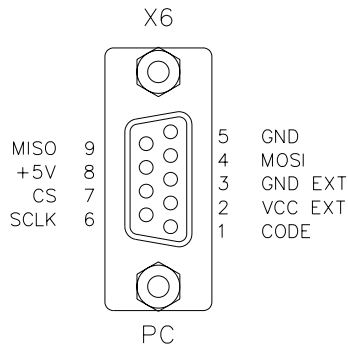
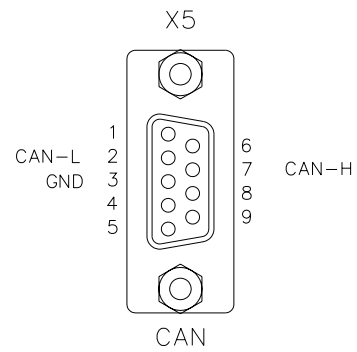
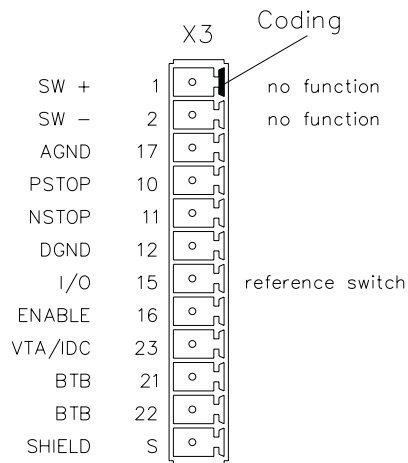
Transmission rate / kbaud	max. cable length / m
1000	20
500	70
250	115

II.1.2 Connection diagram CAN CONNECT

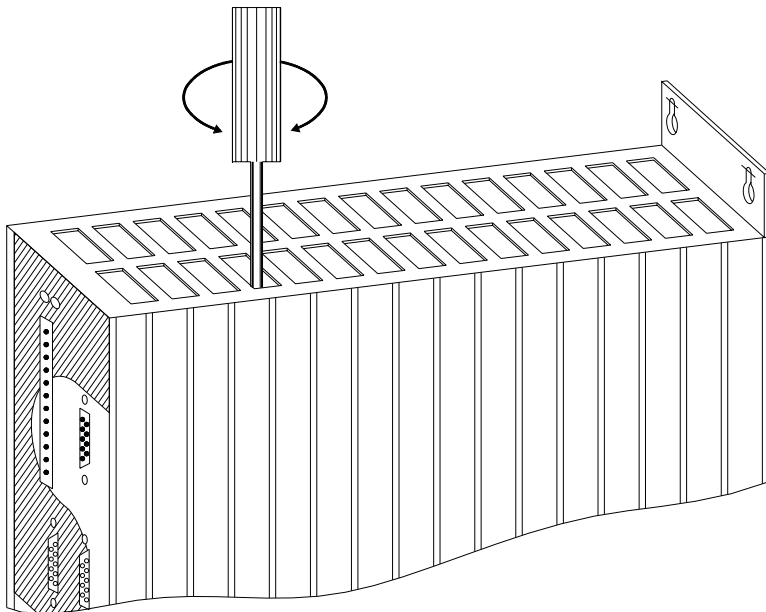
For motor connection and power feed see the installation instructions in the appropriate amplifier manual.



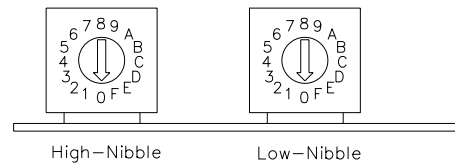
II.1.3 Pin assignments



II.1.4 Code switch for station address



Settings must be made only when mains and auxiliary supplies are switched off.



II.2 Commissioning



Only trained persons with extensive knowledge of control and drive technology are permitted to commission the servo amplifier.

Check assembly and installation

Check that all the safety notes in the installation manual for the servo amplifier and in this manual have been observed and implemented. Check the setting of the station address.

Connect PC, start BS7200

Use the operator software BS7200 for the parameter setting of the servo amplifier. Set the communication to "Control from the PC" (on the CONNECT menu page of the operator software).

Commission basic functions

Commission the basic functions of the servo amplifier and optimise the current and speed controllers. This part of the commissioning is described in greater detail in the installation and commissioning instructions for the servo amplifier which is used.

Save parameters

Save the control parameters after they have been optimised in the servo amplifier. Now switch over to "Control from the bus".

Commission bus communication

Remove the Enable signal (Terminal X3.16) and switch off the power supply to the servo amplifier.

The 24V DC auxiliary supply remains switched on.

Requirement: the software protocol described in Chapter III must be implemented on the master.

Adjust the baud rate of the digifas™ to match that of the master.

Please note that Chapter IV.1.1 also provides help on commissioning.

Test the communication

Recommendation: use Mux 118 to interrogate the status register of the digifas™.



Caution !

Make sure that neither persons nor machinery can be endangered by any unexpected movement of the drive.

Commission the position controller

Use the operator software BS7200 to switch the servo amplifier over to "Control from the PC" again.

Now commission the position controller, as described in the following Chapter II.2.1 (Linear axis) and II.2.2 (Rotary axis).

II.2.1 Linear axis



The parameter values given below for the commissioning of the position controller are not appropriate for all machines and 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 absolutely safe, and which, in no circumstances, can lead to damage to the machine. Read Chapter V as well !

II.2.1.1 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	0.1...0.3	zero offset	0
Ff	1	in position	window larger than required by the application
t_accel_min	twice the min. acceleration time permitted by the machine	contouring error	window larger than required by the application
v_max	less than 50% of the maximum load speed	limit sw. 1	30% of the permitted traverse path, calculated from zero
t_emerg	min. braking time permitted by the machine	limit sw. 2	70% of the permitted traverse path, calculated from zero
resolution	traverse path / motor turn	axis type	linear
count direction	according to the application	control from	PC

- 2.— Set the baud rate 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 from BUS”
- 5.— Switch off the 25V supply of the controller, then switch it on again. The new baud rate is only activated when the amplifier is switched on again.
- 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 point”. Check that the load is at the reference point.
- 8.— Optimise the control-loop behaviour (see Chapter II.2.1.2)
9. — Finally, set the following parameters according to the application: software limit switches 1 and 2, contouring error window, “In position” window, zero offset, t_accel_min, v_max, t_emerg

II.2.1.2 Notes on optimising the linear axis



The current, speed and position controllers function as a classic cascade control. A correct, i.e. stiff, setting of the inner speed controller is therefore a prerequisite for optimising the position controller.

1. — Traverse the drive, as a direct task, between two points at a low speed.
2. — Alter the Ff factor gradually, until the contouring error display (actual value display in 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). A negative contouring error means that the drive is leading the setpoint (oversteering) and Ff should be reduced.

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 speed under load 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 slightly increased.
4. — The Kp factor is gradually 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 on 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 previously entered must be adjusted !

If Irms and / or Ipeak are changed after the position controller has been optimised, then Kp and Ff must be adjusted !

II.2.2 Rotary axis



The parameter values given below for the commissioning of the position controller are not appropriate for all machines and 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 absolutely 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 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	0.1...0.3	zero offset	0
Ff	1	in position	window larger than required by the application
t_accel_min	twice the min. acceleration tiem permitted by the machine	contouring error	window larger than required by the application
v_max	less than 50% of the maximum load speed	limit sw. 1	n/a
t_emerg	min. braking time permitted by the machine	limit sw. 2	n/a
resolution	traverse path / motor turn	axis type	rotary
count direction	according to the application	control from	PC

- 2.— Set the baud rate 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 from BUS”
- 5.— Switch off the 25V supply of the controller, then switch it on again. The new baud rate is only activated when the amplifier is switched on again.
- 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 point”. Check that the load is at the reference point.
- 8.— Optimise the control-loop behaviour (see Chapter II.2.2.2)
9. — Finally, set the following parameters according to the application: contouring error window, “In position” window, zero offset, t_accel_min, v_max, t_emerg

II.2.2.2 Notes on optimising the rotary axis



The current, speed and position controllers function as a classic cascade control. A correct, i.e. stiff, setting of the inner speed controller is therefore a prerequisite for optimising the position controller.

1. — Traverse the drive, as a direct task, between two points at a low speed.
2. — Alter the Ff factor gradually, until the contouring error display (actual value display in 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). A negative contouring error means that the drive is leading the setpoint (oversteering) and Ff should be reduced.

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 speed under load 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 slightly increased.
4. — The Kp factor is gradually 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 on 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 previously entered must be adjusted !

If Irms and / or Ipeak are changed after the position controller has been optimised, then Kp and Ff must be adjusted !

This page has been deliberately left blank

III Software protocol

III.1 General description of CAN

The transmission method which is used here is defined in ISO 11898 (Controller Area Network CAN) for high-speed communication). The layer-1/2 protocol (Physical Layer/Data Link Layer) which is implemented in all CAN modules provides the requirement for data.

Data transport or data request is made by means of a data telegram (Data Frame) with up to 8 bytes of user data. An additional data request telegram (Remote Frame) is available for requesting data (status and fault register).

Both Data Frame and Remote Frame are described as Communication Objects (COB) in the sections below.

In the CAN-CONNECT software protocol 5 to 8 bytes of user data can be contained in a Communication Object, depending on its type. Communication Objects are labelled by an 11 bit Identifier (COB-ID). The COB identifier also determines the priority of an object.

A layer-7 protocol (Application Layer) was developed, in order to decouple the application from the communication. The service elements provided by the application layer make it possible to implement an application which is spread across the network. These service elements are described in the CAN Application Layer (CAL) for Industrial Applications. One of the four service elements which are described in CAL is the CAN-based Message Specification (CMS). The variable protocols which are described in CMS, called COBs here, are the basis of the CAN CONNECT software protocol.

III.1.1 Format of a Communication Object (COB)

S O M	COB-ID	R T R	CTRL	Data Segment	CRC	A C K	EOM
-------------	--------	-------------	------	--------------	-----	-------------	-----

SOM	Start of message
COB-ID	COB Identifier (11 bits)
RTR	Remote Transmission Request
CTRL	Control Field (u.a. Data Length Code)
Data Segment	0...8 bytes (Data-COB)
0	byte (Remote-COB)
CRC	Check sequence
ACK	Acknowledge Slot
EOM	End of message

III.1.2 Construction of the COB identifier

10	9	8	7	6	5	4	3	2	1	0
----	---	---	---	---	---	---	---	---	---	---

Bit 0- 2	Number of the COB defined in the server
Bit 3-10	Station number (range 0 ... 253; set by a hexadecimal-coded rotary switch in the servo amplifier, see Chapter II.1.4)

Warning: Station addresses 254 and 255 must not be used.

III.2 Definition of the Communication Objects

The multiplexer-value in a COB is interpreted as a command (range 0 ... 127). Five COBs have been defined within digifas™ 7100/7200 :

- Fault object**
- Control object**
- Status object**
- Broadcast object** (not CMS-specific)
- Direct motion task object**

III.2.1 The Fault object

The fault variable contains the current status of the digifas™ fault register. The fault variable is automatically transmitted (“notify event protocol”) for every change of the contents of the variable. In addition, the current contents of the fault variable (INTEGER32) can be read at any time (“read event protocol”; requested by RTR) or read out with the aid of the Control object, using Mux 119.

If there is a fault in the fault register, the amplifier will be in fault mode (red fault-LED lit up). You can only reset the fault register by switching off the 24V auxiliary supply of the servo amplifier.

Exception: Fault RESPONSE MONITORING

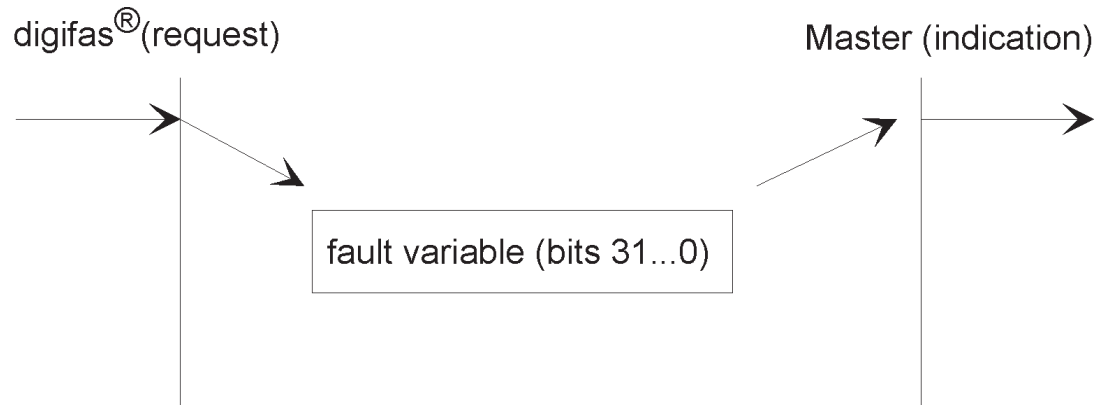
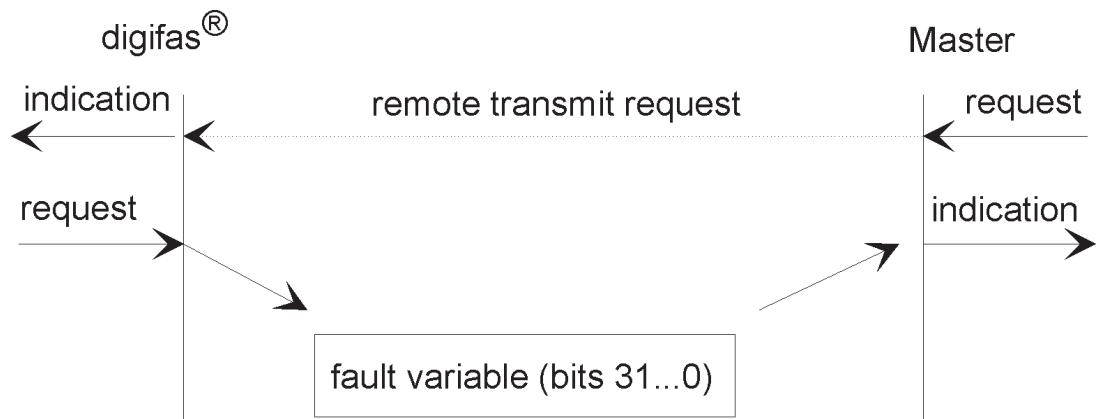
This fault does not affect the operational readiness of the amplifier, but only the ability to carry out commands. Commands which are performable with an activated response monitoring are labelled with an “a” in Chapter II.2.2.1 . The command “Acknowledge contouring error and response monitoring” resets the corresponding bit in the fault register.

Classification:

Object type: uncontrolled event (notify event protocol), stored event (read event protocol)
 Access type: read only
 Data type: INTEGER32
 COB: ID2=0, ID1=0, ID0=0

The following bit assignments are valid for the fault register and the fault variable :

Bit no.	Description	Fault	no fault
16...31	not used	-	-
15	Motor cable	1	0
14	Resolver fault	1	0
13	Auxiliary supply fault	1	0
12	Output stage fault	1	0
11	Brake fault	1	0
10	Mains fault, output stage	1	0
9	Overvoltage	1	0
8	Undervoltage	1	0
7	Earth short	1	0
6	DPR fault	1	0
5	EEPROM fault 2 (parallel)	1	0
4	EEPROM fault 1 (serial)	1	0
3	Response monitoring	1	0
2	Motor temperature too high	1	0
1	Internal temperature too high	1	0
0	Heat sink temperature too high	1	0

Description of the “notify event protocol”:**Description of the “read event protocol”:**

III.2.2 The Control object

The control variable can be used to trigger certain slave functions (write access) or to read data such as the control parameters and actual values (read access). The control variable thus uses the protocol type "Read/Write Variable Protocol (Multiplex Variable)".

In the request direction (see diagram) the data frame is made up of the read/write flag (c), the multiplexer (Mux) and 4 bytes of transmission data (control variable).

In the response direction the data frame is made up of a result bit (r), multiplexer (Mux) and 4 bytes of response data.

The 4 bytes of transmit or receive data are loaded flush-right with the corresponding data type of a variable (see Chapter III.2.2.1).

The result bit shows either a successful (r = 0) or an faulty (r = 1) access to data.

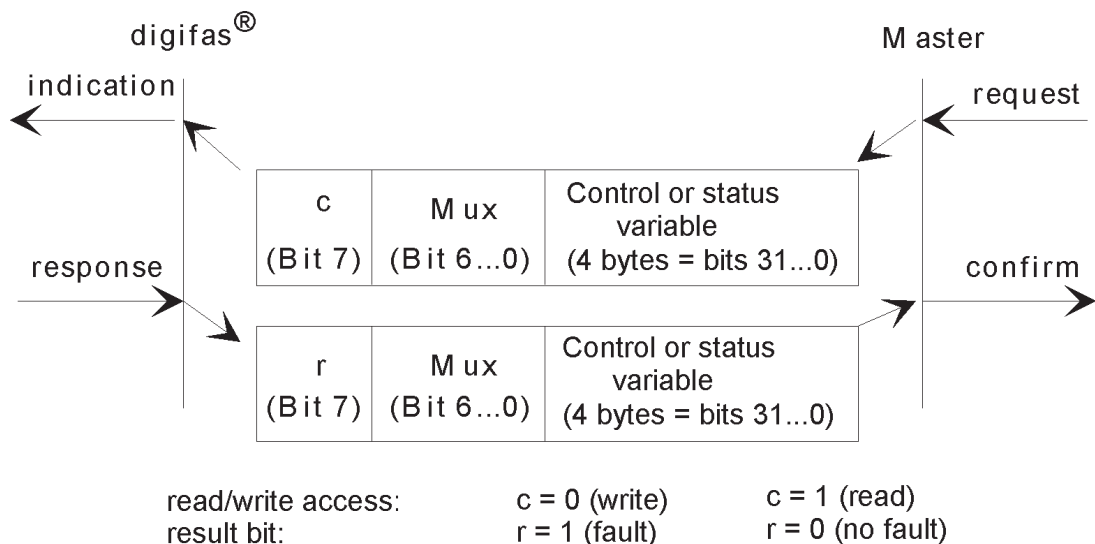
In the event of a fault, the fault variable which follows immediately after the multiplexer is only 1 byte long (usually 4 bytes) and contains a fault number (see Chapter III.2.1). If there is no fault, a mirrored answer is generated in the slave (digifas™).

A detailed description of control variables and multiplexers can be found in Chapter III.2.2.1.

Classification:

Object type: multiplex variable
 Access type: reading/writing
 Data: 4 bytes
 Multiplexer: 0 ... 127
 COB: ID2=0, ID1=0, ID0=1 for transmit direction (slave ⇒ master)
 ID2=0, ID1=1, ID0=0 for receive direction (master ⇒ slave)

Description "Read/Write Variable Protocol (Multiplex Variable)":



Response

(write response):

result bit (r) = 0 (no fault)
 multiplexer = n (0 ≤ n ≤ 127)
 control variable = max. 4 bytes

fault event:

result bit (r) = 1 (fault)
 multiplexer = 128 + n (128 is caused by the result bit r = 1)
 control variable = fault number

III.2.2.1 Multiplexer for Control and Broadcast objects

The following table includes all the multiplexers, numerically ordered, and with a brief description.

An empty 4-byte data frame is transmitted for those multiplexers which do not have any data type listed.

Abbreviations in the "Access" column

The "Access" column shows which access (e.g. read/write) is permitted on the bus, and under which conditions (e.g. disable). The following table shows the abbreviations which are used for the access column.

Abbreviation	Description
a	Command also possible when response monitoring is activated
d	Command only possible when output stage is switched off (disable) The "disable" must be made with Mux = 3H
w	Write access
r	Read access
Int.	Integer
D	Decimal system
H	Hexadecimal system
Mux	Multiplexer

Assignment of the symbolic labels marked with "*" to the parameter values in digifas™

The short descriptions which are marked by a "*" are symbolic labels. The correspondence of these labels is shown in the table "Assignment of the symbolic labels marked with "*" to the parameter values in digifas™" (see Weighting Factors). These parameter values are the data to be transmitted (8-bit char.). The following table explains the correspondence between the labels and the parameter values in the digifas™.

Label	Mux	0	1	2	3	4	5	6	7	8	9	10
Analog/digital setpoint entry	48	position	n analog	l analog	n digital	l digital	—	—	—	—	—	—
I/O	55	Reset	1:1	Int.Off	l²t	Ballast	Reference	Ipeak x%	ROD/ SSI	Netz BTB	Soll /8	INPOS2
Brake	56	no	yes	Netz/BTB	—	—	—	—	—	—	—	—
Speed steadiness	62	off	on	—	—	—	—	—	—	—	—	—
Limit switch	80	off	on	stop	—	—	—	—	—	—	—	—
DC monitor	81	tacho	current	s_fault	l_soll	—	—	—	—	—	—	—
No. of motor poles	86	2	4	6	8	10	12	—	—	—	—	—
No. of resolver poles	91	2	—	—	—	—	—	—	—	—	—	—
Select language	92	German	English	French	—	—	—	—	—	—	—	—
Count direction	97	negative	positive	—	—	—	—	—	—	—	—	—
Select ballast	98	internal	external	—	—	—	—	—	—	—	—	—
Axis type	101	rotary	linear	—	—	—	—	—	—	—	—	—
Control from	102	BUS	PC	—	—	—	—	—	—	—	—	—
Baud rate / kbaud	105	auto	20	50	100	125	250	500	1000	—	—	—
Ramp type	106	Trapeze	Sine²	—	—	—	—	—	—	—	—	—
Type of reference traverse	107	1-	1+	2-	2+	3-	3+	4-	4+	5-	5+	—
Position weighting	126	0	1	2	3	—	—	—	—	—	—	—
Speed weighting	127	0	1	2	3	—	—	—	—	—	—	—

Symbolic parameter values (see Operating instructions, Operator Software BS7200)

Mux (D)	Mux (H)	Data type	Access	Short description	see manual
0	0	—	—	Reserve	
1	1	—	w/a	Stop command	CAN
2	2	—	w	Controller enable ON	CAN
3	3	—	w/a	Controller enable OFF	CAN
4	4	—	w/a	Quick stop ON	CAN
5	5	—	w	Quick stop OFF	CAN
6	6	—	w	Release brake (i.p.)	CAN
7	7	—	w/a	Activate brake (i.p.)	CAN
8	8	—	w	Pause ON	CAN
9	9	—	w	Pause ON	CAN
10	A	—	w/a	Acknowledge contouring error / response monitoring	CAN
11	B	32-bit int.	w	Jog mode	CAN
12	C	8-bit char	w	Start a motion task	CAN
13	D	32-bit int.	w	Go to reference	CAN
14	E	—	w	Set reference point	CAN
15	F	32-bit int.	r/w/a	Read / write the position	CAN
16	10	32-bit int.	r/w/a	Read / write the speed	CAN
17	11	32-bit int.	r/w/a	Read / write acceleration and braking ramps	CAN
18	12	8-bit char	w/a	Store the local motion task	CAN
19	13	8-bit char	w/a	Load a motion task from the motion block store	CAN
20	14	8-bit char	r/w	Type of motion task: absolute / relative	CAN
21	15	—	w/a	Store controller parameters in the EEPROM	CAN
22	16	—	w/a	Activate the baud-rate search	CAN
23	17	—	—	reserved	
24	18	8-bit char	w/a	Activate station for broadcast objects	CAN
25	19	—	w/a	Deactivate station for broadcast objects	CAN
26	1A	—	—	reserved	
27	1B	—	—	reserved	
28	1C	—	—	reserved	
29	1D	—	—	reserved	
30	1E	—	—	reserved	
31	1F	32-bit int.	r/a	Actual position	BS7200
32	20	32-bit int.	r/a	Actual speed (velocity)	BS7200
33	21	32-bit int.	r/a	Actual contouring error	
34	22	32-bit float	r/a	Actual current	BS7200
35	23	32-bit float	r/a	Actual speed (revs.)	BS7200
36	24	16-bit int.	r/a	Actual angle	BS7200
37	25	16-bit int.	r/a	Actual heat sink temperature	BS7200
38	26	16-bit int.	r/a	Actual internal temperature	BS7200
39	27	16-bit int.	r/a	Actual DC-link voltage	BS7200
40	28	16-bit int.	r/a	Actual ballast power	BS7200
41	29	16-bit int.	r/a	Actual I ² t loading	BS7200
42	2A	32-bit float	r/a	Actual running time	BS7200
43	2B	32-bit int.	r/a	Actual digifas™ serial no.	
44	2C	4 char	r/a	Actual digifas™ software version	
45	2D	4 char	r/a	Actual CAN interface software version	
46	2E	4 char	r/a	Actual digifas™ ID	CAN
47	2F	32-bit int.	r/a	Actual interface no.	
*48	30	8-bit char	r/w/a/d	Setpoint entry mode	CAN
49	31	16-bit int.	r/w	Digital setpoint	CAN
50	32	—	—	Reserve	
*51	33	32-bit float	r/w	I _{rms} effective current	BS7200
52	34	32-bit float	r/w	I _{peak} peak current	BS7200
53	35	8-bit char	r/w	I ² t signal	BS7200
54	36	—	—	reserved	
*55	37	8-Bit Char	r/w	Function of the I/O terminal	BS7200
*56	38	8-bit char	r/w	Select with/without brake	BS7200
57	39	32-bit float	r/w	K _p gain factor for current controller	BS7200
58	3A	32-bit float	r/w	T _n integral action time for current controller	BS7200
59	3B	—	—	reserved	
60	3C	16-bit int.	r/w	K _E voltage constant of the motor	CAN
61	3D	—	—	Reserve	
62	3E	—	—	reserved	
63	3F	32-bit float	r/w	L motor inductance	CAN
64	40	16-bit int.	r/w	K _p gain factor for speed controller	BS7200

Mux (D)	Mux (H)	Data type	Access	Short description	see manual
65	41	32-bit float	r/wr/w	Tn integral action time for speed controller	BS7200
66	42	—	—	Reserve	
67	43	32-bit float	r/w	PID-T2 2nd. time constant, speed controller	BS7200
68	44	16-bit int.	r/w	Final limit rpm for tacho feedback	BS7200
69	45	16-bit int.	r/w	Starting value Phi	BS7200
70	46	16-bit int.	r/w	Final value Phi	BS7200
71	47	32-bit float	r/w	Tacho time constant	BS7200
72	48	32-bit float	r/w	Kp gain factor for position controller	CAN
73	49	32-bit float	r/w	Ff feed-forward factor for position controller	CAN
74	4A	16-bit int.	r/w	t-emerg minimum braking time	CAN
75	4B	16-bit int.	r/w	t-accel-min minimum acceleration time	CAN
76	4C	32-bit float	r/w	v-max maximum speed (velocity)	CAN
77	4D	32-bit float	r/w	Contouring error window	CAN
78	4E	32-bit float	r/w	In Position window	CAN
79	4F	32-bit float	r/w/d	Resolution (increments converted to SI units, only when amplifier is disabled)	CAN
*80	50	8-bit char	r/w	Limit switch on/off/stop	BS7200
*81	51	8-bit char	r/w	DC monitor	BS7200
82	52	32-bit float	r/w/d	Resolution (SI units converted to increments, only when amplifier is disabled)	CAN
83	53	—	—	reserved	
84	54	—	—	Reserve	
85	55	—	—	Reserve	
*86	56	8-bit char	r/w	No. of motor poles (only when amplifier is disabled)	BS7200
87	57	—	—	Reserve	
88	58	—	—	Reserve	
89	59	—	—	reserved	
90	5A	—	—	reserved	
*91	5B	8-bit char	r/w	No. of resolver poles (fixed to 2)	BS7200
*92	5C	8-bit char	r/w	Language selection	BS7200
93	5D	32-bit float	r/w	Setpoint offset	BS7200
94	5E	16-bit int.	r/w	Setpoint ramp rising	BS7200
95	5F	16-bit int.	r/w	Setpoint ramp falling	BS7200
96	60	16-bit int.	r/w	Ballast power	BS7200
*97	61	8-bit char	r/w	Count direction for position controller	CAN
*98	62	8-bit char	r/w	Select ballast internal/external	BS7200
99	63	32-bit float	r/w	Limit switch 1	CAN
100	64	32-bit float	r/w	Limit switch 2	CAN
*101	65	8-bit char	r/w	Axis type linear/rotary	CAN
*102	66	8-bit char	r	Control from	CAN
103	67	16-bit int.	r/w	Response monitoring	CAN
104	68	32-bit float	r/w	Zero point offset	CAN
*105	69	8-bit char	r/w	Baud rate CAN BUS	CAN
*106	6A	8-bit char	r/w	Ramp type	CAN
*107	6B	8-bit char	r/w	Type of reference traverse	CAN
108	6C	32-bit float	r/w	Reference offset	CAN
109	6D	—	—	Reserve	
110	6E	—	—	Reserve	
111	6F	8-bit char	r/w	Index for error statistics	CAN
112	70	32-bit long	—	Read error statistics	CAN
113	71	16-bit int.	—	Teach-In function	CAN
114	72	16-bit int.	—	Ipeak2 in %	BS7200
115	73	—	—	Reserve	
116	74	—	—	Reserve	
117	75	—	—	Reserve	
118	76	—	—	Read status register	CAN
119	77	—	—	Read fault register	CAN
120	78	32-bit int.	r/w/a	Read / set status mask	CAN
121	79	32-bit int.	w	Start a direct motion task (absolute)	CAN
122	7A	32-bit int.	w	Start a direct motion task (relative)	CAN
123	7B	—	—	Reserve	
124	7C	—	—	reserved	
125	7D	—	—	reserved	
*126	7E	16-bit int.	r/w/d	Weighting factor for position	CAN
*127	7F	16-bit int.	r/w/d	Weighting factor for speed	CAN

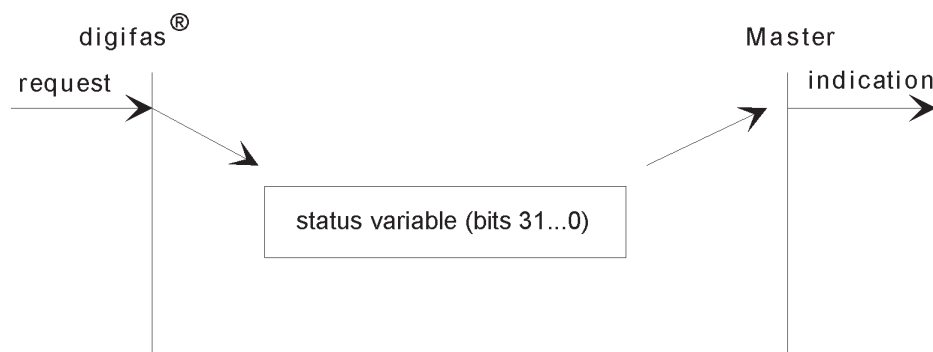
III.2.3 The Status object

The status variable contains the current state of the status register of the digifas™ 7100/7200. Every change of the contents of the status register is automatically transmitted (“notify event protocol”) if the status mask (Mux 120) is appropriately configured (see Chapter IV.1.8). After switching on the amplifier the status register mask has the state 0_H (mask = 00 00 00 00_H). This means that spontaneous messages are not possible. Additionally, the current contents of the status register (INTEGER32) can be requested by RTR at any time (“read event protocol”), or read out with the aid of the Control object, using Mux118.

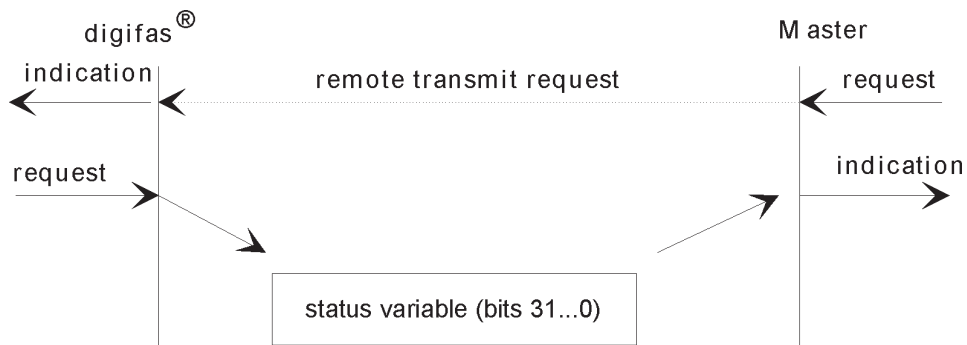
Classification:

Object type: uncontrolled event (notify event protocol), stored event (read event protocol)
 Access type: read only
 Data type: INTEGER32
 COB: ID2=0, ID1=1, ID0=1

Description of “notify event protocol”:



Description of “read event protocol”:



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

digifas™ 71xx (Bit 28=1)			digifas™ 72yy (Bit 28=0)		
Bit 26	25	24	Bit 26	25	24
Rated current / A			Rated current / A		
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	1	1	1

Description of the status register (SR) of the 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	rpm = 0
	0	rpm ≠ 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, 1 st 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 at reference point
	0	Position not at reference point
12	1	Reference point set
	0	Reference point not set
13	1	Speed or acceleration limit active
	0	Speed or acceleration limit not active
14	1	Ballast power exceeded
	0	Ballast power not exceeded
15	1	Digital setpoint entry
	0	Analog setpoint entry
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 ON (via BUS)
	0	Amplifier enable OFF
24-26	1	Output stage ID
	0	Output stage ID
27	1	With LCA
	0	No 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)

III.2.4 The Broadcast object

The broadcast functions make it possible to transfer data, such as motion blocks, simultaneously via the broadcast object to the stations of a group, and then to start them synchronously by using a broadcast start command.

The control command (BROADCAST OBJECT ON/OFF) can be used to switch the broadcast functions on and off for the individual stations.

When the broadcast functions are activated each station receives a broadcast group number. Separate groups of servo amplifier can thus be established.

You have to define the COB-ID for the broadcast object (bits: ID10-ID3) – see also the function BROADCAST ON/OFF). The COB-ID is independent of the station addresses which have already been used.

The broadcast object uses a modified "Read/Write Variable Protocol", which is not described in the CMS standard. If there are n members in a group, then n responses will be generated for a single transferred command. As far as CMS is concerned, this is just a modified type of protocol.

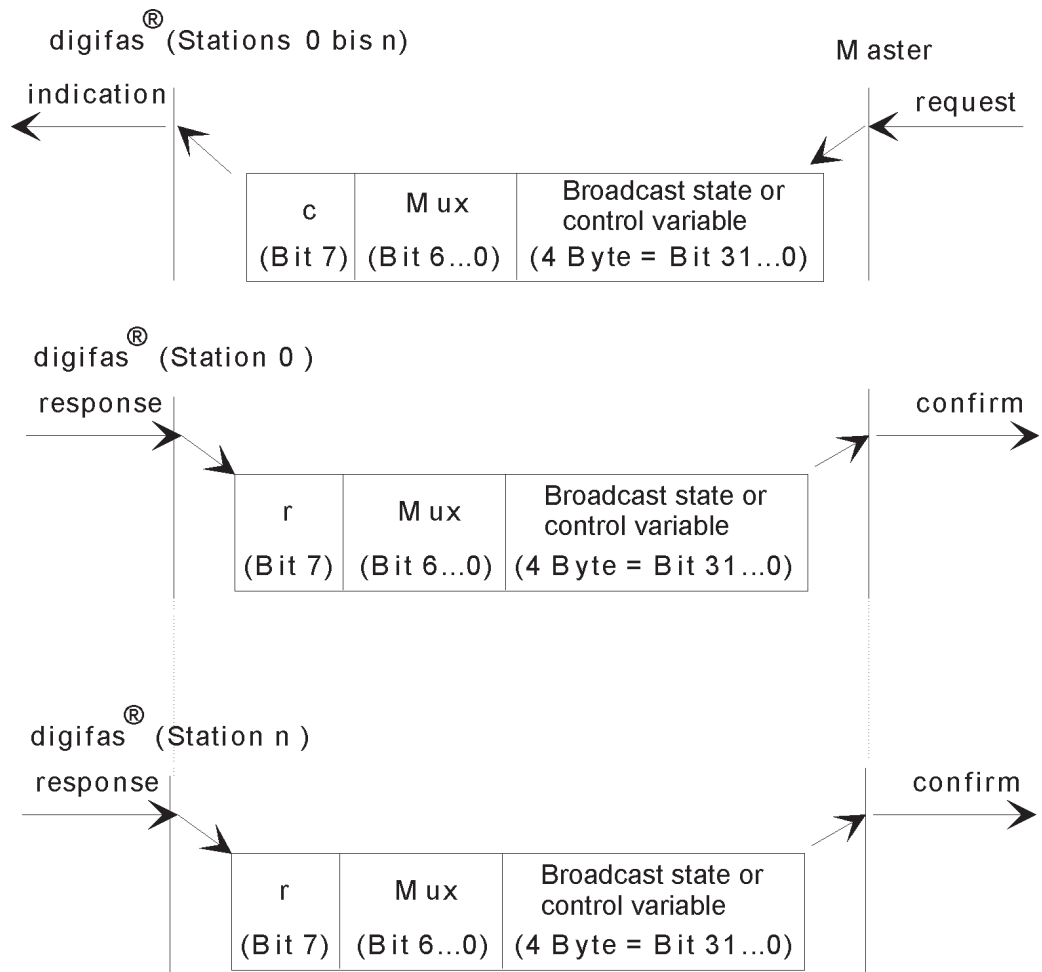


Warning !

In complex applications there is a possibility that dangerous situations may occur. Depending on the circumstances, these might not be recognized by the servo-amplifier. This is the responsibility of the system designer.

Classification:

Object type:	multiplex variable
Access type:	read / write
Data:	4 bytes
Multiplexer:	0 ... 127
COB:	ID2=0, ID1=0, ID0=1 for transmit direction (slave ⇒ master) ID2=1, ID1=1, ID0=1 for receive direction (master ⇒ slave)

Description “read/write variable protocol (broadcast)”:


Result bit : r=1 (fault), r=0 (no fault)

Write-response:

result bit (r) = 0 (no fault)

multiplexer = n ($0 \leq n \leq 127$)

control variable = 4 bytes

Fault event:

result bit (r) = 1 (fault)

multiplexer = 128 + n (128 is caused by the result bit r = 1)

control variable = fault number (1 byte, see Chapter III.2.1)

III.2.5 The Direct motion-task object

The direct motion-task object (CMS ⇒ Read/Write Access, Basic Variable) uses 8 bytes of data of the CAN data segment (see Chapter III.1.1) :

- 4 bytes of data : speed
- 2 bytes of data : acceleration ramp
- 2 bytes of data : braking ramp

Together, the definition and start of the direct motion-task object take up a maximum of 4ms. The previously defined direct motion-task is started by means of the control object

- Mux 121 : start direct motion task / absolute incl. weighted position setpoint
- or Mux 122 : start direct motion task / relative incl. weighted position setpoint

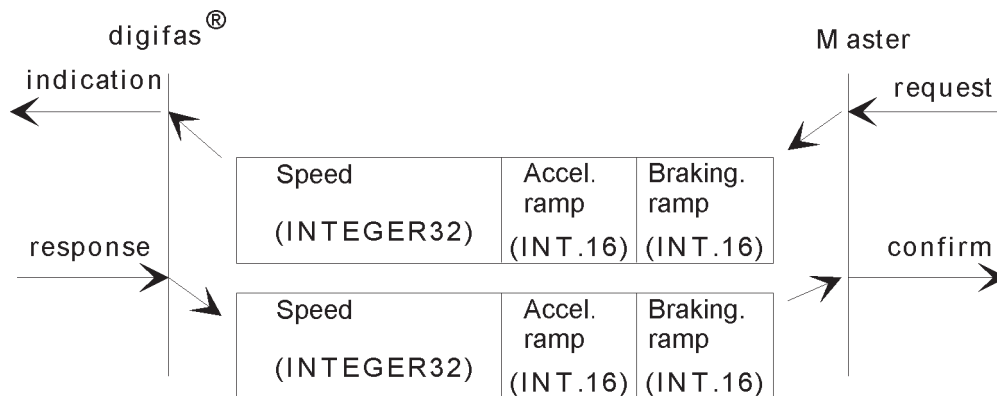
The motion task object can also be used to define motion tasks for the motion-task store of the digifas™ . However, use the control object with Mux 15 (read/write position) and Mux 20 (type of motion task absolute/relative) for the definition of the position setpoint and the type of motion task.

For reasons of compatibility, there are also multiplexers (control and broadcast object) available for the transfer of the direct motion task, as well as the direct motion-task object. Together, the definition and start of the direct motion task via the control / broadcast object takes up a maximum of 10ms.

Classification:

- Object type: basic variable
- Access type: write
- Data type: INTEGER32 (speed)+INTEGER32 (acceleration and braking ramp)
- COB: ID2=1, ID1=0, ID0=1 for transmit direction (slave ⇒ master)
- ID2=1, ID1=1, ID0=0 for receive direction (master ⇒ slave)

Description “Read/Write variable protocol (basic variable)”:



IV Description of the functions

IV.1 Control functions

The control functions are triggered by the transfer of control commands. These control objects (commands) do not contain any data (control variables). The multiplexers which are used for control functions are assigned the identifiers 1 ... 10.

IV.1.1 STOP function

When this command is received the drive is stopped. The current position is held, since the position control is still active. If a motion task was active at the time, bit 20 of the status register will be set to 0.

Definition:	Access type	:	write-request
	Multiplexer	:	1
	Object type	:	control and broadcast object
	Variable	:	no significance
	Data type	:	no significance

IV.1.2 CONTROLLER ENABLE ON/OFF

This command is used to set or remove the controller enable. The controller enable signal is AND-linked with the Enable input of the digifas™. The controller (output stage) is only enabled when both signals are set.

Definition:	Access type	:	write-request
	Multiplexer	:	2 CONTROLLER ENABLE ON 3 CONTROLLER ENABLE OFF
	Object type	:	control and broadcast object
	Variable	:	no significance
	Data type	:	no significance

IV.1.3 QUICK STOP ON/OFF

When the RAPID STOP ON command is received, the drive brakes with maximum deceleration down to $n = 0$ (standstill). In order to be able to perform motion functions the command QUICK STOP OFF must be transmitted. A motion function which was active before the QUICK STOP ON command is deactivated after this command.

Definition:	Access type	:	write-request
	Multiplexer	:	4 QUICK STOP ON 5 QUICK STOP OFF
	Object type	:	control and broadcast object
	Variable	:	no significance
	Data type	:	no significance

IV.1.4 RELEASE/APPLY BRAKE function (in preparation)

If a brake is available (optional) it can be controlled by means of the BRAKE RELEASE/APPLY command.

Definition:	Access type	:	write-request
	Multiplexer	:	6 RELEASE BRAKE
		:	7 APPLY BRAKE
	Object type	:	control and broadcast object
	Variable	:	no significance
	Data type	:	no significance

IV.1.5 PAUSE ON/OFF

This command can be used to interrupt a motion task which is in progress for a short time and then to carry on (important for relative positioning tasks).

Definition:	Access type	:	write-request
	Multiplexer	:	8 PAUSE ON
		:	9 PAUSE OFF
	Object type	:	control and broadcast object
	Variable	:	no significance
	Data type	:	no significance

IV.1.6 ACKNOWLEDGE CONTOURING ERROR/RESPONSE MONITORING

This command can be used to acknowledge the contouring error (SR bit 8) and/or the response monitoring (FR bit 3).

Definition:	Access type	:	write-request
	Multiplexer	:	10
	Object type	:	control and broadcast object
	Variable	:	no significance
	Data type	:	no significance

IV.1.7 BROADCAST ON/OFF function

The BROADCAST ON command can be used to switch stations over to broadcast functionality. To achieve this you must define the ID10 ... ID3 and transfer the variables. Stations with the same COB-ID for broadcast objects form a group. It is thus possible to form several broadcast groups. The BROADCAST ON command must only be transferred as a control object, a transmission as a broadcast object, on the other hand, would not define broadcast functionality for the stations concerned. The BROADCAST OFF command can be transferred as a control or a broadcast object, so that on the one hand it is possible to deactivate individual members of a broadcast group, on the other hand a complete group can be deactivated. All commands and functions can be used as broadcast commands.

Definition:	Access type	:	write-request
	Multiplexer	:	24 BROADCAST ON
		:	25 BROADCAST OFF
	Object type	:	control and broadcast object
	Variable	:	Identifier ID10 ... ID3 (Mux 24) no significance (Mux 25)
	Data type	:	8-bit char (Mux24)

IV.1.8 Setting the status register mask

The slave transmits the contents of the status register automatically, if a change in status occurs (i.e. the change of a bit in the status register). If it is *not* required that every bit change results in transmitting the status register contents, then a status register mask can be applied. This means that the bits where a change should *not* result in a transmission of the contents of the status register can be masked.

When the controller is switched on the status register has the value 0Hex (Mask = 00 00 00 00Hex). This means that spontaneous messages are not possible.

Operation of the mask for the automatic transmission of the status register

Example:

Contents of the status register		Mask	Automatic transmission
old	new		
FF FF FF FF	FF FF FF FE	FF FF FF FF	yes
FF FF FF FE	FF FF FF FF	FF FF FF FE	no
EE EE EE EE	EE EE EE EE	FF FF FF FF	no

Definition:

Access type	:	write
Multiplexer	:	120
Object type	:	control and broadcast object
Variable	:	status register mask
Data type	:	32-bit integer

IV.1.9 Activate baud rate recognition

To change the transfer speed of the CAN network, the baud rate recognition of the slave is activated. The slave then has the capability of recognising baud rates itself, and of adjusting itself to match.

The possible baud rates are:

20 kbit/s, 50 kbit/s, 100 kbit/s, 125 kbit/s, 250 kbit/s, 500 kbit/s, 1000 kbit/s

The search for the new baud rate which is to be set is active after about 100 ms, so that the baud rate recognition/search by other stations can be activated first.

The command ACTIVATE BAUD RATE RECOGNITION is defined as a control and broadcast object. Groups can be formed, with the aid of the broadcast object, thus making it possible to initiate a baud rate search by several stations at the same time. This group must include all the bus participants, apart from the master. Make sure that all the bus participants have the same baud rate after the baud rate change, otherwise a communication fault will be produced.

Definition:

Access type	:	write
Multiplexer	:	22
Object type	:	control and broadcast object
Variable	:	no significance
Data type	:	no significance

IV.1.10 ANALOG / DIGITAL setpoint entry

The “ANALOG/DIGITAL setpoint entry” function can be used to switch the controller between 5 modes. (Modes 1 and 2 are not possible with a standard amplifier)

Mode	Description	Remarks
0	position control	transfer and start of motion tasks
1 (reserved)	analog speed control	not possible for standard CAN-instrument
2 (reserved)	analog current control	not possible for standard CAN-instrument
3	digital speed control	parameter entered form the CAN-bus
4	digital current control	parameter entered from the CAN-bus

Motion tasks can **not** be started in modes 1 to 4.

Definition:

Access type	:	read/write
Multiplexer	:	48
Object type	:	control and broadcast object
Variable	:	mode (0, 1, 2, 3, 4)
Data type	:	8-bit char



Never change the operating mode while the motor is running !

If the amplifier is enabled, a change of mode is only permitted while the the speed is 0. Set the speed to 0 before changing over.

IV.2 Digital speed and torque control

When using the digital speed and torque (current) control, the setpoint for the speed or current control is provided via the bus.

In this case it is necessary to switch the controller over to speed or torque /current) control by means of the function “ANALOG/DIGITAL setpoint entry” (see Chapter IV.1.10).



The digifas™ position controller is switched off when speed or torque control is active.

IV.2.1 DIGITAL setpoint for the speed controller

If mode 3 is selected by the function “ANALOG/DIGITAL setpoint entry”, then the speed setpoint is provided digitally (Mux. 49).

max. negative speed setpoint (rated r.p.m.) : 9999H
 max. positive speed setpoint (rated r.p.m.) : 6666H

The command is carried out immediately. If the command is repeated with various speed setpoints, there must be **no** STOP-commands in between.

Definition:	Access type	:	read/write
	Multiplexer	:	49
	Object type	:	control and broadcast object
	Variable	:	speed setpoint
	Data type	:	16-bit integer

IV.2.2 DIGITAL setpoint for the torque controller

If mode 4 is selected by the function “ANALOG/DIGITAL setpoint entry”, then the current setpoint is provided digitally (Mux. 49).

max. negative current setpoint (rated current) : 9999H
 max. positive current setpoint (rated current) : 6666H

The command is carried out immediately. If the command is repeated with various current setpoints, there must be **no** STOP-commands in between.

Definition:	Access type	:	read/write
	Multiplexer	:	49
	Object type	:	control and broadcast object
	Variable	:	current setpoint
	Data type	:	16-bit integer

IV.3 Description of the position and motion functions

The position and motion functions (position control) are initiated by the transfer of control objects (control commands).

There must not be any other position or motion function active (jog mode , go to reference, motion task) when starting a position or motion function.

If the next motion task is to be started before the present function has reached its target position, then a Stop command must be given first.

IV.3.1 Jog mode

The command JOGGING ON can be used to move the drive in jog mode. The signed speed setpoint is transferred as a part of the command.

Definition:

Access type	:	write-request
Multiplexer	:	11
Object type	:	control and broadcast object
Variable	:	weighted set speed
Data type	:	32-bit integer

Conditions:

Controller enabled	(SR bit01	= 1)
External control	(SR bit09	= 1)
No Quick Stop	(SR bit16	= 0)
Jog mode not active	(SR bit19	= 0)
No motion task in progress	(SR bit20	= 0)
No "go to reference" in progress	(SR bit21	= 0)

IV.3.2 Set reference point

The command SET REFERENCE POINT defines the present position as the reference point. Bit 12 in the status register is set to 1 and the position functions are enabled.

Definition:

Access type	:	write-request
Multiplexer	:	14
Object type	:	control and broadcast variable
Variable	:	no significance
Data type	:	no significance

Conditions:

Controller enabled	(SR Bit01	=1)
Speed = 0	(SR Bit02	=1)
External control	(SR Bit09	=1)



Take care, when using this function, that the position of the reference point permits the subsequent positioning movements. The software limit switches which are parameterised in the digifas™ may be ineffective. The axis could then drive up to the hardware limit switch or the mechanical limit. There is a danger of damage.

IV.3.3 Go to reference (reference traverse)

The reference traverse (go to reference, homing) is initiated by the START REFERENCE TRAVERSE command.

The reference switch is connected to the input I/O (terminal X3/15) of the digifas[®].

The drive moves in the direction which is set in the digifas[®] as far as the target point. The target point and the behaviour of the drive during the reference traverse are determined by the type of reference traverse (see Chapter V.3.19, Mux 107). When the target is reached, the reference point is set and the drive is stopped. The speed for the reference traverse is taken from the "Parameter value". The mathematical sign of the speed setpoint is not evaluated.

The value set for the reference traverse speed must not be more than 10 % of the preset maximum speed v_max (can be set in the BS 7200 operator software, on the CONNECT page, or by using Mux = 76).

If the 10 % threshold is exceeded, the START REFERENCE TRAVERSE command will be answered by the error message "Wrong parameter value" (error number 4) (see Chapter IV.7).

Definition:	Access type	:	write-request
	Multiplexer	:	13
	Object type	:	control object
	Variable	:	weighted speed setpoint
	Data type	:	32-bit integer
Conditions:	Controller enabled		(SR bit01 = 1)
	External control		(SR bit09 = 1)
	No Quick Stop		(SR bit16 = 0)
	Jog mode not active		(SR bit19 = 0)
	No motion task in progress		(SR bit20 = 0)
	No "go to reference" in progress		(SR bit21 = 0)

IV.3.4 Start a motion task

For the command START MOTION TASK, all the data required for the positioning procedure are present in the motion task store of the digifas[™] (Chapter IV.5). The motion task number which is transferred specifies the motion store in the digifas[™]. With motion task number 0 a task is started from the local storage of the communication processor (direct task), for task numbers 1 ... 120 the motion tasks are selected from the motion task store of the digifas[™].

Definition:	Access type	:	write-request
	Multiplexer	:	12
	Object type	:	control and broadcast object
	Variable	:	motion task number (0 ... 120)
	Data type	:	8-bit char
Conditions:	Controller enabled		(SR bit01 = 1)
	External control		(SR bit09 = 1)
	Reference point set		(SR bit12 = 1)
	No Quick Stop		(SR bit16 = 0)
	Jog mode not active		(SR bit19 = 0)
	No motion task in progress		(SR bit20 = 0)
No "go to reference" in progress		(SR bit21 = 0)	

When the positioning task is finished and the set position has been reached, SR bit10 is set to 1 (In Position). The "In Position" signal remains available as long as the axis is within the "In Position" window or until a new motion task is started. If the programmed values for v_max or a_max (limit speed or limit acceleration) are exceeded during the performance of the motion task, then they are limited to these values. The limitation is marked in SR bit 13.

IV.3.5 Start a direct-motion task (ABSOLUTE / RELATIVE)

For the command START DIRECT-MOTION TASK (ABSOLUTE/RELATIVE) all the data for acceleration and positioning which are required for the positioning procedure are present in the local storage of the digifas™. This command should be used in connection with the direct motion task object, in order to achieve a more efficient transmission time and a time-optimal conversion to the CAN interface (definition + start of the direct motion task $\leq 4\text{ms}$).

Definition:	Access type	:	write-request
	Multiplexer	:	ABSOLUTE 121
		:	RELATIVE 122
	Object type	:	control and broadcast object
Variable	:	weighted position value	
	Data type	:	32-bit integer
Conditions:	Controller enabled		(SR bit01 = 1)
	External control		(SR bit09 = 1)
	Reference point set		(SR bit12 = 1)
	No Quick Stop		(SR bit16 = 0)
	Jog mode not active		(SR bit19 = 0)
	No motion task in progress		(SR bit20 = 0)
	No reference traverse in progress		(SR bit21 = 0)

When the positioning task is finished and the set position has been reached, SR bit10 is set to 1 (In Position).

The "In Position" signal remains available as long as the axis is within the "In Position" window or until a new motion task is started or jog mode or a reference traverse is started. If the programmed values for v_{max} or a_{max} are exceeded during the performance of the motion task, then they are limited to these values. The limitation is marked in SR bit 13.

IV.4 Actual value functions

The actual value functions are used to read the current values in the server (digifas™). The multiplexers used for actual values are assigned to the IDs 31 ... 47.

Definition:	Access values	:	read (write-request)
	Multiplexer	:	31-47
	Object art	:	control and broadcast object
	Variable	:	contents and data type: see table
	Data type	:	32/16 - bit integer, 32-bit float, 4 ASCII char.

Summary of the actual value multiplexers (for data types see Chapter III.2.2.1) :

Function (reading)	Multiplexer	Function (reading)	Multiplexer
Actual position	31	Ballast power	40
Actual speed	32	I ² t-utilisation	41
Actual contouring error	33	Operating time	42
Actual current	34	Serial number: digifas™ Hardware	43
Speed	35	Software-version: digifas™	44
Angle	36	Software-version: CAN-Interface	45
Heat sink temperature	37	digifas™ ID (DFAS)	46
Internal temperature	38	Interface number	47
DC-link voltage	39		

The actual values for position, speed and contouring error are available on demand as an INTEGER32-value of signed 32 bit numbers. The actual value (SI-unit) is given by the transferred INTEGER32-value and the weighting factor, and it can be described by the following equation:

$$Actual\ value = \frac{Integer32\ value}{10^{Weighting\ factor}}$$

The position weighting factor is also used for the calculation of the contouring error. The weighting factors can be programmed by the function SET CONTROLLER PARAMETERS. The transfer of weighted actual values is necessary, because some master stations do not permit floating point processing.

Examples:

Actual value of position	= 167456	Actual value of speed	= 12345
Position weighting factor	= 2	Speed weighting factor	= 3
⇒ Position (SI)	= 1674.56 mm	⇒ speed (SI)	= 12.345 mm/s

IV.5 Data transfer functions

The memory of the digifas™ is divided into the motion-block memory and the controller parameter memory.

Controller parameters are stored in a non-volatile memory by using the command SAVE CONTROLLER PARAMETERS IN EEPROM, and remain stored when the digifas™ is switched off. The motion-block memory consists of a local motion-block memory (volatile), 16 RAM motion-block memory positions (129 ... 144 , volatile) and 120 EEPROM-motion-block memory positions 1 ... 120 , non-volatile).

All the data transfer functions refer to the local motion block. The commands SAVE MOTION TASK or LOAD MOTION TASK can be used to transfer a motion task between the local buffer and the RAM or the EEPROM.

IV.5.1 Read/write a parameter of the local motion task

The command READ/WRITE MOTION PARAMETER can be used to read or change a parameter of the local motion task.

IV.5.1.1 Read/write the position value

Definition:	Access type	:	read/write
	Multiplexer	:	15
	Object type	:	control and broadcast object
	Variable	:	weighted position value
	Data type	:	32-bit integer

IV.5.1.2 Read/write the speed value

Definition:	Access type	:	read/write
	Multiplexer	:	16
	Object type	:	control and broadcast object
	Variable	:	weighted speed value
	Data type	:	32-bit integer

IV.5.1.3 Read/write the acceleration/braking ramp

Definition:	Access type	:	read/write
	Multiplexer	:	17
	Object type	:	control and broadcast object
	Variable	:	acceleration and control ramp (bits 31..16 acceleration, bits 15..0 braking ramp)
	Data type	:	32-bit integer

IV.5.1.4 Read/write the type of motion task absolute/relative

This command is used to determine the type of motion task. The task type is a bit-variable. The significance of the individual bits can be seen in the table below.

Definition:	Access type	:	read/write
	Multiplexer	:	20
	Object type	:	control and broadcast object
	Variable	:	type of motion task
	Data type	:	8-bit char

Coding of the task type

Bit		Task type	Description
0	3 4		
0	x x	absolute	Move to an absolute target position referred to the reference point
1	0 0	relative	Load is within the InPosition window - relative to last target position Load is not within the InPosition window - relative to actual position at start
1	1 x	relative 1	Relative to last target (in conjunction with motion-block change-over: e.g summing operation)
1	0 1	relative 2	Relative to actual position at Start (in conjunction with motion-block change-over: e.g print register mark control)



When using sequences of relative motion blocks, please note that each turn has 16 bit (0 ... 65535) internal resolution. If the path cannot be represented precisely with this resolution, then approximation errors can occur.

Coding for single task / following task

Bit		Task type	Description
1	2		
0	x	single motion task	Drive stops at the target position, InPosition message is output
1	0	following task with pause	The drive is braked to a stop at the target position, using the predetermined braking ramp, before the next task is started. No InPosition message during the pause.
1	1	following task without pause	The drive is not braked to the target position. The speed is adjusted to the speed required for the following task, using the preset acceleration ramp. This function can be used to move along speed profiles..



If the ramp type is set to sine², braking is always to the target position. The setting for a pause (bit 2) is then irrelevant.

Bits 8 ... 15 - Number of the following task (is only evaluated if bit 1 = 1)

IV.5.1.5 Save the local motion task to the EEPROM

The command SAVE MOTION TASK can be used to save the local motion task to the RAM or EEPROM. Enter the target motion-task number as an "Index" in the RAM/EEPROM.

120 motion tasks can be saved in the EEPROM and 16 in the RAM.

Definition:

Access type	:	write-request
Multiplexer	:	18
Object type	:	control and broadcast object
Variable	:	motion task number 1 ... 120 (EEPROM, non-volatile) motion task number 129...144 (RAM volatile)
Data type	:	8-bit char

IV.5.2 Load a motion task from the EEPROM to the local buffer

Use the command READ MOTION TASK to load a motion task from the RAM or EEPROM to the local buffer.

Definition:

Access type	:	write-request
Multiplexer	:	19
Object type	:	control and broadcast object
Variable	:	motion task number 1 ... 120 (EEPROM, non-volatile) motion task number 129...144 (RAM volatile)
Data type	:	8-bit char

IV.5.3 Read the BAUD RATE

This command can be used to read out the baud rate on the digifas™. (correspondence of the baud rates and the parameter values: see table in Chapter III.2.2.1).

Definition:

Access type	:	read
Multiplexer	:	105
Object type	:	control variable
Variable	:	baud rate (Auto, 20, 50, 100, 125, 250, 500, 1000 kbit/s)
Data type	:	8-bit char neue seite

IV.5.4 Read/write a controller parameter

This command can be used to write a controller parameter, which has been recognised through the multiplexer, into the volatile memory of the digifas™ – or to read it out from the memory.

The function SAVE CONTROLLER PARAMETERS TO EEPROM transfers the controller parameters to the non-volatile memory.

Definition:

Access type	:	read/write
Multiplexer	:	see Chapter III.2.2.1
Object type	:	control and broadcast object
Variable	:	see table in Chapter III.2.2.1
Data type	:	see table in Chapter III.2.2.1, below summary

IV.5.5 Save the controller parameters to the EEPROM

With this command you can transfer the control parameters, which are in the volatile memory of the digifas™, to the non-volatile memory (EEPROM).

Definition :

Access type :	write
Multiplexer :	21
Object type :	control variable
Variable :	no significance
Data type :	no significance

IV.5.6 Read/write the POSITION weighting factor

The POSITION weighting factor gives a weighting to all position data which are transferred as Long Data (32-bit Int.), but are really Float Data.

This is valid for the following data:

Actual position	(Mux 31)
Actual contouring error	(Mux 33)
Position value in motion task	(Mux 15)

The transfer of weighted data is necessary because some master stations do not permit floating-point processing. This command can only be executed while the output stage is disabled (for the assignment of weighting factors to the parameter values: see Chapter III.2.2.1 Table "Correspondence of identifiers and parameter values in digifas™").

Definition :

Access type :	read/write
Multiplexer :	126
Object type :	control variable
Variable :	position weighting factor (0,1,2,3)
Data type :	8-bit char

$$actual\ value = \frac{Integer32value}{10^{weighting\ factor}}$$

Example:

Position actual value	= 167456
Position weighting factor	= 2
⇒ Position (SI)	= 1674.56 mm

IV.5.7 Read/write the SPEED weighting factor

The SPEED weighting factor gives a weighting to all position data which are transferred as Long Data (32-bit integer), but are really Float Data.

This is valid for the following data:

Actual speed	(Mux 32)
Speed value in motion task	(Mux 16)
Jog mode	(Mux 11)
Reference traverse	(Mux 13)

This command can only be executed while the output stage is disabled.

Definition :

Access type :	read/write
Multiplexer :	127
Object type :	control variable
Variable :	speed weighting factor (0,1,2,3)
Data type :	8-bit char

$$actual\ value = \frac{Integer32value}{10^{weighting\ factor}}$$

Example:

Speed actual value	= 12345
Speed weighting factor	= 3
⇒ Speed (SI)	= 12.345 mm/s

IV.5.8 Teach In

The Teach In function can be used to define new motion tasks, based on motion tasks which are already available.

The parameter is transferred as the number of the initial motion task (1 ... 120, 129 ... 144) in the upper byte, and the number of the target motion task in the lower byte.

When the Teach In command is received, the interface program in the amplifier reads out the initial motion task from the motion-block memory, enters the current position as target position, designates the motion task as absolute, and stores it under the target task position.

IV.6 K_E , voltage constant of the motor

This parameter makes an optimum adaptation of the speed (rpm) dependent voltage pre-control of the current controller to the motor which is used.

The value is maximum the motor voltage constant \hat{K}_E [V/1000 min⁻¹] as given in the motor manual.

The motor-dependent default values, which we deliver with the operator software BS7200, are also used to adjust this parameter and should therefore not be changed.



When using the function “DIGITAL setpoint” for the torque controller (see Chapter IV.2.2) a value of K_E which is set too high as a result of incorrect adjustment will lead to instability and provokes a runaway of the motor at small setpoint values.

Effect:	Value too small	:	high motor speeds cause oversteering of the current controller. The final speed may not be achieved.
	Value too large	:	very low gain in the current controller or a low setting of the pulse current I_{peak} results in positive feedback. The motor can run away.

IV.7 L, motor inductance

This parameter makes an optimal adjustment of the speed-dependent voltage precontrol to match the inductance of the motor which is used. The value which is set for L is the same as the phase-phase value of the motor inductance L [mh] as given in the motor data sheet.

The parameter is adapted by the motor-dependent default values which we provide with the operator software BS7200. They should not be altered.

Effect:	Value too small	:	high motor speeds cause oversteering of the current controller. The available voltage is not fully utilized.
	Value too large	:	the motor current is unnecessarily increased and the thermal utilization of the motor is poor.

IV.8 Fault signals

If a write command is carried out without any errors, the command telegram which was received is mirrored back in response. In the case of a read command the requested data are transmitted.

If an error occurs in the execution of a command, the most significant bit in the first response byte ($r = 1$) will be set. In this case, a fault designation (fault number) is contained in the second response byte (the byte just after the multiplexer) .

The following faults can be signalled:

Fault number	Fault designation	Cause / description
1	Wrong command	An undefined command has been used
2	Wrong task number	Motion task number is <0 or >120
3	Wrong parameter number	An undefined parameter number has been used
4	Wrong parameter value	<ul style="list-style-type: none"> — a negative position entry was discovered while starting or saving an absolute motion task — the combination of selected Kp- and Tn-parameters for the current or speed controller cannot be represented internally
5	EEPROM access not possible	Motion tasks can only be saved to the motion-block memory of the digifas™ at fixed intervals (ca. 180ms)
6	Reference point not set	Motion task could not be started, because the reference point was not set after switching on the amplifier
7	No operator capability	Amplifier is not under bus control (use the operator software BS7200 to switch over the digifas™ to "control from bus")
8	Command cannot be executed because of operating state	A motion function (position mode) cannot be performed, because some type of motion function (jog mode, reference traverse, motion task) is active. Execute a Stop command first. The fault signal will also be generated if a motion function is performed, during which the command "digital setpoint" is transferred.
9	Command can only be performed in disabled state	The commands: Change weighting factors (Mux 126, 127), Change resolution (Mux 79, 82) and Change to position control mode (Mux 48) can only be performed while the output stage is disabled. Disable the output stage via the bus (Mux 3)
10	Parameter $<$ min value	Controller parameter is smaller than the permitted minimum value (see manual for Operator Software BS 7200, or Chapter V in this manual)
11	Parameter $>$ max value	Controller parameter is larger than the permitted minimum value (see manual for Operator Software BS 7200, or Chapter V in this manual)
12	BCC-error in motion task	Invalid or faulty task in the EEPROM (Checksum incorrect, EEPROM may be faulty)
13	Motion task not available	A motion task which should be started (start command with the appropriate motion task number), is not available in the motion-block memory.
14	Bus monitoring active	Missing master-digifas™ communication has triggered the response monitoring. Only certain commands are possible in this state (see Chapter III.2.2.1)
15	Not permitted in this mode	The command which was transferred cannot be performed in the present mode. (see Mux 48)
16	ANALOG mode not permitted	The available amplifier hardware does not permit ANALOG mode (see Mux 48).

IV.9 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 error can be found in the description of the BS7200 operator software.

IV.9.1 Error index

The components of the error statistics can be read out in association with the error index.

Access type : read
 Multiplexer : 111
 Variable : error index (0 ... 17)
 Data type : 8-bit char

IV.9.2 Error statistics

The error statistics depend directly on the error index. The component of the error statistics which is to be read must be identified with the aid of the error index.

The error statistics show different data formats, depending on the error index.

Access type : read
 Multiplexer : 112
 Variable : see table below
 Data type : see table below

The following table designates the individual components of the error statistics according to the error index. The column for error ID 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 ID.

Error index	Data format	Error ID	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 fault
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 Operator Software

V.1 General

The digital servo amplifiers of the digifas™ 7100 / 7200 series have to be adapted to the characteristics of your machine. Both types of amplifier are parameterised by using the BS7200 operator software.



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

Fieldbus monitor

In the menu page “Service” an item “FIELDBUS MONITOR” appears. This leads to a screen page where the data words in the transmit and receive direction can be displayed, as seen from the bus-master. This page is helpful when searching for faults and when commissioning the bus communication. A precise description of the monitor page can be found in Chapter V.2 .

CONNECT menu page

The menu item “Connect” appears in the menu line of the operator software. All the relevant parameters for setting the position controller can be found under this menu item. A description of all Connect parameters can be found in Chapter V.3 .

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

s_act	actual position of the load	(0 ... 99,999,999.999 mm)
s_fault	actual contouring error of the load	(0 ... 99.999 mm)
v_act	actual speed of the load	(0 ... 9,999.999 mm/s)

V.2 Fieldbus monitor

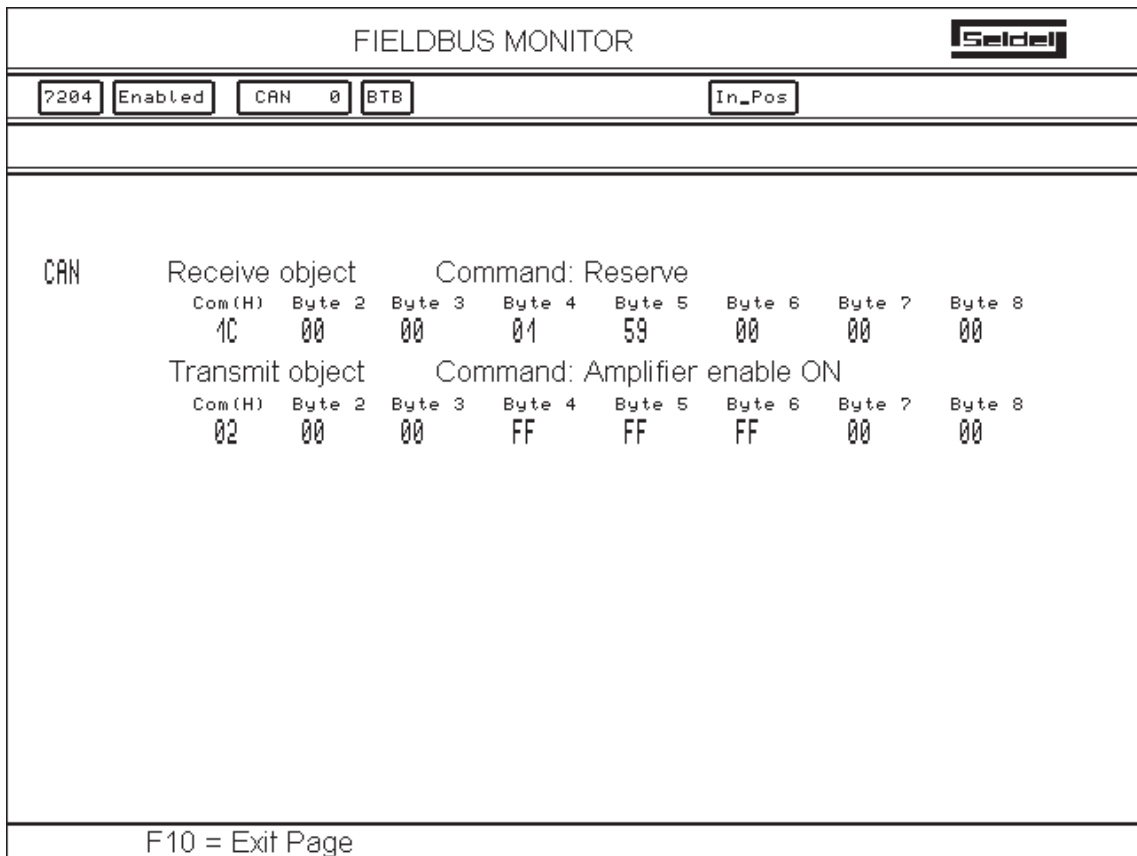


Fig: snapshot of the fieldbus monitor

- Row 1 : Header row
- Row 2 : Status line, as described in the BS7200 operating manual
- CAN rows :
 - Receive object : the last bus object received by the master
 - Transmit object : the last bus object transmitted by the master
 - Command : The meaning of a command is displayed in clear text
 - Com(H) : Command byte (multiplexer) hexadecimal
 - Byte2...8 : Data bytes

V.3 Parameter description CONNECT menu page

V.3.1 Kp, P-gain

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

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

V.3.2 Ff, feed-forward factor

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 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, the drive is lagging
Value too high — the dynamic range of the P-controller is limited, the drive is oversteering

V.3.3 t_accel_min, maximum acceleration

A drive is always so dimensioned that it can deliver more torque than the application requires. With this parameter you can set a limit value for the maximum mechanical acceleration which cannot be exceeded by the drive. The setting is effective as a limit for the acceleration and deceleration time.

Setting range: 10 ... 2550 ms

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

V.3.4 v_max, maximum speed

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 the 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.3.5 t_emerg, maximum deceleration

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 shorter than the minimum braking / acceleration time t_accel_min. Setting range: 10 ... 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.3.6 Resolution

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.

No-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 motor revolutions: $i = 10$ revs. , distance travelled for i motor revs.: $s = 50$ mm

$$\text{resolution} = s / i \quad \text{Resolution} = \frac{50 \text{ mm}}{10 \text{ turns}} = 5 \frac{\text{mm}}{\text{turn}}$$

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

$$ds = \frac{\text{Resolution}}{4096 \frac{\text{steps}}{\text{turn}}} = \frac{5 \text{ mm}}{4096 \frac{\text{steps}}{\text{turn}}} = 0,0012207 \frac{\text{mm}}{\text{step}}$$

Effects : Value too low — the desired physical values will not be reached
Wert zu hoch — the desired physical values will be exceeded

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



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 (CAN bus), as to whether they are within the permitted min/max limits. If not, adjust the parameters !

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")



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.

V.3.7 Count direction

Determines the count direction for positional values. Selection: positive / negative

Effects :

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

In both cases, the actual position counts upwards !

V.3.8 Contouring error

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.3.9 In Position

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

Effects :

Value too low	—	positioning time increases, no In Position signal
Value too high	—	axis approaches the target jerkily



After a traverse, although the drive stops at the target point, for reasons of control technology the motor axis may be 1/4096 of a turn away from the target point. This residual deviation will be taken into account when the next relative motion task is started, so that these deviations do not accumulate. The processing of the remaining path is exclusively determined by deviations from the position. Rounding-off errors (max. 0.5/4096 of a turn) cannot be compensated in the calculation of the target position. This means that traversing in incremental distances with relative tasks always leads to small but accumulating positional errors.

Depending on the accuracy required, you should therefore either completely avoid using incremental coordinates, or use an absolute motion task to return to the start position.

V.3.10 Zero point offset

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+

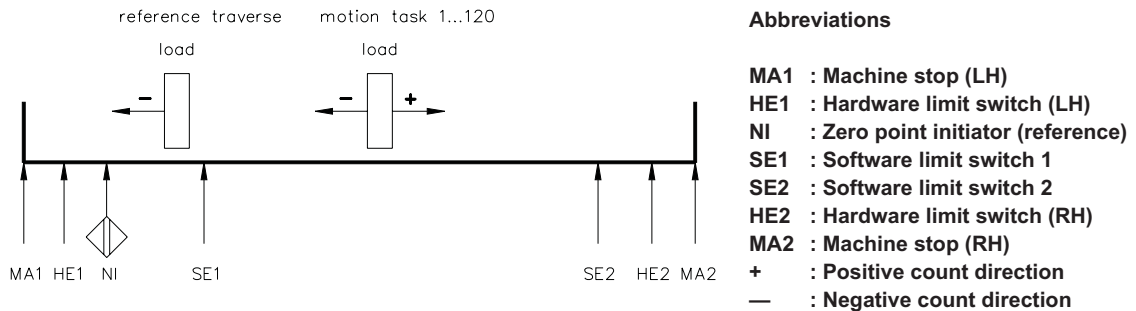
Setting range: 0 ... resolution

V.3.11 Limit switch 1

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 smaller than the preset value, and if this is so it blocks the direction of rotation (negative count direction).

Setting range: —20% resolution ... +maximum path entry
 (maximum path entry = 32767 * resolution < 999.999,99 mm)

Positioning principle for the software limit switch :



Effects : Value too low — mechanical stop is reached
 Value too high — if larger than Limit Switch 2, movement is impossible

V.3.12 Limit switch 2

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)

Positioning principle for the software limit switch: see Chapter V.3.11 above

Effects : Value too low — if smaller than Limit Switch 1, movement is impossible
 Value too high — mechanical stop is reached

V.3.13 Axis type

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 rotary axis, differences result in the treatment of the software limit switches. Selection: linear / rotary

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. In this case the software limit switches have no significance. 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. Moving in incremental dimensions leads to minimum accumulative system errors from rounding off. Consult our service department.

A rotary axis can be a: axle drive, rotary table, winder, conveyor belt (endless belt), roller drive

V.3.14 Controlled by

Determines which instrument is to be used for parameterisation.
Selection: BUS, PC

V.3.15 Response monitoring

Response monitoring is one of the safety functions of digifas™. If the digifas™ is not accessed from the bus within the preset response-monitoring time, then the response monitoring ensures that, if bus communication is missing, a previously started function is cancelled and the drive is brought to rest.

The time which is set depends on the bus loading, i.e.:

the lower the load on the bus, the shorter the response-monitoring time.

Setting range: 0 ... 5000 ms

Effects: Value too low — the digifas™ will not perform any functions
Value too high — the safety function is impaired

V.3.16 Baud rate

Determines the transmission rate of the CAN interface in the digifas™. The baud rate which is set is only activated when the servo amplifier is switched off and on again.

Selection: Auto, 20, 50 100, 125, 250, 500, 1000 kbaud

Effects : no communication if set incorrectly

V.3.17 Ramp type

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

Selection : trapeze / sine²

Trapeze

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

Sinus²

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.3.18 Reference offset

Reference offset is used to define the reference point to be a position other than 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 = $32767 * \text{resolution} < 999.999,99 \text{ mm}$.

 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 making a fresh reference traverse, check that the programmed software limit-switches and target positions 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.3.19 Type of reference traverse

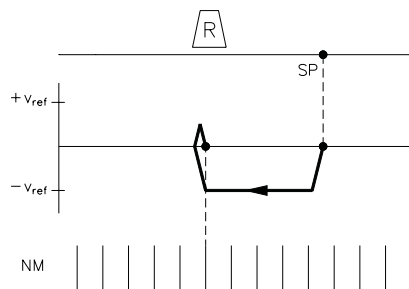
You can select which type of reference traverse should be performed.

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

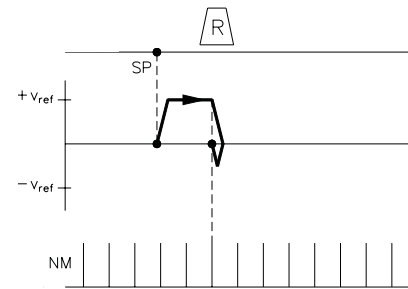
Reference traverse 1 (1-, 1+) Go to reference switch, with zero mark recognition

In this case, a reference traverse is also possible without hardware limit-switches. The pre-condition 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 (MUX 108) 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 switch transition. A two-pole resolver has exactly one zero-crossing per turn, so the positioning of the zero mark within a motor turn is unambiguous. If the switching transition 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 a 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 a reference switch without zero-mark recognition
The reference point is set to the switching transition of the reference switch.

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

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

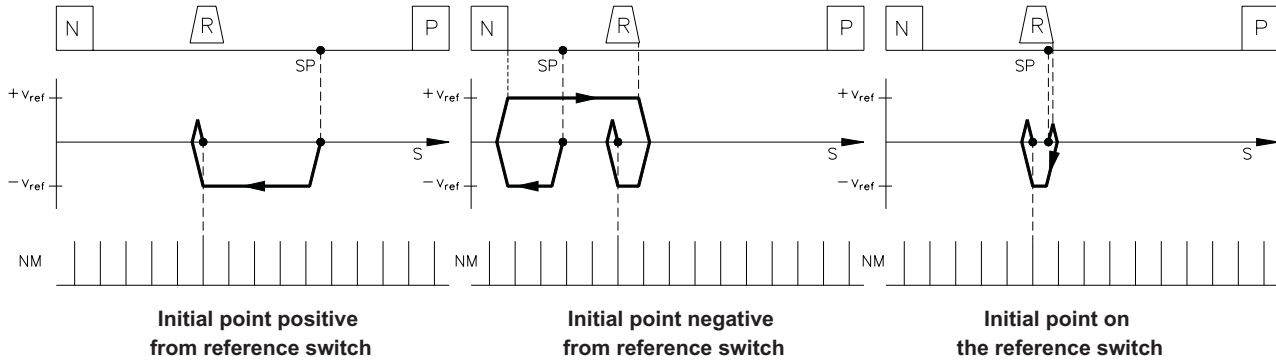
On the following pages you can find the paths traversed during the various reference traverses for every possible initial situation (positive count direction).

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

Reference traverse 1 negative

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



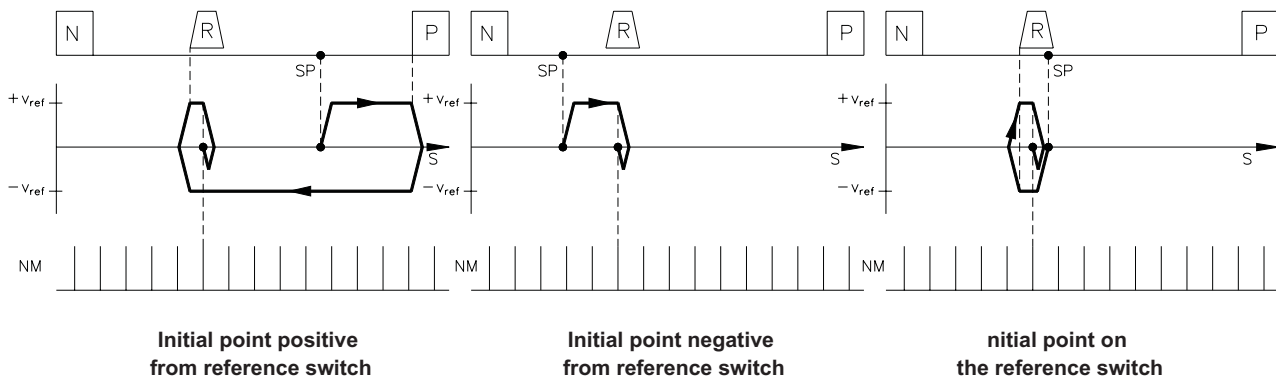
Attention !

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 (Mux 80).

Reference traverse1 positive

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



Attention !

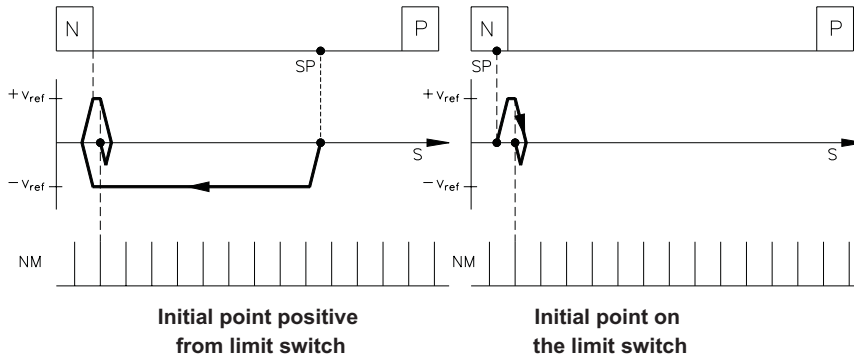
Before starting the reference traverse, check the safety of the system to ensure that the load can also be moved if the limit switches are disconnected or defective.

The STOP limit switch function must be activated to achieve the full reference traversing functionality (Mux 80).

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

Reference traverse 2 negative

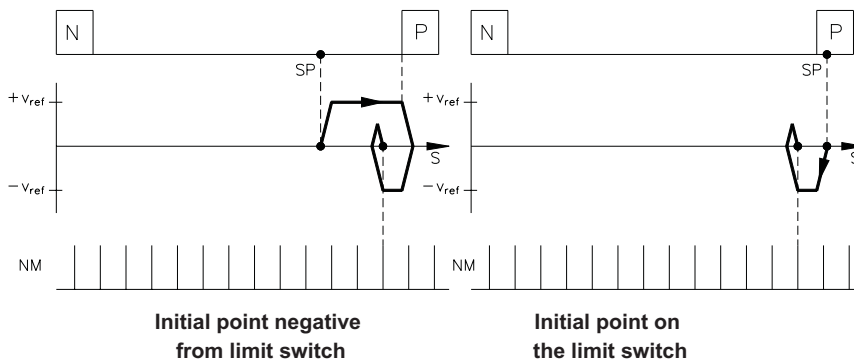
(without reference switch, traverse direction negative, 2 initial situations, positive count direction)



Attention !
Hardware limit switches must be present and connected.
The STOP limit switch function must be activated (Mux 80).

Reference traverse 2 positive

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



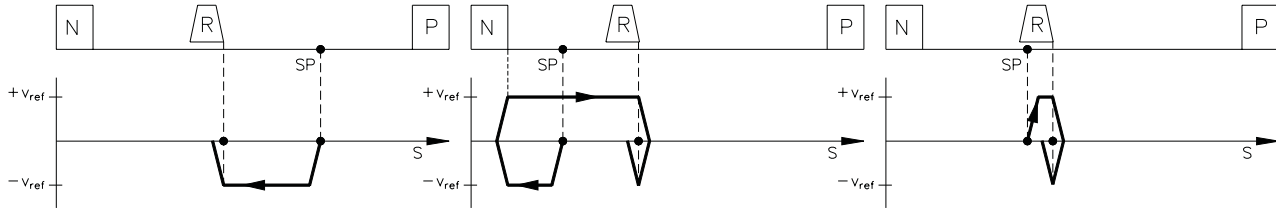
Attention !
Hardware limit switches must be present and connected.
The STOP limit switch function must be activated (Mux 80).

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

Reference traverse 3 negative

(with reference switch, negative traverse direction, 3 initial situations, positive count direction, no zero mark)



Initial point positive from reference switch

Initial point negative from reference switch

Initial point on the reference switch



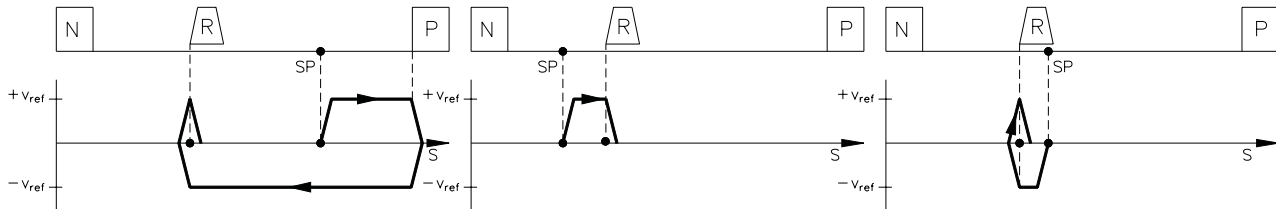
Attention !

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 (Mux 80).

Reference traverse 3 positive

(with reference switch, positive traverse direction, 3 initial situations, positive count direction, no zero mark)



Initial point positive from reference switch

Initial point negative from reference switch

Initial point on the reference switch



Attention !

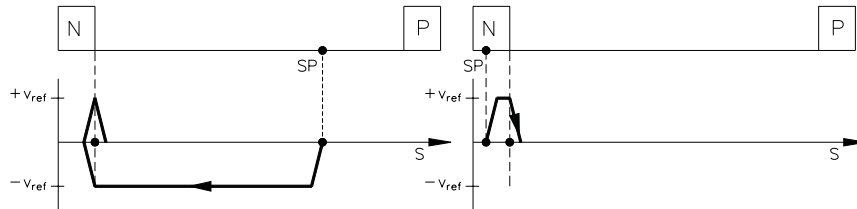
Before starting the reference traverse, check the safety of the system to ensure that the load can also be moved if the limit switches are disconnected or defective.

The STOP limit switch function must be activated to achieve the full reference traversing functionality (Mux 80).

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

Reference traverse 4 negative

(without reference switch, negative traverse direction, 2 initial situations, positive count direction, no zero mark)



Initial point positive from limit switch

Initial point on the limit switch



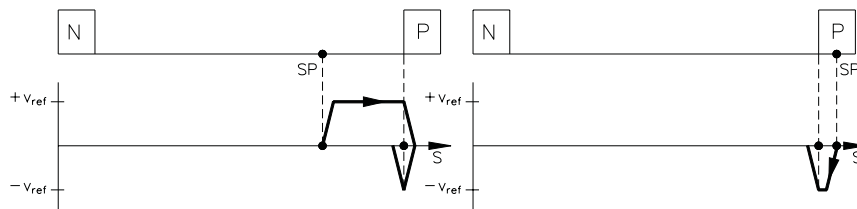
Attention !

Hardware limit switches must be present and connected.

The STOP limit switch function must be activated (Mux 80).

Reference traverse 4 positive

(without reference switch, positive traverse direction, 2 initial situations positive count direction, no zero mark)



Initial point negative from limit switch

Initial point on the limit switch



Attention !

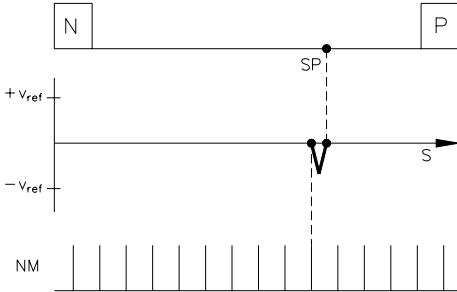
Hardware limit switches must be present and connected.

The STOP limit switch function must be activated (Mux 80).

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

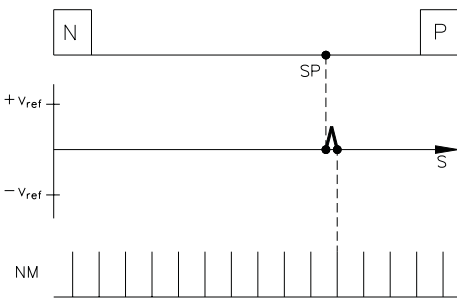
Reference traverse 5 negative

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



Reference traverse 5 positive

(without reference switch, negative traverse direction, positive count direction, with zero mark)



V.3.20 Mode

The servo-amplifier can be changed over between 5 different modes (Mux 48, see also Chapter IV.1.10). For digital speed and torque control (current control) enter the setpoint for the speed or current control via the bus (Mux 49, see Chapter IV.2). Modes 1 and 2 are not possible with the standard CAN instrument.

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



Never change the mode while the motor is running !

When the amplifier is enabled, a change of mode is only permitted if the speed is 0. Set the speed to 0 before changing over.

VI Appendix

VI.1 User notes and examples

VI.1.1 Communication setup

VI.1.1.1 Commissioning the CAN-bus master

Since a great variety of equipment is available for controlling CAN systems on various platforms (PC, PLC, other controls), it is only possible to give general advice:

- The CAN (High-Speed) Standard ISO 11898 must be implemented in the master. This affects the data frames which are used, as described in Chapter III.1.1, and the realisation of Layers 1 and 2 of the ISO/OSI model.
- CAN Specification 2.0 A, in which the data frame for the protocol and transport of an 11-bit COB (Communication Object)-ID is defined, must be fulfilled.
- Processing software for the CAN-Connect protocol of the servo amplifier must be available in the master. This must be created by the customer, according to the protocol description.

VI.1.1.2 Connection test master ↔ digifas™

When the station address is set at "0" on the servo amplifier, the fault register of the digifas™ can be requested with a remote frame by the master.

The COB identifier for this purpose is 000H (the bits 3-10 (station address) are 0, the fault object is requested with ID2 = ID1 = ID0 = 0 (see Chapter III.2.1). If the communication setup is successful (response monitoring is not activated) then the appropriate fault register is transmitted back by the digifas™ in a data frame.

VI.1.2 Examples of various functions

The numbers which are given in the examples are to be interpreted as decimal numbers, unless an H (for Hexadezimal) is appended (station address 0 ⇒ bit 3-10 = 0).

VI.1.2.1 Weighting factors

Conditions: output stage disabled, response monitoring inactive.

Disable amplifier

COB-ID	:	2	master ⇒ slave
Multiplexer	:	3	controller enable OFF
Variable	:		none

Setting the weighting factors for position and speed

Position

COB-ID	:	2	master ⇒ slave
Multiplexer	:	126	position weighting factor
Variable	:	2	division of the position setpoint by 10 ²

Speed

COB-ID	:	2	master ⇒ slave
Multiplexer	:	127	speed weighting factor
Variable	:	1	division of the speed setpoint by 10 ¹

VI.1.2.2 Reference traverse (“go to reference”)

Weighting factors as described in Chapter VI.1.2.1

Start reference traverse

COB-ID	:	2	master ⇒ slave
Multiplexer	:	13	reference traverse
Variable	:	0H 0H 7H D0H	= 2000 in Motorola format => speed = 200 mm/s

Traverses until the reference switch is activated or the position reached is defined as a reference.

Set reference point

COB-ID	:	2	master ⇒ slave
Multiplexer	:	14	set reference point
Variable	:		none

VI.1.2.3 Jog mode

Weighting factors as described in Chapter VI.1.2.1

Conditions: hardware and controller enable set via the bus, see Chapter IV.3.1

Enable amplifier

COB-ID	:	2	master ⇒ slave
Multiplexer	:	2	enable
Variable	:		none

Jog command

COB-ID	:	2	master ⇒ slave
Multiplexer	:	11	jog
Variable	:	0H 0H 3H E8H	= 1000 in Motorola format => speed = 100 mm/s

VI.1.2.4 Writing and performing a motion block

Weighting factors as described in Chapter VI.1.2.1

Write a local motion block

Write the position

COB-ID	:	2	master ⇔ slave
Multiplexer	:	15	write position value
Variable	:	100 000	=> position = 1000 mm

Write the speed

COB-ID	:	2	master ⇔ slave
Multiplexer	:	16	write speed value
Variable	:	4000	=> speed = 400 mm/s

Write the acceleration/braking ramp

COB-ID	:	2	master ⇔ slave
Multiplexer	:	17	write ramps
Variable	:	0H C8H 0H 96H	=> acceleration ramp = 200 ms braking ramp = 150 ms

Write the type of motion task

COB-ID	:	2	master ⇔ slave
Multiplexer	:	20	write motion task
Variable	:	0	absolute task

Save the local motion task in the motion task-memory

COB-ID	:	2	master ⇔ slave
Multiplexer	:	18	save local motion task
Variable	:	12	save as motion task 12

Enable amplifier

Condition: hardware enable must also be set

COB-ID	:	2	master ⇔ slave
Multiplexer	:	2	enable
Variable	:		none

Start motion task

Conditions: (see also section "Starting a motion task"):

- controller enabled (SR bit 1 = 1)
- reference point set (SR bit 12 = 1)
- no Quick Stop (SR bit 16 = 0)
- no Pause (SR bit 18 = 0)

COB-ID	:	2	master ⇔ slave
Multiplexer	:	12	start motion task
Variable	:	12	motion task 12 from the motion-task memory

VI.1.2.5 Direct motion tasks / examples of response times

Weighting factors as described in Chapter VI.1.2.1

There are several possibilities, with different execution times, for the definition and start of direct motion tasks (= local motion task, motion-task number = 0):

Transmission of the motion task as components and start of the motion task

(as for writing a local motion task)

In this way the execution time is

$$T_R \leq 8 \text{ ms} + 1 \text{ ms (Stop command)}$$

Stop command (if required)

COB-ID	:	2	master ⇒ slave
Multiplexer	:	1	Stop
Variable	:		none

Definition of a direct motion-task object

The direct motion-task object is used to transmit the speed and the ramps.

In this way the execution time is

$$T_R \leq 4 \text{ ms} + 1 \text{ ms (Stop command)}$$

COB-ID	:	6	master ⇒ slave
Multiplexer	:	none	no control object
Variable	:	5000	=> speed = 500 mm/s
		0H C8H 0H 96H	=> acceleration ramp = 200 ms braking ramp = 150 ms

The direct motion-task object can then be started with a special multiplexer:

Start a direct motion-task object

COB-ID	:	2	master ⇒ slave
Multiplexer	:	121/122	start direct motion-task absolute/relative
Variable	:	100000	=> position = 1000 mm

If all the components of the local motion task can be re-used, apart from the position, then the reaction time can be further reduced.

COB-ID	:	2	master ⇒ slave
Multiplexer	:	121/122	start direct motion-task absolute (relative)
Variable	:	180000	=> new position = 1800 mm

In this case the response time can be $T_R < 3 \text{ ms}$ (+ Stop command).

VI.1.2.6 Broadcast operation

broadcast groups (see Chapter III.2.4) should be formed when the system is set up. The assignment of stations to groups can be changed by simply overwriting the broadcast-group numbers.

It is possible for very problematic operating states to arise in a complex application. In this case the user must bear the responsibility.

Groups of units can be joined together on the CAN bus into groups, with the aid of the command "Activate stations for Broadcast commands". In the example, the units with the numbers 1, 5 and 23 are combined to form a broadcast group.

COB-ID	:	00AH	master ⇔ digifas™ 1
Multiplexer	:	24	broadcast ON
Variable	:	1	broadcast group 1
COB-ID	:	02AH	master ⇔ digifas™ 5
Multiplexer	:	24	broadcast ON
Variable	:	1	broadcast group 1
COB-ID	:	0BAH	master ⇔ digifas™ 23
Multiplexer	:	24	broadcast ON
Variable	:	1	broadcast group 1

On the assumption that all three digifas™ have motion blocks from 1 to 10 stored in the memory, a simultaneous start of the servo amplifiers can now be made.

Conditions:	digifas™ group 1 enabled	(SR bit 9 = 1)
	digifas™ group 1 reference point set	(SR bit 12=1)
	digifas™ group 1 no Quick Stop	(SR bit 16 = 0)
	digifas™ group 1 no Pause	(SR bit 18 = 0)
	(otherwise as in the section "Start a motion task")	

COB-ID	:	00FH	master ⇔ broadcast group 1
Multiplexer	:	12	start a motion task
Variable	:	7	motion block 7 from the motion-task memory all amplifiers in broadcast group 1

The responses are returned as described in Chapter III.2.4 . It is a matter for the user, whether and how these responses are evaluated.

COB-ID	:	009H	digifas™ 1 ⇔ master
Multiplexer	:	12(140)	motion block (not) started
Variable	:	7	(fault number) motion-block number
COB-ID	:	029H	digifas™ 5 ⇔ master
Multiplexer	:	12(140)	motion block (not) started
Variable	:	7	(fault number) motion-block number
COB-ID	:	0B9H	digifas™ 23 ⇔ master
Multiplexer	:	12(140)	motion block (not) started
Variable	:	7	(fault number) motion-block number

The broadcast group can then be disbanded by:

COB-ID	:	00FH	master ⇔ broadcast group 1
Multiplexer	:	25	broadcast OFF
Variable	:	1	broadcast group 1

VI.1.2.7 Fault evaluation

If the amplifier does not react as expected, and the reason is not immediately apparent, then the status register should be requested (here again, an example for station address 0):

COB-ID : 3 master ⇔ slave

The 32-bit status register is then transmitted back, which contains the relevant status bits for the operation of the digifas™ (see Chapter II.2.3). For example, it can be seen whether the amplifier has been enabled via the bus (bit 23) and overall (bus & hardware).

Bits 0 to 22 are relevant for the evaluation of faults.

If serious faults occur, this will be indicated by bit 3 = 1 (fault). These faults (e.g. motor temperature, resolver fault) can also cause the red fault-LED to light up on the digifas™.

The fault evaluation can be made through the fault register (see Chapter III.2.1).

COB-ID : 0 master ⇔ slave

The 32-bit fault register is then returned. In this case the bits 0 to 14 are of importance. For example, if the maximum motor temperature is exceeded this is shown by bit 2 = 1, or a resolver fault by bit 14 = 1.

Fault signals can only be removed by switching the amplifier off and on again (after removing the cause of the fault). The bus monitoring is an exception. By using the command

COB-ID : 2 master ⇔ slave
 Multiplexer : 10 acknowledge contouring error / response monitoring
 Variable : none

the response monitoring can be reset. The bus communication should then continue to function without any faults.

VI.3 Index

A	abbreviations	I-2	linear axis, commissioning	II-6
	acceleration/braking ramp	IV-10	load from EEPROM	IV-12
	acknowledge contouring error	IV-2	M master, commissioning	VI-1
	acknowledge response monitoring	IV-2	mode	V-14
	actual values	IV-9	motion block, example	VI-3
	actual values, Connect page	V-1	motor inductance	IV-14
	analog/digital setpoint entry	IV-4	multiplexer, list	III-5
	assembly	II-1	N numerical format	I-3
	axis type	V-6	P pause	IV-2
B	baud rate recognition	IV-3	pin assignments digifas®	II-3
	baud rate, read	IV-12	position value	IV-10
	baud rate, set	V-7	position weighting factor	IV-13
	brake	IV-2	Q Quick Stop	IV-1
	broadcast	IV-2	R ramp type	V-7
	broadcast object	III-10	Reference offset	V-8
	broadcast operation, example	VI-5	reference point	IV-6
	bus cable	I-3	reference traverse	V-9
C	COB	III-1	reference traverse, example	VI-2
	commissioning	II-5	remote frame	III-1
	commissioning, master	VI-1	resolution	V-4
	communication object	III-1	response monitoring	V-7
	connection diagram	II-2	response times	I-4
	connection methods	II-1	rotary axis	V-6
	connection test	VI-1	rotary axis, commissioning	II-8
	contouring error	V-5	S s_actual	V-1
	control functions	IV-1	s_lag	V-1
	control object	III-4	safety instructions	I-D
	controlled by	V-7	save to EEPROM	IV-13
	controller enable	IV-1	save to EEPROM/RAM	IV-12
	controller parameter	IV-12	sine ²	V-7
	count direction	V-5	speed value	IV-10
D	data frame	III-1	speed weighting factor	IV-13
	digital speed control	IV-5	start a direct-motion task	IV-8
	digital speed setpoint	IV-5	start a motion task	IV-7
	digital torque control	IV-5	station address	II-4
	digital torque setpoint	IV-5	status object	III-8
	direct motion task, example	VI-4	status register	III-9
	direct motion-task object	III-12	status register mask	IV-3
E	Enable	IV-1	STOP	IV-1
	error index	IV-16	T t_accel_min, maximum acceleration	V-3
	error log	IV-16	t_emerg, maximum deceleration	V-3
	error statistics	IV-16	Teach In	IV-14
F	fault evaluation, notes	VI-6	transmission rate	V-7
	fault object	III-2	trapeze	V-7
	fault signals	IV-15	type of motion task	IV-11
	Ff, feed-forward factor	V-3	Type of reference traverse	V-9
	fieldbus monitor	V-2	U Use as directed	I-1
G	go to reference	IV-7	V v_actual	V-1
I	In Position	V-5	v_max, maximum speed	V-3
	installation	II-1	W weighting factor, example	VI-2
J	jog mode	IV-6	Z zero point offset	V-5
	jog mode, example	VI-2		
K	KE, motor voltage constant	IV-14		
	Kp, P-gain	V-3		
L	L, motor inductance	IV-14		
	limit switch	V-6		
	linear axis	V-6		

This page has been deliberately left blank

Vertrieb und Service / Sales and Service / Agence et Services

<u>Bundesrepublik Deutschland / Germany / Allemagne</u> Kollmorgen Seidel GmbH & Co. KG Verkaufsniederlassung Nord Wacholderstr. 40-42 40489 Düsseldorf Tel.: +49(0)203 - 99 79 214 Fax: +49(0)203 - 99 79 182	<u>Dänemark / Denmark / Danemark</u> DIGIMATIC A/S "Laerkefeldt" Aalkaergaardvej 20 8700 Horsens Nord Tel.: +45 - 75 65 66 66 Fax: +45 - 75 65 68 33	<u>Italien / Italy / Italie</u> M.C.A. s.r.l. Via f. Turati 21 20016 Pero (Mi) Tel.: +39(0)02 - 33 91 04 50 Fax: +39(0)02 - 33 90 85 8
Kollmorgen Seidel GmbH & Co. KG Verkaufsniederlassung West Lilienstraße 3 42719 Solingen Tel.: +49(0)212 - 2 30 77 99 Fax: +49(0)212 - 2 30 77 97	<u>Finnland / Finland / Finlande</u> Drivematic OY Hevosenkentä 4 28430 Pori Tel.: +358 - 2 - 61 00 33 11 Fax: +358 - 2 - 61 00 33 50	<u>Niederlande / Netherlands / Pays-Bas</u> Dynamic Drives Wattstraat 26f 2723 RC Zoetermeer Tel.: +31(0)79 - 59 39 214 Fax: +31(0)79 - 59 39 840
Kollmorgen Seidel GmbH & Co. KG Verkaufsniederlassung Mitte Bussardweg 38 61118 Bad Vilbel Tel.: +49(0)6101 - 55 866 00 Fax: +49(0)6101 - 55 866 06	<u>Frankreich / France / France</u> Kollmorgen Seidel GmbH & Co. KG Parc technologique St.Jacques 2 rue Pierre et Marie Curie 54320 Maxéville Tel.: +33(0)3 83 95 44 80 Fax: +33(0)3 83 95 44 81	<u>Schweden / Sweden / Suède</u> S D T AB 25467 Helsingborg Tel.: +46(0)42 - 380 800 Fax: +46(0)42 - 380 813 Stockholm 12030 Stockholm Tel.: +46(0)8 - 640 77 30 Fax: +46(0)8 - 641 09 15 Göteborg 42671 Västra Frölunda Tel.: +46(0)31 - 69 62 60 Fax: +46(0)31 - 69 62 69
Kollmorgen Seidel GmbH & Co. KG Verkaufsniederlassung Süd-West Lessingstr. 41 75015 Bretten Tel.: +49(0)7252 - 97 39 040 Fax: +49(0)7252 - 97 39 055	Kollmorgen Seidel GmbH & Co. KG 216 Lotissement Les Peiffendes Le Sonnant d'Uriage 38410 Uriage Tel.: +33(0)4 76 59 22 30 Fax: +33(0)4 76 59 22 31	<u>Schweiz / Switzerland / Suisse</u> Kollmorgen Seidel GmbH & Co. KG Bühnrain 30 8052 Zürich Tel.: +41(0)1 - 300 29 65 Fax: +41(0)1 - 300 29 66
Kollmorgen Seidel GmbH & Co. KG Verkaufsniederlassung Süd-Ost Landsbergerstr. 17 86947 Weil Tel.: +49(0)8195 - 99 92-50 Fax: +49(0)8195 - 99 92-33	<u>Großbritannien / Great Britain / Royaume-Uni</u> 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	<u>Spanien / Spain / Espagne</u> BROTOMATIC S.L. C/San Miguel de Acha, 2 Pab.3 01010 Vitoria (ALAVA) Tel.: +34 945 - 24 94 11 Fax: +34 945 - 22 78 32
Servo-Dyn Technik GmbH Münzgasse 10 01067 Dresden Tel.: +49(0)351 - 49 05 793 Fax: +49(0)351 - 49 05 794	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	

Systempartner / System partners / Partenaires du système

<u>Bundesrepublik Deutschland / Germany / Allemagne</u> Werner P. Hermes Ingenieurbüro Turmstr. 23 40750 Langenfeld Tel.: +49(0)212 - 65 10 55 Fax: +49(0)212 - 65 10 57	<u>Großbritannien / Great Britain / Royaume-Uni</u> Motor Technology Ltd. Unit 1 Chadkirk Industrial Estate Otterspool Road Romiley, Stockport Cheshire SK6 3LE Tel.: +44(0)161 - 42 73 641 Fax: +44(0)161 - 42 71 306	<u>Italien / Italy / Italie</u> Servo Tecnica Viale Lombardia 20 20095 Cusano Milanino (MI) Tel.: +39 (0)02 - 66 42 01 Fax: +39 (0)02 - 66 40 10 20
EAT GmbH Elektronische Antriebstechnik Hanferstraße 23 79108 Freiburg Tel.: +49(0)761 - 13 03 50 Fax: +49(0)761 - 13 03 555	<u>Niederlande / Netherlands / Pays-Bas</u> Kiwiet Ingenieurbüro Helenaveenseweg 35 5985 NK Panningen (Grashoek) Tel.: +31(0)77 - 30 76 661 Fax: +31(0)77 - 30 76 646	<u>Türkei / Turkey / Turquie</u> Robotek Otomasyon Teknolojileri Ali Nihat Tarlan CAD. Kartal Sk. No: 16/7 Üstbostancı YSTANBUL Tel: +90 216 464 50 64 pbx Fax: +90 216 464 50 72
IBK Ingenieurbüro Keßler GmbH Dachtmisser Str. 10 21394 Kirchgellersen Tel.: +49(0)4135 - 12 88 Fax: +49(0)4135 - 14 33	<u>Schweiz / Switzerland / Suisse</u> Bobby Servo Electronic AG Zentralstr. 6 6030 Ebikon Tel.: +41(0)41- 440 - 77 22 Fax: +41(0)41- 440 - 69 43	<u>Griechenland / Greece / Grèce</u> Alpha Motion 5 - 7 Alkamenoy's Str. 104.39 Athens Tel.: +30 1 82 27 470 Fax: +30 1 82 53 787
MACCON GmbH Kühlbachstr. 9 81543 München Tel.: +49(0)89 - 65 12 20-0 Fax: +49(0)89 - 65 52 17	<u>Ungarn / Hungary / Hongrie</u> Q-TECH Mernöki Szolgáltató Kft. 1161 Budapest Batthyány u. 8. Tel.: +36 (1) 405 - 33 38 Fax: +36 (1) 405 - 91 34	<u>Australien / Australia / Australie</u> Motion Technologies PTY. Ltd. 1/65 Alexander Avenue Taren Point NSW 2229 Sydney Tel.: +61 (0)295 24 47 82 Fax: +61 (0)295 25 38 78

Kollmorgen Seidel GmbH & Co. KG

Hausanschrift

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

Postanschrift

Postfach 34 01 61
D-40440 Düsseldorf

Kollmorgen

Motion Technologies Group

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