FAULHABER

Motion Control

Series MCBL 300x CF
Series MCDC 300x CF
Series 3564...B CC
Series 32xx...BX4 CC
Series 22xx...BX4 CCD

Communication/ Function Manual

EN

CANopen

with **FAULHABER CAN**

WE CREATE MOTION

Imprint

Version:
4th edition, 27.08.2014

This technical manual has been prepared with care. Dr. Fritz Faulhaber GmbH & Co. KG cannot accept any liability for any errors in this technical manual or for the consequences of such errors. Equally, no liability can be accepted for direct or consequential damages resulting from improper use of the equipment.

The relevant regulations regarding safety engineering and interference suppression as well as the requirements specified in this technical manual are to be noted and followed when using the software.

Subject to change without notice.

The respective current version of this technical manual is available on FAULHABER's internet site:
www.faulhaber.com

Overview

Overview of the FAULHABER Motion Control Drives documents

Document	Contents
Technical Manual	Device installation, safety, specification
Communication and function manual (CANopen)	Initial start-up, function overview, protocol description and parameter description.
Motion Manager instruction manual	Operation of the "FAULHABER Motion Manager" PC software for configuration and commissioning
Product data sheets	Technical limit and operating data

Guide to the Document

Notes on the initial start-up of a FAULHABER Motion Control system at the PC in the default configuration

Quick Start Page 9

Overview of the possible operating modes in Faulhaber mode

Operation in FAULHABER mode Page 17

Specification of the CANopen communication protocol

CANopen protocol description Page 59

Extended CAN functions Page 72

Overview of the supported drive profiles according to CiA 402

Functional description of the CANopen CiA 402 Page 74

Detailed description of the parameters for the implemented Function blocks within the drive

Commissioning Page 98

Description of all the drive's parameters and commands, broken down into functional areas

Parameter description Page 122

Table of Contents

1 Important Information 7

1.1 Symbols used in this manual 7

1.2 Additional information 8

2 Quick Start 9

2.1 Set node number and baud rate 9

2.2 Operation using FAULHABER Motion Manager 10

2.2.1 Activate CANopen nodes 10

2.2.2 Configuring the drives 10

2.2.3 Operation using FAULHABER commands 11

2.2.4 Operation in one of the CANopen CiA 402 drive profiles 11

2.3 Operation using own host application 15

2.3.1 Activate CANopen nodes 15

2.3.2 Configuring the drives 15

2.3.3 Operation using FAULHABER commands 15

2.3.4 Operation in one of the CANopen CiA 402 drive profiles 16

3 Operation in FAULHABER mode 17

3.1 Position control 19

3.1.1 Set-point presetting via CAN/PDO2 19

3.1.2 Analog positioning mode (APCMOD) 22

3.1.3 External encoder as actual position value (ENCMOD) - not for MCDC 24

3.2 Velocity control 27

3.2.1 Target velocity via CAN/PDO2 28

3.2.2 Velocity presetting via an analog voltage or a PWM signal 30

3.2.3 External encoder as actual velocity value (ENCMOD) - not for MCDC 32

3.3 Homing and limit switches 35

3.3.1 Limit switch connections and switching level 36

3.3.2 Motion control commands 37

3.3.3 Configuration of homing and limit switches 38

3.4 Extended operating modes 40

3.4.1 Stepper motor mode 40

3.4.2 Gearing mode (electronic gear) 42

3.4.3 Voltage regulator mode 44

3.4.4 Current control with analog current presetting 45

3.4.5 IxR control for MCDC 47

3.5 Special fault output functions 48

Table of Contents

3.6 Technical information 50
3.6.1 Ramp generator 50
3.6.2 Sinus commutation 54
3.6.3 Current controller and I²t current limitation 54
3.6.4 Overtemperature protection 56
3.6.5 Under-voltage monitoring 56
3.6.6 Overvoltage regulation 56
3.6.7 Adjustment of the controller parameters 56

4 CANopen protocol description 59
4.1 Introduction 59
4.2 PDOs (process data objects) 61
4.3 SDO (service data object) 63
4.4 Emergency Object (error message) 65
4.5 SYNC object 66
4.6 NMT (network management) 67
4.7 Entries in the object dictionary 70

5 Extended CAN functions 72
5.1 The FAULHABER channel 72
5.2 Trace 72

6 Functional description of the CANopen CiA 402 74
6.1 Device Control 75
6.1.1 State machine of the drive 75
6.1.2 Selection of the operating mode 79
6.2 Factor Group 80
6.3 Profile Position Mode and Position Control Function 82
6.4 Homing Mode 87
6.5 Profile Velocity Mode 91
6.6 Drive parameters/Common entries 94
6.7 Inputs/Outputs 95
6.8 Error handling 97

Table of Contents

7 Commissioning 98

7.1 Node number and baud rate 98
7.2 Basic settings 100
7.3 Configuration using the Motion Manager 101
7.3.1 Connection setting 102
7.3.2 Motor selection 103
7.3.3 Drive configuration 103
7.3.4 Selection of the operating mode 104
7.4 Configuration in FAULHABER mode 105
7.4.1 Basic settings 105
7.4.2 Drive parameters 108
7.4.3 Controller settings 109
7.4.4 I/O connection and use 111
7.5 Configuration in a drive profile according to CIA 402 113
7.5.1 Basic settings 113
7.5.2 Drive parameters 115
7.5.3 Controller setting 116
7.5.4 I/O connection and use 118
7.6 Data set management 120
7.7 Status display 121
7.7.1 Trace function 121

8 Parameter description 122

8.1 Communication objects according to CiA 301 122
8.2 Manufacturer-specific objects 128
8.3 Drive profile objects according to CiA 402 130
8.4 FAULHABER commands 136
8.4.1 Basic setting commands 137
8.4.2 Query commands for basic settings 141
8.4.3 Miscellaneous commands 143
8.4.4 Motion control commands 144
8.4.5 General query commands 145
8.4.6 Command overview 146

1 Important Information

1.1 Symbols used in this manual

WARNING! ***Warning!***

This pictogram with the wording "Warning!" indicates an imminent danger which can result in physical injuries.

► *This arrow points out the appropriate action to take to prevent the imminent danger.*

CAUTION! ***Caution!***

This pictogram with the wording "Caution!" indicates an imminent danger which can result in slight physical injuries or material damage.

► *This arrow points out the appropriate precautions.*

REGULATION! ***Regulations, guidelines and directives***

This pictogram with the wording "Regulation" indicates a statutory regulation, guideline or directive which must be observed in the respective context of the text.

NOTE ***Note***

This "Note" pictogram provides tips and recommendations for use and handling of the component.

1 Important Information

1.2 Additional information

WARNING! ***Risk of injuries***

Failure to comply with the safety instructions during installation and operation can result in irreparable damage to the device and a risk of injuries to the operating personnel.

- ► *Please read through the whole of your drive's technical manual before installing the drive.*
- ► *Keep this communication and function manual in a safe place for subsequent use.*

NOTE

Always use the current version of the FAULHABER Motion Manager.
The respective current version is available to download from www.faulhaber.com/Motion Manager.

NOTE

The information given in this instruction manual refers to the standard version of the drives.
Please refer to any additional information sheet provided in the event of differences in information due to a customer-specific motor modification.

NOTE

Motion Controllers with a CANopen interface are designed as slaves in a CANopen environment and always require a connection with a CANopen Master to operate.

2 Quick Start

To facilitate introduction, this chapter highlights the initial steps for commissioning and operation of FAULHABER Motion Controllers with CANopen interface. However, the detailed documentation must always be read and taken into account, particularly Chapter 7.2 “Basic settings”!

2.1 Set node number and baud rate

The standard units are delivered without valid node address (Node ID = 0xFF) and with automatic baud rate detection set.

In order to set the baud rate and node address, the unit must first be connected via CAN to an appropriate configuration tool, which supports the LSS protocol according to CiA DSP305.

NOTE

FAULHABER Motion Manager installed on a PC with supported CAN interface can also be used for this. The LSS compatible configuration tool can be used to set the node address and baud rate, either in Global mode, if only one drive is connected, or in Selective mode via the serial number, if a drive is to be configured in the network (see Chapter 7.1 “Node number and baud rate”).

If the FAULHABER Motion Manager is to be used as the configuration tool, proceed as follows:

The following steps are necessary for commissioning using the default configuration:

1. Connect the drive unit to a voltage source (24V). For details of connection cable assignment and the operating voltage range of the drive, see Chapter 3 “Installation” in the technical manual.
2. Connect drive unit to the CAN interface of the PC and switch on or connect PC to the CAN network.
3. Start FAULHABER Motion Manager.
4. Activate CAN interface as communication interface and configure using the menu item “Terminal – Connections…” or the Connection Wizard.
5. Select menu item “CAN - LSS (DSP305)…”.
6. Select Configuration mode:
   a. Globally configure individual drive (LSS Switch Mode Global) if only one LSS node is connected and you do not want to enter any further data.
   b. Selectively configure specified nodes (LSS Switch Mode Selective) if a node is to be configured in the network. If the node has not yet been found in the Node Explorer, enter the serial number of the drive node to be configured here, otherwise the data fields are already correctly preset.
7. In the next dialog, select the required transfer rate or “Auto” and enter the required node number.
8. Press “Send” button.
9. The settings are transferred and are permanently stored in the controller. The Motion Manager then calls up the Scan function again and the node should now be displayed with the correct node number in the Node Explorer. After switching off and on again, the drive will operate with the set configuration.

2 Quick Start

2.2 Operation using FAULHABER Motion Manager

The FAULHABER Motion Manager provides easy access to the CANopen state machine using menu entries, which can be opened either with the Node Explorer's context menu (right-click) or with the "CAN" menu. The required node must have been activated beforehand by double clicking in the Node Explorer. The current statuses are always displayed in the status line at the bottom edge of the Motion Manager window.

Further information on the state machine of a CANopen node is given in Chapter 4 "CANopen protocol description".

NOTE

The FAULHABER commands described below can be entered directly in the command input line or selected from the Commands menu. After sending the command, a command interpreter is activated which converts the command into a corresponding CAN message frame on PDO2.

2.2.1 Activate CANopen nodes

In order to drive a motor using the Motion Manager, follow the procedure below (assuming a valid node number and matching baud rate are set):

1. **Start network nodes.**

   Select the "CANopen Network Management (NMT) – Start Remote Node" entry in the Node Explorer's context menu or in the "CAN" menu.

   The state of the node is then "Operational", FAULHABER commands are now available!

2. **Configure drive functions:**

   A user-friendly dialog that enables the desired settings to be made is available under the menu item "Configuration – Drive functions…".

2.2.2 Configuring the drives

CAUTION!

Check basic settings

Incorrect values in the Motion Controller's settings can result in damage to the controller and/or drive.

Motion Control systems with electronics built-onto the motor are already preset in the factory. Motion controllers with an externally connected motor must be equipped with current limitation values suitable for the motor and suitable controller parameters before being started up. The Motor Wizard is available in Motion Manager for selection of the motor and basic parameters suitable for the motor.

Other settings, e.g. for the function of the fault output, can be made under the "Configuration – Drive functions" menu item, where a convenient dialog is provided (see Chapter 7.3 "Configuration using the Motion Manager"). The configuration dialog is also available for direct access in the Wizard bar of the Motion Manager (Configuration Wizard).

Switch to the required mode (Modes of Operation/OPMOD 1, 3, 6 or –1), depending on whether you want to operate the drive using the standard CANopen objects or the FAULHABER commands.

2.2.3 Operation using FAULHABER commands

3. Activate drive:

"**EN**" command.

Input in command input field and press "Send" button or select the "Enable Drive (EN)" button or the relevant entry from the "Commands – Motion Control" menu.

4. Operate drive (examples):

- Operate drive with 100 rpm velocity control:

  "**V100**" command.

  Enter in command input field and press "Send" button or

  select from "Commands – Motion control – Initiate Velocity Mode (V)" menu, enter value 100 in dialogue box, press OK and "Send" button.

- Stop drive:

  "**V0**" command.

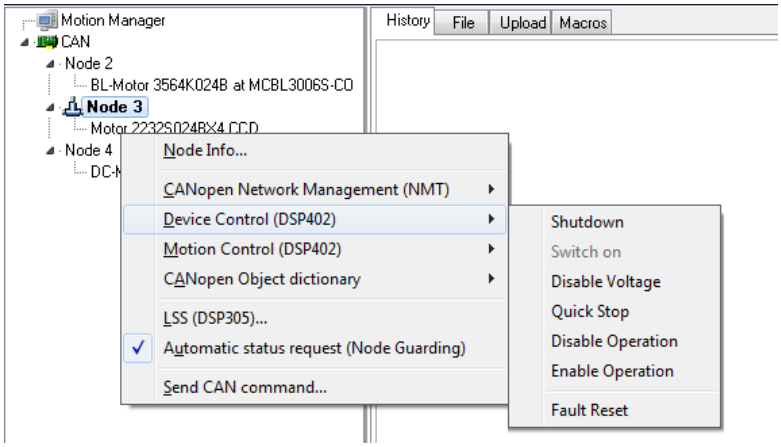
- Move motor relatively by 10 000 increments:

  "**LR10000**" command to load the relative target position, "M" command to move to loaded target position.

2.2.4 Operation in one of the CANopen CiA 402 drive profiles

3. Activate drive using the CiA 402 state machine:

A CiA 402 drive must be activated according to a fixed sequence of steps. The necessary commands are directly available in the context menu of the drive node:



- Shutdown

  Select "Device Control (DSP402) – Shutdown" entry using the context menu in Node Explorer or using the "CAN" menu.

- Switch On

  Select the "Device Control (DSP402) – Switch On" entry using the context menu in Node Explorer or using the "CAN" menu.

2 Quick Start

2.2 Operation using FAULHABER Motion Manager

- Enable Operation

  Select the “Device Control (DSP402) - Enable Operation” entry using the context menu in Node Explorer or using the “CAN” menu.

Alternatively, you can also simply press the green “Switch on output stage” button or F5, in order to carry out these steps all at once.

4. **Drive motor (examples):**

   Drive motor with 100 rpm velocity control:

   Set Profile Velocity mode:

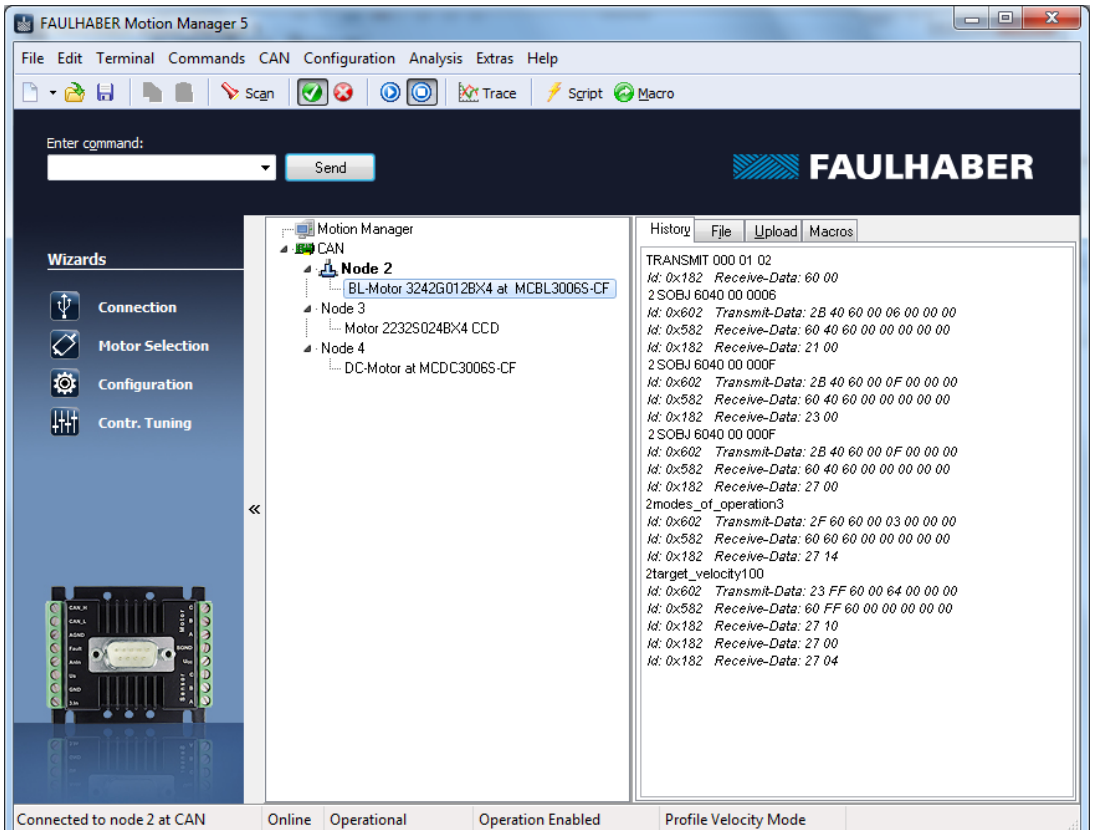
   - Select the “Motion Control (DSP402) - modes_of_operation (6060h)” entry in the node explorer’s context menu or in the “CAN” menu.
   - Enter value 3 for the “profile velocity mode” in the dialogue box → the necessary command is entered directly in the command field of the Motion Manager.
   - Press “Send” button next to the command field.

2 Quick Start

2.2 Operation using FAULHABER Motion Manager

Set Target Velocity to value 100 (Object 0x60FF):

- Select the “Motion Control (DS402) - target_velocity (60FFh)” entry using the context menu of the Node Explorer or using the “CAN” menu.
- Enter value 100 for the target velocity in the dialogue box → the necessary command is entered directly in the command field of the Motion Manager.
- Press “Send” button next to the command field.



Stop motor:

- Set Target Velocity to value 0 (Object 0x60FF) or
- Select “Disable Operation” from the toolbar.

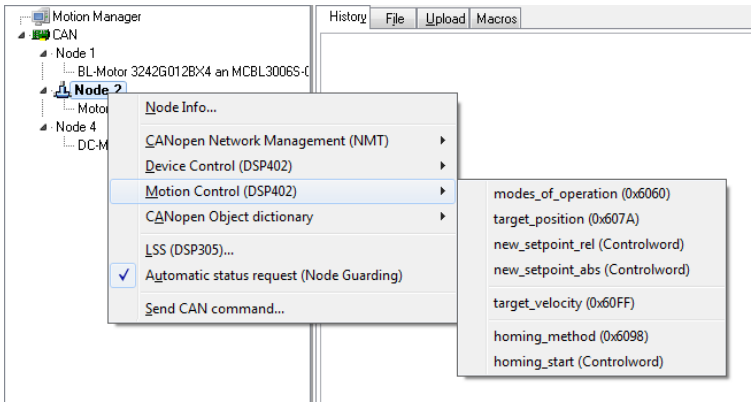
2 Quick Start

2.2 Operation using FAULHABER Motion Manager

Move motor relatively by 10 000 increments:

Set Profile Position mode:

- Select the “Motion Control (DSP402) - modes_of_operation (6060h)” entry in the node explorer’s context menu or in the “CAN” menu.
- Enter value 1 for the “Profile Position Mode” in the dialogue box → the necessary command is entered directly in the command field of the Motion Manager.
- Press “Send” button next to the command field.



Set Target Position to value 10 000:

- Select the “Motion Control (DSP402) - target_position (607Ah)” entry in the node explorer’s context menu or in the “CAN” menu.
- Enter value 10 000 for the target position in the dialogue box → the necessary command is entered directly in the command field of the Motion Manager.
- Press “Send” button next to the command field.

Move to target position: Set “New set-point” and “rel” in controlword.

- Select the “Motion Control (DSP402) - new_setpoint_rel” entry in the node explorer’s context menu or in the “CAN” menu.

2.3 Operation using own host application

2.3.1 Activate CANopen nodes

The broadcast command “Start Remote Node” with CAN ID 0 is used to start either an individual node or the whole network and to set it to “Operational“ status:

11 bit identifier	2 bytes user data	
ID 0x000	01	00

The first data byte contains the start command “Start Remote Node”, the second data byte contains the node address or 0 for the whole network.

All functions can be proceeded after the node has been started. The drive can now be activated and operated using the Device Control functions according to CiA DSP402 or using the FAULHABER commands on PDO2.

The identifiers of the individual objects are allocated according to the Predefined Connection Set and depend on the node number (see Chapter 4.6 “NMT (network management)”). Here are the most important objects:

Command	CAN ID	Description
TxPDO1	0x180 + Node ID	Receive the drive's statusword
RxPDO1	0x200 + Node ID	Transfer controlword to the drives
TxPDO2	0x280 + Node ID	Receive FAULHABER data from the drive
RxPDO2	0x300 + Node ID	Transfer FAULHABER command to the drive
TxSDO1	0x580 + Node ID	Read entry of the object dictionary
RxSDO1	0x600 + Node ID	Write entry of the object dictionary

In delivery status, after they are switched on, the drives are in operating mode Modes of operation = 1 (Profile Position Mode according to CIA 402). In this operating mode the drive is controlled using the Device Control state machine, which is operating using the controlword (Object 0x6040 or RxPDO1) and is queried using the statusword (Object 0x6041 or TxPDO1).

2.3.2 Configuring the drives

The drive can be configured both by means of SDO transfer using the objects of the object dictionary, and using PDO2 with the commands of the FAULHABER channel.

NOTE

Not all the configuration options are accessible using the object dictionary. Advanced operating modes are only available via the FAULHABER channel (see Chapter 8 “Parameter description”).

Use of the FAULHABER Motion Manager is recommended for the basic settings (see Chapter 7.2 “Basic settings”).

2.3.3 Operation using FAULHABER commands

All features of the drive can be operated even without in-depth CANopen knowledge, such as Device Control, SDO protocol and object dictionary. The FAULHABER channel on PDO2 provides an easy means of executing all supported commands.

RxPDO2: FAULHABER command

11 bit identifier	5 bytes user data				
0x300 (768d) + Node ID	Command	LLB	LHB	HLB	HHB

It is necessary to switch to operating mode Modes of operation = -1 first for drive control using the FAULHABER channel:

2 Quick Start

2.3 Operation using own host application

Example:

- Start node 3 using the CANopen Network Management (NMT):
  `ID 000: 01 03` (Start Remote Node 3)
- Switch to FAULHABER mode using RxPDO2:
  `ID 303: FD FF FF FF FF` (OPMOD-1)
- Switch On using FAULHABER command on RxPDO2:
  `ID 303: 0F 00 00 00 00` (EN)
  Start drive with 500 rpm using FAULHABER command on RxPDO2:
  `ID 303: 93 F4 01 00 00` (V500)

NOTE

All available commands are listed in Chapter 8.4 "FAULHABER commands".

2.3.4 Operation in one of the CANopen CiA 402 drive profiles

A CiA 402 drive must be activated according to a fixed sequence of steps (see Chapter 6.1 "Device Control"). Write access to the controlword is possible using the object dictionary at address 0x6040 or using the RxPDO1: .

1. **Shutdown:**
   Controlword = 0x00 06
2. **Switch on:**
   Controlword = 0x00 07
   The drive is then in "Switched On" status. Operation must then be released to enable drive commands to be executed.
3. **Enable Operation:**
   Controlword = 0x00 0F
   The drive is then in "Operation Enabled" state, in which it can be operated using the relevant objects of the set operating mode (see Chapter 6.1 "Device Control" and Chapter 6.2 "Factor Group").
4. **Drive motor (examples):**
   Drive motor with 500 rpm velocity control:
   Modes of operation (object 0x6060): Set 3 (profile velocity mode) by SDO access.
   Target velocity (object 0x60FF): 500
   Stop motor:
   - Set Target Velocity to value 0 (Object 0x60FF) or
   - Controlword = 0x00 07 (Disable Operation).

   Move motor relatively by 10 000 increments:
   Modes of Operation (Object 0x6060): Set 1 (Profile Position Mode) by means of SDO access.
   Target Position (Object 0x607A): 10 000
   Controlword = 0x00 7F (New set-point, Change set immediately, rel)

3 Operation in FAULHABER mode

Guide

Position control	Page 19
Velocity control	Page 27
Homing and limit switches	Page 35
Extended operating modes	Page 40
Special fault output functions	Page 48
Technical information	Page 50

The Motion Controllers can be configured for different operating modes. As a default, the drive unit is delivered as a servo motor in “Profile Position Mode” according to CiA DSP402.

It is necessary to set Modes of operation or OPMODE to -1 for operation control using the FAULHABER channel.

The drive can be reconfigured by means of the corresponding FAULHABER commands. If the settings are to be permanently stored, the SAVE command must be executed following configuration; this saves the current settings in the Flash data memory, from where they are reloaded when the unit is next switched on. Alternatively, the EEPSAV command can also be run. Both commands are identical, therefore SAVE only is used in the following.

NOTE

Operation of the drive in one of the operating modes listed here requires that the device is in NMT “Operational” state and that power stage is activated (“Switched On” or “EN” command). All commands and objects listed below are summarised and explained again in Chapter 8.4 “FAULHABER commands”. The FAULHABER commands transmitted as CAN message frames on PDO2 are given for each operating mode.

The FAULHABER Motion Manager enables simple setting of the configuration parameters and operating modes via corresponding dialogue windows. The specified commands can be entered in plain text or selected from the Commands menu. CANopen state machines can be conveniently operated by means of menu selection. The current states are automatically displayed in the status line.

The command tables given in this chapter contain the syntax for direct entry in the Motion Manager. The complete command telegrams are described in Chapter 8 “Parameter description”.

NOTE

Please note that FAULHABER commands can only be received in "Operational" state (Motion Manager "CAN" menu – "Network Management (NMT)" – "Start Remote Node").

3 Operation in FAULHABER mode

Overview of the operating modes in FAULHABER mode and the FAULHABER commands for changing the operating mode

Command	Argument	Function	Description
SOR	0 – 4	Source for Velocity	Source for velocity presetting 0: CAN interface (default) 1: Voltage at analog input 2: PWM signal at analog input 3: Current target value via analog input 4: Current target value via analog input with presetting of the direction of rotation via input polarity
CONTMOD	-	Continuous Mode	Switch back to normal mode from an enhanced mode .
STEPMOD	-	Stepper Motor Mode	Change to stepper motor mode
APCMOD	-	Analog Position Control Mode	Change to position control via analog voltage
ENCMOD	-	Encoder Mode	Change to encoder mode An external encoder serves as position detector (the current position value is set to 0)
HALLSPEED	-	Hall sensor as speed sensor	Speed via Hall sensors in encoder mode
ENCSPEED	-	Encoder as speed sensor	Speed via encoder signals in encoder mode
GEARMOD	-	Gearing Mode	Change to gearing mode
VOLTMOD	-	Set Voltage Mode	Activate Voltage Regulator Mode

3 Operation in FAULHABER mode

3.1 Position control

Guide

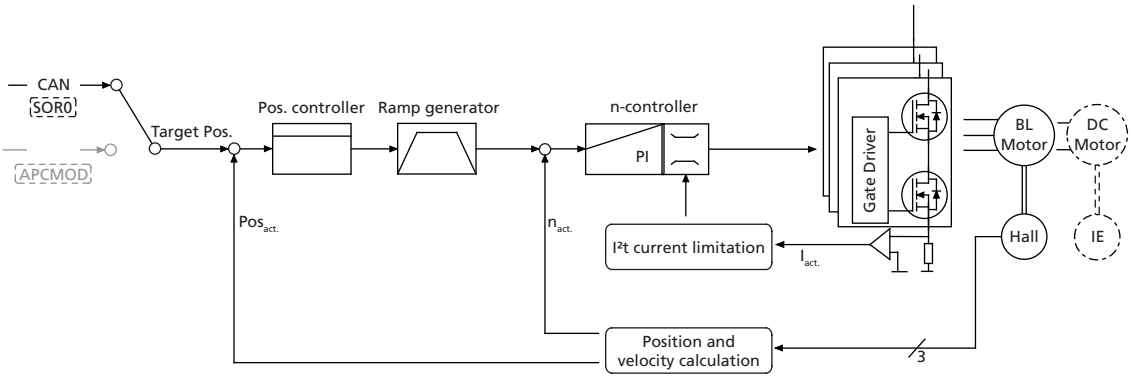
Positioning mode with set value presetting via CAN
Set-point presetting via CAN/PDO2 Page 19

Positioning mode with set value presetting via the analog input
Analog positioning mode (APCMOD) Page 22

Positioning mode with external encoder as actual value
External encoder as actual position value (ENCMOD) - not for MCDC Page 24

3.1.1 Set-point presetting via CAN / PDO2

Controller structure for set-point presetting via CAN / PDO2



In this operating mode, target positions can be preset using the FAULHABER commands via PDO2:

Basic settings

CONTMOD and SOR0 operating mode.

The positioning range limits can be set via the command LL and activated via APL.

The proportional amplification PP and a differential term PD can be set for the position controller.

3 Operation in FAULHABER mode

3.1 Position control

Command	Argument	Function	Description
PP	Value	Load Position Proportional Term	Load position controller amplification. **Value: 1 … 255**
PD	Value	Load Position Differential Term	Load position controller D-term. **Value: 1 … 255**
LL	Value	Load Position Range Limits	Load limit positions (the drive cannot be moved out of these limits). Positive values specify the upper limit and negative values the lower. The range limits are only active if APL1 is. **Value: -1.8 · 10^9 … +1.8 · 10^9**
APL	0-1	Activate/Deactivate Position Limits	Activate range limits (LL) (valid for all operating modes except VOLTMOD). 1: Position limits activated 0: Position limits deactivated

Additional settings

Ramp generator

The slopes of the acceleration and deceleration ramps, and the maximum speed can be defined using the AC, DEC and SP commands (see Chapter 3.6.1 “Ramp generator”).

Velocity controller / current limitation

The controller parameters POR and I of the velocity controller can be adjusted. In addition, the current limitation values LPC and LCC can be used to protect the drive against overload (see Chapter 3.2 “Velocity control”).

Motion control commands

The positioning is executed via the FAULHABER Motion Control commands. An overview of all Motion Control commands is given in Chapter 8 “Parameter description”.

Command	Argument	Function	Description
EN	-	Enable Drive	Activate drive
DI	-	Disable Drive	Deactivate drive
LA	Value	Load Absolute Position	Load new absolute target position **Value: -1.8 · 10^9 … 1.8 · 10^9**
LR	Value	Load Relative Position	Load new relative target position, in relation to last started target position. The resulting absolute target position must lie between the values given below. **Value: -2.14 · 10^9 and 2.14 · 10^9**
M	-	Initiate Motion	Activate position control and start positioning
HO	-/value	Define Home Position	Without argument: Set actual position to 0. With argument: Set actual position to specified value. **Value: -1.8 · 10^9 … +1.8 · 10^9**

Example:

- Load target position: `LA40000`
- Start positioning: `M`

Attainment of the target position is signalled by bit 10 “Target reached” in the statusword of the drive. If the transmission type for the TxPDO1 is set to 255, the PDO is transmitted asynchronously, triggered by the change in state.

Position resolution

If the linear Hall sensors of the brushless motors are used as position transducers, 3 000 pulses per revolution are supplied.

3 Operation in FAULHABER mode

3.1 Position control

Complex motion profiles

More complex motion profiles can be generated through appropriate presetting of new values (maximum speed, acceleration, end position) during positioning. After a value change, simply execute a new motion start command (M).

Further information on compiling motion profiles is given in Chapter 3.6.1 “Ramp generator”.

Positioning beyond the range limits

In the case of APL0, relative positioning can also be executed beyond the range limits. If the upper (1 800 000 000) or lower (-1 800 000 000) limit is exceeded, counting is continued at 0 without loss of increments.

Digital signal target position

The entry into the target corridor can be displayed via the fault output as a digital output signal in the POSOUT function. The signal is not reset until a further Motion start command (M).

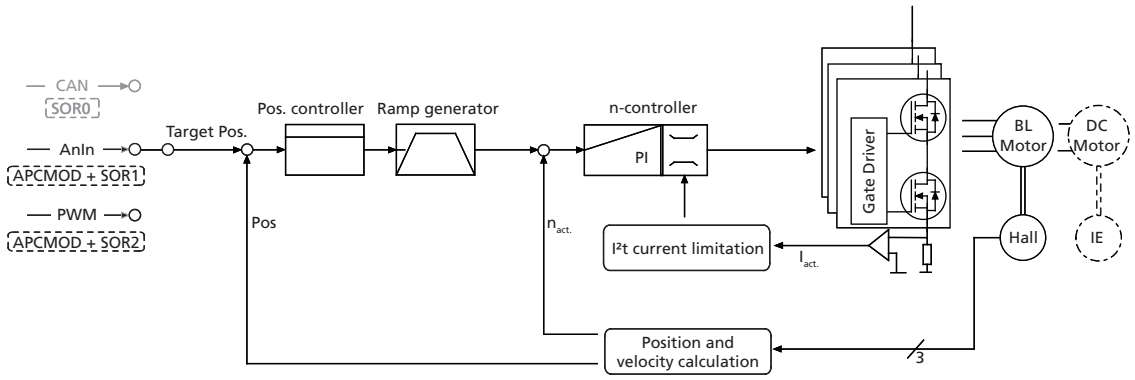
For notes on configuration, see Chapter 3.5 “Special fault output functions”.

3 Operation in FAULHABER mode

3.1 Position control

3.1.2 Analog positioning mode (APCMOD)

Controller structure for set-point presetting via an analog voltage



In this operating mode the target position can be preset using an analog voltage at the AnIn input.

Basic settings

APCMOD mode and SOR1 or SOR2.

The positioning range limits can be set via the command LL and activated via APL.

The proportional amplification PP and a differential term PD can be set for the position controller.

The maximum position to be approached with a voltage of 10 V can be preselected with the LL command. At -10 V the drive moves in the opposite direction up to the set negative range limit.

Irrespective of the preset LL value, the maximum position is limited to 3 000 000 in APCMOD.

Comment: The resolution of the analog input is limited to 12 bit (4 096 steps).

The direction of rotation can be predefined with the commands ADL and ADR.

Command	Argument	Function	Description
PP	Value	Load Position Proportional Term	Load position controller amplification. **Value: 1 … 255**
PD	Value	Load Position Differential Term	Load position controller D-term. **Value: 1 … 255**
LL	Value	Load Position Range Limits	Load limit positions (the drive cannot be moved out of these limits). Positive values specify the upper limit and negative values the lower. The range limits are only active if APL1 is set. **Value: –3 000 000 … 3 000 000 in the APCMOD**
ADL	-	Analog Direction Left	If the set-point is positive the drive rotates to the left (anti-clockwise).
ADR	-	Analog Direction Right	If the set-point is positive the drive rotates to the right (clockwise).

3 Operation in FAULHABER mode

3.1 Position control

Additional settings

Ramp generator

The slopes of the acceleration and deceleration ramps, and the maximum speed can be defined using the AC, DEC and SP commands (see Chapter 3.6.1 “Ramp generator”).

Velocity controller / current limitation

The controller parameters POR and I of the velocity controller can be adjusted. In addition, the current limitation values LPC and LCC can be used to protect the drive against overload (see Chapter 3.2 “Velocity control”).

Positioning via pulse width signal (PWM) at the analog input (SOR2)

If SOR2 is set in APCMOD, the pulse duty factor of a PWM signal can be used as position set-point.

On delivery:

- Pulse duty factor > 50% ➔ positive target position
- Pulse duty factor = 50% ➔ target position = 0
- Pulse duty factor < 50% ➔ negative target position

Absolute positioning within one revolution (only for BL 2 pole):

In motion control systems with brushless 2-pole motors, the initial position is absolutely initialised within one revolution after the motor is switched on (0 - 3 000 corresponds to 0 - 360° of the rotor position). This means that even if the power supply is disconnected, the position determination supplies the correct position value after restarting (if the rotor has only been turned within one revolution).

The following commands enable the drive to be accurately positioned in the voltage range 0 V … 10 V within one revolution and to return to the correct position even after the supply has been switched off, without homing.

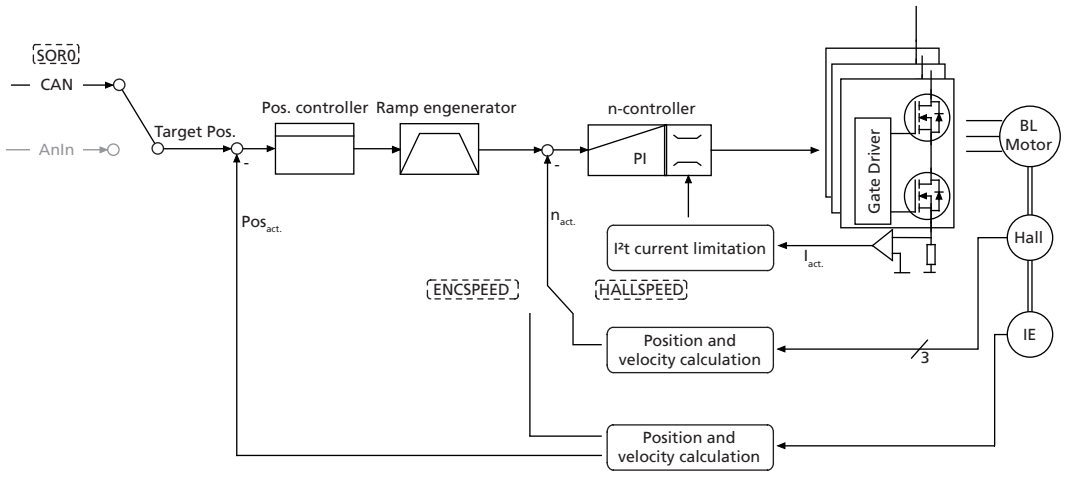
- Switch over to analog positioning: `APCMOD`
- Hide negative range: `LL-1`
- Fix maximum position to 1 revolution: `LL3000`

3 Operation in FAULHABER mode

3.1 Position control

3.1.3 External encoder as actual position value (ENCMOD) - not for MCDC

Controller structure for using an external encoder as the actual value encoder



For high-precision applications, the actual values of BL motors can be derived from an external encoder.

- Depending on the application, the velocity can be derived from the encoder or from the Hall sensors.
- The external encoder can be mounted directly on the motor shaft, but an encoder that is mounted to the application output (e.g. glass scale) is particularly advantageous. This allows the high precision to be set directly at the output.
- Commutation still occurs via the analog Hall sensors.

Basic settings

ENCMOD and SOR0 operating mode.

The positioning range limits can be set via the command LL and activated via APL.

The proportional amplification PP and a differential term PD can be set for the position controller.

Command	Argument	Function	Description
PP	Value	Load Position Proportional Term	Load position controller amplification. **Value: 1 … 255**
PD	Value	Load Position Differential Term	Load position controller D-term. **Value: 1 … 255**
LL	Value	Load Position Range Limits	Load limit positions (the drive cannot be moved out of these limits). Positive values specify the upper limit and negative values the lower. The range limits are only active if APL1 is. **Value: $-1.8 \cdot 10^9$ … $+1.8 \cdot 10^9$**
APL	0 - 1	Activate/Deactivate Position Limits	Activate range limits (LL) (valid for all operating modes except VOLTMOD). 1: Position limits activated 0: Position limits deactivated

3 Operation in FAULHABER mode

3.1 Position control

Settings for external encoder

Command	Argument	Function	Description
ENCMOD	-	Encoder Mode	Change to encoder mode (not for MCDC) An external encoder serves as position transducer (the current position value is set to 0).
ENCSPEED	-	Encoder as speed sensor	Speed via encoder signals in encoder mode
HALLSPEED	-	Hall sensor as speed sensor	Speed via hall sensors in encoder mode
ENCRES	Value	Load Encoder Resolution	Load resolution of external encoder (4 times pulse/rev). **Value: 8 to 65 535**

Additional settings

Ramp generator

The slopes of the acceleration and deceleration ramps, and the maximum speed can be defined using the AC, DEC and SP commands (see Chapter 3.6.1 “Ramp generator”).

Velocity controller / current limitation

The controller parameters POR and I of the velocity controller can be adjusted. In addition, the current limitation values LPC and LCC can be used to protect the drive against overload (see Chapter 3.2 “Velocity control” and Chapter 3.6.3 “Current controller and I²t current limitation”).

Motion control commands

Positioning in the ENCMOD is executed in precisely the same way as in CONTMOD, using the FAULHABER Motion Control commands. An overview of all Motion Control commands is given in Chapter 8 “Parameter description”.

Command	Argument	Function	Description
EN	-	Enable Drive	Activate drive
DI	-	Disable Drive	Deactivate drive
LA	Value	Load Absolute Position	Load new absolute target position **Value: $-1.8 \cdot 10^9$ … $1.8 \cdot 10^9$**
LR	Value	Load Relative Position	Load new relative target position, in relation to last started target position. The resulting absolute target position must lie between the values given below. **Value: $-2.14 \cdot 10^9$ and $2.14 \cdot 10^9$**
M	-	Initiate Motion	Activate position control and start positioning
HO	-/value	Define Home Position	Without argument: Set actual position to 0. With argument: Set actual position to specified value. **Value: $-1.8 \cdot 10^9$ … $+1.8 \cdot 10^9$**

3 Operation in FAULHABER mode

3.1 Position control

Example:

- Load target position: `LA40000`
- Start positioning: `M`

Attainment of the target position is signalled by bit 10 “Target reached” in the statusword of the drive. If the transmission type for the TxPDO1 is set to 255, the PDO is transmitted asynchronously, triggered by the change in state.

Position resolution

In ENCMOD the resolution of the position values depends on the resolution of the encoder.

Complex motion profiles

More complex motion profiles can be generated through appropriate presetting of new values (maximum speed, acceleration, end position) during positioning. After a value change, simply execute a new motion start command (M).

Further information on compiling motion profiles is given in Chapter 3.6.1 “Ramp generator”.

Positioning beyond the range limits

In the case of APL0, relative positioning can also be executed beyond the range limits. If the upper (1 800 000 000) or lower limit (-1 800 000 000) is exceeded, counting is continued at 0 without loss of increments.

Digital signal target position

The entry into the target corridor can be displayed via the fault output as a digital output signal in the POSOUT function. The signal is not reset until a further Motion start command (M).

For notes on configuration, see Chapter 3.5 “Special fault output functions”.

3 Operation in FAULHABER mode

3.2 Velocity control

Guide

Velocity control mode with set value presetting via CAN
Target velocity via CAN/PDO2 Page 28

Velocity control mode with set value presetting via the analog input
Velocity presetting via an analog voltage or a PWM signal Page 30

Velocity control mode with external encoder as actual value
External encoder as actual velocity value (ENCMOD) - not for MCDC Page 32

In velocity control mode the velocity of the drive is controlled by a PI controller. Provided the drive is not overloaded, the drive follows the presetting without deviation.

The current velocity of BL motors can be detected both from the Hall signals and via an additional encoder; an incremental encoder is always required for DC motors. One exception is the IxR control, as described in Chapter 3.4.5 “IxR control for MCDC”.

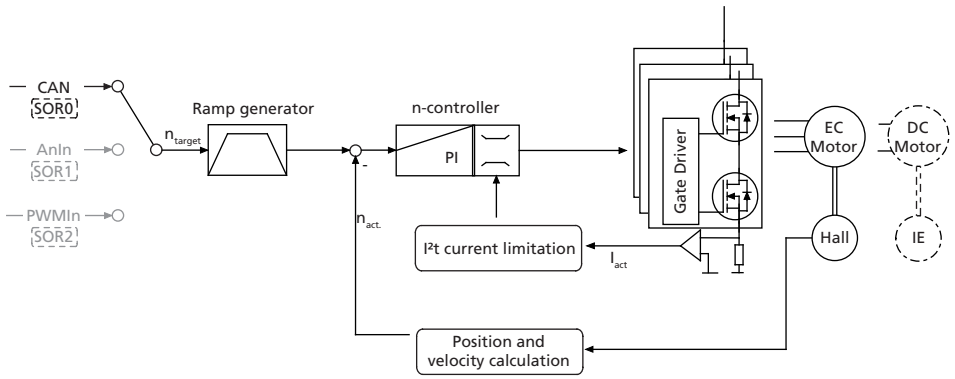
The velocity can be preset via the CAN interface (PDO2), via an analog voltage preset or via a PWM signal.

3 Operation in FAULHABER mode

3.2 Velocity control

3.2.1 Target velocity via CAN / PDO2

Controller structure for velocity control



In this operating mode, the drive velocity can be controlled with set-point presetting via FAULHABER commands on PDO2. The velocity of BL motors is registered by the analog hall sensors, incremental encoders are only supported for DC motors.

Basic settings

CONTMOD and SOR0 operating mode.

The controller parameters POR and I and the sampling rate can be adjusted for the velocity controller.

Command	Argument	Function	Description
POR	Value	Load Velocity Proportional Term	Load velocity controller amplification. **Value: 1 ... 255**
I	Value	Load Velocity Integral Term	Load velocity controller integral term. **Value: 1 ... 255**
SR	Value	Load Sampling Rate	Load sampling rate of the velocity controller as a multiple of the basic sampling time. **Value: 1 ... 20**

Velocity input

In BL motors the current velocity is determined in CONTMOD by evaluating the Hall sensor signals, which supply 3 000 pulses per revolution.

In DC motors the velocity is determined using an incremental encoder whose resolution has to be set using the ENCRES command. DC motors without an incremental encoder can also be operated with limited accuracy in IxR mode (see Chapter 3.4.5 “IxR control for MCDC”).

Command	Argument	Function	Description
ENCRES	Value	Load Encoder Resolution	Load resolution of external encoder (4 times pulse/rev). **Value: 8 to 65 535**

Additional settings

Movement limits

The LL command can also be used to define a movement range limit for velocity mode. The APL1 command activates monitoring of these limits.

Ramp generator

The slopes of the acceleration and deceleration ramps, and the maximum speed can be defined using the AC, DEC and SP commands (see Chapter 3.6.1 “Ramp generator”).

Current limitation

The current limitation values LPC and LCC can be used to protect the drive against overload (see Chapter 3.6.3 “Current controller and I²t current limitation”).

Motion control commands

An overview of all Motion Control commands is given in Chapter 8 “Parameter description”.

Command	Argument	Function	Description
EN	-	Enable Drive	Activate drive
DI	-	Disable Drive	Deactivate drive
V	Value	Select Velocity Mode	Activate velocity mode and set specified value as target velocity (velocity control). Unit: rpm **Value: –30 000 … 30 000**

Example:

- Drive motor at 100 rpm: `V100`

  In order to change the direction of rotation, simply assign a negative velocity value (e.g. `V-100`).
- Stop motor: `V0`

NOTE

Make sure that APL0 is set, if you do not want the drive to stop at the set range limits (LL)!

Also check that the maximum speed SP is not set below the desired target velocity.

Complex motion profiles

Attainment of the target velocity is signalled by bit 10 “Target reached” in the statusword of the drive. If the transmission type for the TxPDO1 is set to 255, the PDO is transmitted asynchronously, triggered by the change in state.

Command	Argument	Function	Description
LL	Value	Load Position Range Limits	Load limit positions (the drive cannot be moved out of these limits). Positive values specify the upper limit and negative values the lower. The range limits are only active if APL1 is. **Value: $-1.8 \cdot 10^9$ … $+1.8 \cdot 10^9$**
APL	0 - 1	Activate/Deactivate Position Limits	Activate range limits (LL) (valid for all operating modes except VOLTMOD). 1: Position limits activated 0: Position limits deactivated

3.2.2 Velocity presetting via an analog voltage or a PWM signal

In this operating mode, the drive velocity can be controlled with set value presetting via an analog voltage or a PWM signal.

Basic settings

CONTMOD mode and SOR1 (AnIn) or SOR2 (PWMIn).

The controller parameters POR, I and the sampling rate can be adjusted for the velocity controller. In addition, commands are available for configuring the analog velocity presetting.

Command	Argument	Function	Description
SP	Value	Load Maximum Speed	Load maximum speed (here: Target velocity at 10 V). Setting applies to all modes (except VOLTMOD). Unit: rpm **Value: 0 … 30 000**
MV	Value	Minimum Velocity	Specifies the lowest velocity. Unit: rpm **Value: 0 … 30 000**
MAV	Value	Minimum Analog Voltage	Specifies the minimum start voltage. Unit: mV **Value: 0 … 10 000**
ADL	-	Analog Direction Left	Positive voltages at the analog input result in counter-clockwise rotation of the rotor
ADR	-	Analog Direction Right	Positive voltages at the analog input result in clockwise rotation of the rotor.
DIRIN	-	Direction Input	Use fault pin as rotational direction input. Low: … anticlockwise rotation (corresponding to ADL command) High: … clockwise rotation (corresponding to ADR command)
POR	Value	Load Velocity Proportional Term	Load velocity controller amplification. **Value: 1 … 255**
I	Value	Load Velocity Integral Term	Load velocity controller integral term. **Value: 1 … 255**
SR	Value	Load Sampling Rate	Load sampling rate of the velocity controller as a multiple of the basic sampling time. **Value: 1 … 20**

Velocity input

By default, in BL motors the current speed is determined by evaluating the Hall sensor signals. Additional incremental encoders cannot be connected to BL motors for analog velocity presetting

In DC motors the velocity is solely determined using the incremental encoder. DC motors without an incremental encoder can also be operated with limited accuracy in IxR mode (see Chapter 3.4.5 “IxR control for MCDC”).

3 Operation in FAULHABER mode

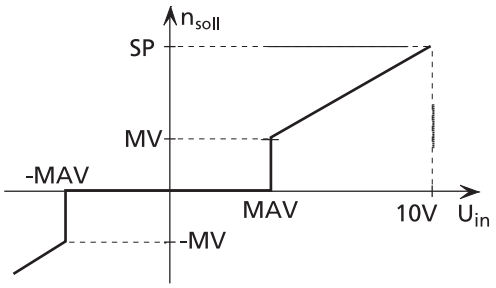
3.2 Velocity control

Target value input

Example:

The drive is only to start moving with voltages over 100 mV or below -100 mV at the analog input:

- MAV100



Advantage:

As 0 mV is usually difficult to set at the analog input, 0 rpm is also not easy to implement. The dead band produced by the minimum start voltage prevents the motor from starting as a result of small interference voltages.

Additional settings

Movement limits

The LL command can also be used to define a movement range limit for velocity mode. The APL1 command activates monitoring of these limits.

Ramp generator

The slopes of the acceleration and deceleration ramps, and the maximum speed can be defined using the AC, DEC and SP commands (see Chapter 3.6.1 “Ramp generator”).

Current limitation

The current limitation values LPC and LCC can be used to protect the drive against overload (see Chapter 3.6.3 “Current controller and I²t current limitation”).

Velocity control using pulse width modulated (PWM) signal at the analog input (SOR2)

If SOR2 is set in CONTMOD, the pulse duty factor of a PWM signal can be used as velocity target.

On delivery:

- Pulse duty factor > 50% ➔ clockwise rotation
- Pulse duty factor = 50% ➔ stoppage n = 0
- Pulse duty factor < 50% ➔ anti-clockwise rotation

The commands SP, MV, MAV, ADL and ADR can also be used here.

NOTE

Make sure that APL0 is set, if you do not want the drive to stop at the set range limits (LL)!

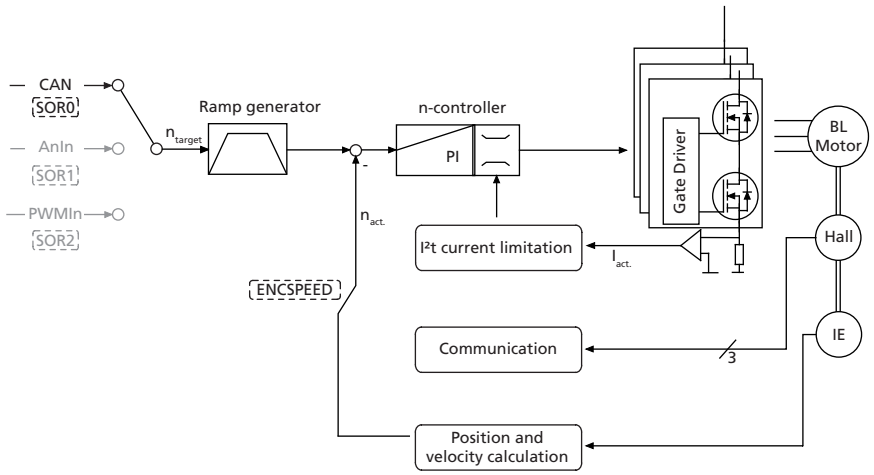
Also check that the maximum speed SP is not set below the desired target velocity.

Input circuit

The input circuit at the analog input is designed as a differential amplifier. If the analog input is open, an undefined velocity can be set. The input must be connected to AGND with low-impedance or set to the voltage level of the AGND, in order to generate 0 rpm. For a protective circuit example, see **Chapter 3.4 in the technical manual.**

3.2.3 External encoder as actual velocity value (ENCMOD) - not for MCDC

Velocity control with external encoder as actual value



In this operating mode, the drive velocity can be controlled with set-point presetting via FAULHABER commands on PDO2. The velocity is evaluated via an additional encoder, external or built onto the motor. In particular, this enables a specific load speed to be controlled by an incremental encoder at the output.

ENCMOD mode is available for BL motors only. The analog Hall sensors of the motors are also evaluated in ENCMOD mode for the motor commutation.

Basic settings

ENCMOD and SOR0 operating mode.

The controller parameters POR and I and the sampling rate can be adjusted for the velocity controller.

Command	Argument	Function	Description
POR	Value	Load Velocity Proportional Term	Load velocity controller amplification. **Value: 1 ... 255**
I	Value	Load Velocity Integral Term	Load velocity controller integral term. **Value: 1 ... 255**
SR	Value	Load Sampling Rate	Load sampling rate of the velocity controller as a multiple of the basic sampling time. **Value: 1 ... 20**

3 Operation in FAULHABER mode

3.2 Velocity control

Velocity input

The external incremental encoder's resolution must be specified with 4 edge evaluation using the ENCRES parameter.

In addition to ENCMOD mode, velocity evaluation on the basis of the encoder must be activated using the ENCSPEED command.

Command	Argument	Function	Description
ENCRES	Value	Load Encoder Resolution	Load resolution of external encoder (4 times pulse/rev). **Value: 8 to 65 535**
ENCMOD	-	Encoder Mode	Change to encoder mode (not for MCDC) An external encoder serves as position detector (the current position value is set to 0)
ENCSPEED	-	Encoder as speed sensor	Speed via encoder signals in encoder mode
HALLSPEED	-	Hall sensor as speed sensor	Speed via hall sensors in encoder mode

Additional settings

Movement limits

The LL command can also be used to define a movement range limit for velocity mode. The APL1 command activates monitoring of these limits.

Ramp generator

The slopes of the acceleration and deceleration ramps, and the maximum speed can be defined using the AC, DEC and SP commands (see Chapter 3.6.1 "Ramp generator").

Current limitation

The current limitation values LPC and LCC can be used to protect the drive against overload (see Chapter 3.6.3 "Current controller and I²t current limitation").

Motion control commands

An overview of all Motion Control commands is given in Chapter 8 "Parameter description".

Command	Argument	Function	Description
EN	-	Enable Drive	Activate drive
DI	-	Disable Drive	Deactivate drive
V	Value	Select Velocity Mode	Activate velocity mode and set specified value as target velocity (velocity control). Unit: rpm **Value: –30 000 … 30 000**

Example:

- Drive motor at 100 rpm: `V100`

  In order to change the direction of rotation, simply assign a negative velocity value (e.g. `V-100`).
- Stop motor: `V0`

NOTE

Make sure that APL0 is set, if you do not want the drive to stop at the set range limits (LL)!

Also check that the maximum speed SP is not set below the desired target velocity.

Complex motion profiles

Attainment of the target velocity is signalled by bit 10 “Target reached” in the statusword of the drive. If the transmission type for the TxPDO1 is set to 255, the PDO is transmitted asynchronously, triggered by the change in state.

Command	Argument	Function	Description
LL	Value	Load Position Range Limits	Load limit positions (the drive cannot be moved out of these limits). Positive values specify the upper limit and negative values the lower. The range limits are only active if APL is 1. **Value: $-1.8 \cdot 10^9$ … $+1.8 \cdot 10^9$**
APL	0 - 1	Activate/Deactivate Position Limits	Activate range limits (LL) (valid for all operating modes except VOLTMOD). 1: Position limits activated 0: Position limits deactivated

3 Operation in FAULHABER mode

3.3 Homing and limit switches

Guide

Overview of the connections of the Faulhaber Motion Control systems available for limit switches and their configuration
Limit switch connections and switching level Page 36

Motion control commands (trigger homing sequence)
Motion control commands Page 37

Configuration of the behaviour at the limit switch and the homing sequence
Configuration of homing and limit switches Page 38

Homing on limit switches can be used to re-initialise the absolute position of an application after switching on.

The GOHOSEQ command is used to perform previously defined homing up to the set limit switch and then perform the actions defined for it. The ramp generator settings for maximum acceleration and the movement limits are taken into account.

3 Operation in FAULHABER mode

3.3 Homing and limit switches

3.3.1 Limit switch connections and switching level

The connections

- AnIn
- Fault
- 3rd input
- 4th, 5th input (MCDC only)

can be used as reference and limit switch inputs.

In addition, the zero crossing of the Hall sensor signals is also available as an index pulse in BL motors. Depending on the motor type (two-pole or four-pole), the index pulse occurs once or twice per revolution. The index pulse of an external encoder can also be connected to the fault pin, enabling the actual position to be exactly zeroed.

The AnIn and Fault connections are designed as interrupt inputs, which means that they are edge-triggered. All other inputs are not edge-triggered, so that the signal must be applied for at least 500 µs to enable it to be reliably detected. The maximum response time to level changes at all inputs is 500 µs.

Digital input configuration

Command	Argument	Function	Description
SETPLC	-	Set PLC inputs	Digital inputs PLC-compatible (24 V level) (For level definition, see technical manual)
SETTTL	-	Set TTL inputs	Digital inputs TTL-compatible (5 V level) (For level definition, see technical manual)
REFIN	-	Reference Input	Fault pin as reference or limit switch input

The limit switch functions for the fault pin are only accepted if REFIN is activated (setting must be saved with SAVE)!

CAUTION!

Configure before applying a voltage

The electronics can be damaged if a voltage is applied to the fault pin while it is not configured as input.

► *Configure the fault pin as input first before applying external voltage!*

3.3.2 Motion control commands

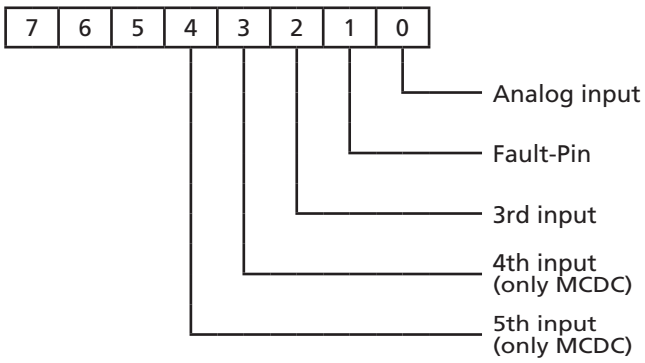
The function of the inputs and the homing behaviour are set using the FAULHABER commands described in Chapter 3.3.3 “Configuration of homing and limit switches”. A previously configured homing is then started with the following FAULHABER commands. An overview of all motion control commands is given in Chapter 7.4 “Motion control commands”.

Command	Argument	Function	Description
GOHOSEQ	-	Go Homing Sequence	Execute FAULHABER homing sequence. A homing sequence is executed (if programmed) irrespective of the current mode.
FHIX	-	Find Hall Index	The nearest index pulse in the preset direction of rotation is approached. For BX4 drives only
GOHIX	-	Go Hall Index	Move BL motor to Hall zero point (Hall index) and set actual position value to 0. Not for BX4 and MCDC drives
GOIX	-	Go Encoder Index	Move to the encoder index at the Fault pin and set actual position value to 0 (DC motor or ext. encoder).

If the drive is already located in the limit switch when GOHOSEQ is invoked, first of all it moves out of the switch, in the opposite direction to that specified for HOSP.

3.3.3 Configuration of homing and limit switches

The following commands use the following bit mask for configuration of the limit switch functions:



Set or delete the bit at the position of the required input for each command and assign the resulting numeric value to the commands described below.

Polarity and limit switch function

Limit switches can respond to the rising or falling edge (or level).

In addition, the hard blocking function can be configured for the limit switches. The hard blocking function provides reliable protection against overshooting of the range limit switch. If the drive is located in an HB limit switch, then the direction of rotation set with HD will be blocked, i.e. the drive can only move further out of the limit switch.

The speed stays at 0 rpm if target velocity is preset in the wrong direction.

Command	Argument	Function	Description
HP	Bit mask	Hard Polarity	Define valid edge and polarity of respective limit switches: 1: Rising edge and high level effective. 0: Falling edge and low level effective.
HB	Bit mask	Hard Blocking	Activate Hard Blocking function for relevant limit switch.
HD	Bit mask	Hard Direction	Presetting of direction of rotation that is blocked with HB of respective limit switch. 1: Clockwise rotation blocked 0: Anticlockwise rotation blocked

Example:

- Setting the hard blocking function for fault pin and 4th input: $2^1+2^3 = 2+8 = 10$ ➔ `HB10`

Definition of homing behaviour

In order to be able to execute a homing sequence with the command GOHOSEQ, a homing sequence must be defined for a specific limit switch! To do this, at least one of the following actions must be defined for the limit switch. Definition of the hard blocking behaviour is an additional option.

Command	Argument	Function	Description
SHA	Bit mask	Set Home Arming for Homing Sequence	Homing behaviour (GOHOSEQ): Set position value to 0 at edge of respective limit switch
SHL	Bit mask	Set Hard Limit for Homing Sequence	Homing behaviour (GOHOSEQ): Stop motor at edge of respective limit switch.
SHN	Bit mask	Set Hard Notify for Homing Sequence	Homing behaviour (GOHOSEQ): If an edge is at the respective limit switch, the hard notify bit is set in the statusword of the drive.

These settings must be saved with SAVE so that they are available immediately after switching on!

Example:

- Homing with 3rd input as reference input (rising edge):
  - **HP4** Low level or falling edge was evaluated at AnIn and at the fault pin, the rising edge is evaluated at the 3rd input.
  - **SHA4** Activate a homing sequence for 3rd input (all others are in bit mask = 0) Action: Set Pos = 0 on reaching the limit switch
  - **SHL4** Activate a homing sequence for 3rd input (all others are in bit mask = 0) Action: Stop motor
  - **SHN4** Activate a homing sequence for 3rd input (all others are in bit mask = 0) Action: Notify in statusword of the drive or via TxPD01

Homing Speed

Command	Argument	Function	Description
HOSP	Value	Load Homing Speed	Load speed and direction of rotation for homing (GOHOSEQ, GOHIX). Unit: rpm

Example:

- Homing with 100 rpm and negative direction of rotation:

```
HOSP-100
```

Direct programming via HA, HL and HN commands

These special commands can be used to define actions that are to be triggered at an edge of the relevant input, independently of a homing sequence. A programmed limit switch function will remain effective until the preselected edge occurs. The programming can be changed with a new command before an edge occurs.

Command	Argument	Function	Description
HA	Bit mask	Home Arming	Set position value to 0 and delete relevant HA bit at edge of respective limit switch. Setting is not saved
HL	Bit mask	Hard Limit	Stop motor and delete relevant HL bit at edge of respective limit switch. Setting is not saved.
HN	Bit mask	Hard Notify	If an edge is at the respective limit switch, the hard notify bit is set in the statusword of the drive. Setting is not saved.

The settings are not saved with the SAVE command, therefore all configured limit switches are inactive again after power-on.

HL / SHL command:

- **Positioning mode**

  When the edge occurs, the motor positions itself on the reference mark with maximum acceleration.

- **Velocity controller mode**

  The motor is decelerated at the set acceleration value when the edge occurs, i.e. it goes beyond the reference mark. The reference mark can be precisely approached with a subsequent positioning command (command M).

  Advantage: No abrupt motion changes.

3 Operation in FAULHABER mode

3.4 Extended operating modes

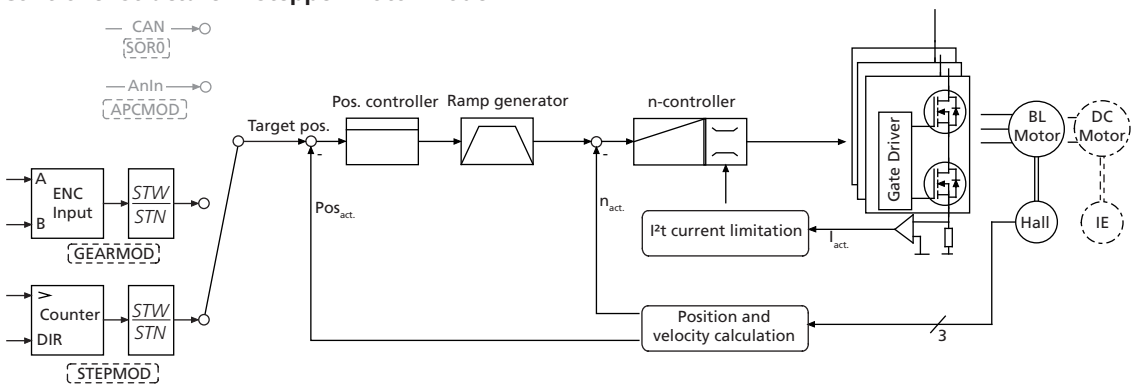
Guide

Stepper motor mode Page 40
Gearing mode (electronic gear) Page 42
Voltage regulator mode Page 44
Current control with analog current presetting Page 45
IxR control for MCDC Page 47

Use the CONTMOD command to revert from an enhanced operating mode to normal mode.

3.4.1 Stepper motor mode

Controller structure in stepper motor mode



In stepper motor mode the drive moves one programmable angle further for each pulse at the analog input, and thus simulates the function of a stepper motor.

There are a number of considerable advantages in comparison with a real stepper motor:

- The number of steps per revolution is freely programmable and of a very high resolution (encoder resolution)
- The individual step widths are freely programmable
- No detent torque
- The full dynamics of the motor can be used
- The motor is very quiet
- The motor monitors actual position so that no steps are “lost” (even with maximum dynamics)
- No motor current flows in settled state (actual position reached)
- High efficiency

3 Operation in FAULHABER mode

3.4 Extended operating modes

Basic settings

In stepper motor mode, the analog input acts as frequency input. The error output must be configured as rotational direction input if the direction of rotation is to be changed via a digital signal.

Alternatively, the direction of rotation can also be preset via the commands ADL and ADR.

Command	Argument	Function	Description
STEPMOD	-	Stepper Motor Mode	Change to stepper motor mode
DIRIN	-	Direction Input	Fault pin as rotational direction input
ADL	-	Analog Direction Left	Positive voltages at the analog input result in anticlockwise rotation of the rotor
ADR	-	Analog Direction Right	Positive voltages at the analog input result in clockwise rotation of the rotor

Input

Maximum input frequency: see technical manual.

Level: 5 V TTL or 24 V PLC-compatible, depending on configuration.

The number of steps of the emulated stepper motor can be set to virtually any required settings using the following formula:

$$\text{Revolutions} = \text{pulses} \cdot \frac{\text{STW}}{\text{STN}}$$

Motor revolutions … revolutions generated on the drive
Pulses … number of pulses at the frequency input (= number of steps)

Command	Argument	Function	Description
STW	Value	Load Step Width	Load step width for step motor and gearing mode **Value: 1 … 65 535**
STN	Value	Load Step Number	Load number of steps per revolution for step motor and gearing mode **Value: 1 … 65 535**

Example:

The motor should move by 1/1 000th of a revolution per input signal:

- `STW1`
- `STN1000`

Additional settings

Movement limits

The LL command can be used to define a limit for the movement range for stepper motor mode also. The APL1 command activates monitoring of these limits.

Ramp generator

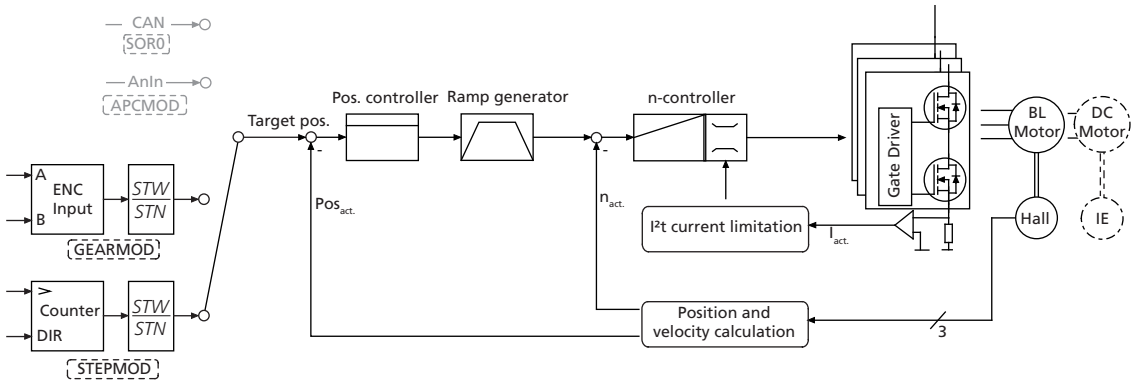
The slopes of the acceleration and deceleration ramps, and the maximum speed can be defined using the AC, DEC and SP commands (see Chapter 3.6.1 “Ramp generator”).

Current limitation

The current limitation values LPC and LCC can be used to protect the drive against overload (see Chapter 3.6.3 “Current controller and I²t current limitation”).

3.4.2 Gearing mode (electronic gear)

Controller structure in gearing mode



Gearing mode enables the use of an external encoder as set-point source for the position. This enables several drives to be synchronised. Several drives can be synchronised in this way. If the direction of rotation is to be changed by a digital signal, the function of the fault pin must be reconfigured as a rotational direction input.

Alternatively, the direction of rotation can also be preset via the commands ADL and ADR.

Basic settings

Command	Argument	Function	Description
GEARMOD	-	Gearing Mode	Change to gearing mode
DIRIN	-	Direction Input	Fault pin as rotational direction input

Input

The two channels of an external encoder are connected to connections AnIn and AGND, which may need to be connected to the 5 V encoder supply via a 2.7 kΩ pull-up resistor.

The gear ratio between the pulses per revolution (PPR) count of the external encoder and the resulting movement of the motor can be set using the following formula:

$$\text{Revolutions} = \text{pulses} \cdot \frac{\text{STW}}{\text{STN}}$$

Motor revolutions … revolutions generated on the drive
Pulses … actually counted pulses during four edge evaluation

Command	Argument	Function	Description
STW	Value	Load Step Width	Load step width for step motor and gearing mode **Value: 1 … 65 535**
STN	Value	Load Step Number	Load number of steps per revolution for step motor and gearing mode **Value: 1 … 65 535**

3 Operation in FAULHABER mode

3.4 Extended operating modes

Example:

Motor has to move one revolution at 1 000 pulses of the external encoder:

- **STW1**
- **STN1000**

Additional settings

Movement limits

The range limits set with LL are also active in gearing mode with APL1.

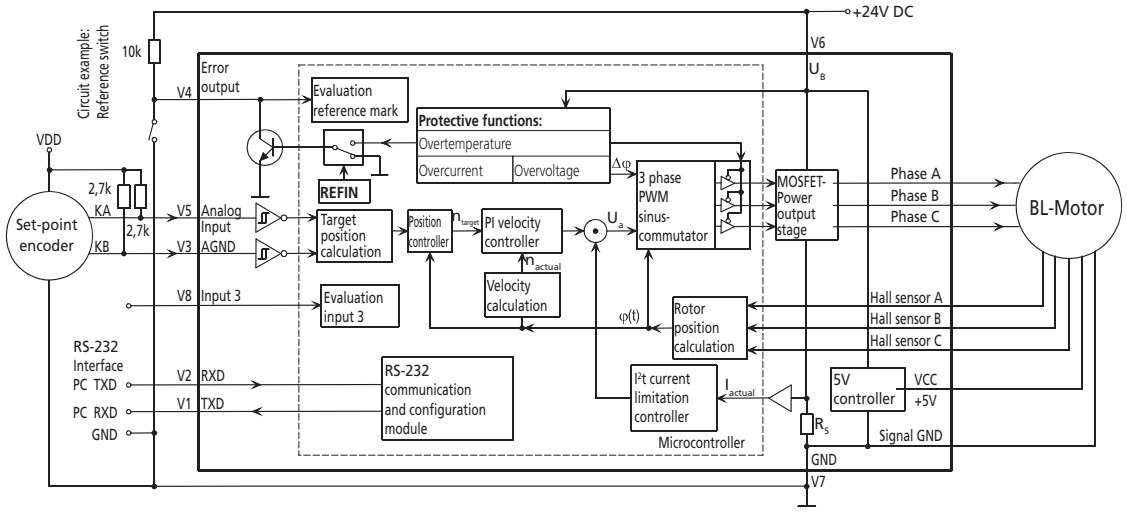
Ramp generator

The slopes of the acceleration and deceleration ramps, and the maximum speed can be defined using the AC, DEC and SP commands (see Chapter 3.6.1 “Ramp generator”).

Current limitation

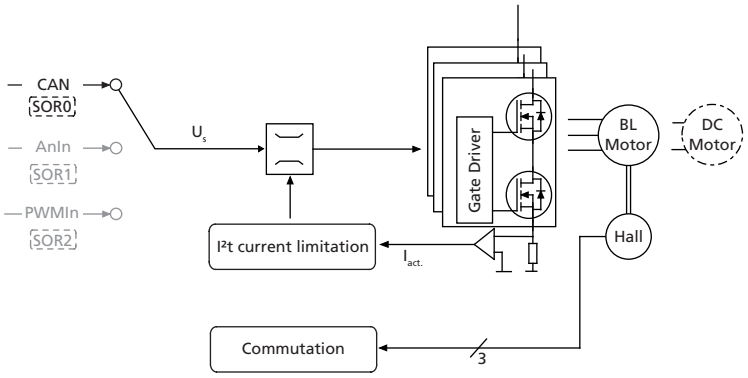
The current limitation values LPC and LCC can be used to protect the drive against overload (see Chapter 3.6.3 “Current controller and I²t current limitation”).

Circuit example, gearing mode for MCBL 3003/06 C



3.4.3 Voltage regulator mode

Controller structure in voltage regulator mode



In voltage regulator mode a motor voltage is output proportional to the preset value. Current limitation remains active.

With this mode, it is possible to use a higher level controller. The controller then serves only as a power amplifier.

Basic settings

Command	Argument	Function	Description
VOLTMOD	-	Set Voltage Mode	Activate Voltage Regulator Mode
U	Value	Set Output Voltage	Output motor voltage (corresponds to -U_B … +U_B) at SOR0 only **Value: -32 767 … 32 767**

Input

SOR0 (CAN / PDO2)	SOR1 (AnIn)	SOR2 (PWMIn)	U_{MOT}
U-32767	-10 V	0 %	-U_B
U0	0 V	50 %	0
U32767	10 V	100 %	+U_B

Additional settings

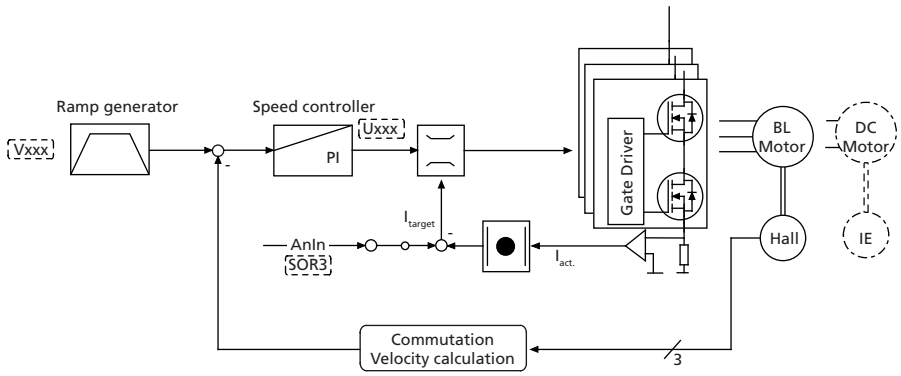
Current limitation

The current limitation values LPC and LCC can be used to protect the drive against overload.

3.4.4 Current control with analog current presetting

Fixed direction of rotation (SOR3)

Controller structure for analog current presetting with fixed preset direction of rotation



You can switch to analog target current presetting with the SOR3 command. In this way, both in velocity mode and in voltage regulator mode, current absolute value can be limited proportional to the voltage at the analog input. The set current is weighted with the maximum current LPC.

The motor is activated either in velocity mode by a previously fixed target velocity, or in voltage regulator mode via a voltage value. The error output must be configured as rotational direction input if the direction of rotation is to be changed via a digital signal.

Basic settings

Command	Argument	Function	Description
SOR	3	Source for Velocity	3:Current target value via analog input
LPC	Value	Load Peak Current Limit	Load peak current (mA). **Value: 0 … 12 000**

Input

If 10 V are present at the analog input, the current is accordingly limited to the maximum current set with LPC.

Even if negative voltages are present at the analog input, the current is limited to the absolute value of the applied voltage. Negative target current presettings therefore have no effect on the direction of rotation!

SOR3 (AnIn)	I_{max}	n_{max}
-10 V	LPC	SP
0 V	0	SP
10 V	LPC	SP

Warning!

Risk of destruction

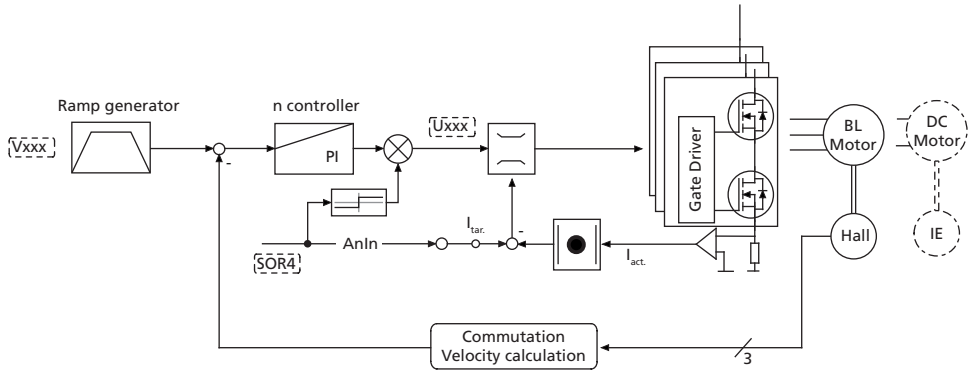
In current control mode with analog current presetting the internal I²t current limitation is deactivated.

3 Operation in FAULHABER mode

3.4 Extended operating modes

Direction of rotation depending on current target value (SOR4)

Controller structure for analog current presetting with variable direction of rotation



You can switch to analog target current presetting with the SOR4 command. In this way, both in velocity mode and in voltage regulator mode, current absolute value can be limited proportional to the voltage at the analog input. The set current is weighted with the maximum current LPC.

The motor is activated either in velocity mode by a previously fixed target velocity, or in voltage regulator mode via a voltage value. The direction of rotation is determined from the sign of the current target value.

This mode corresponds to direct current control.

Basic settings

Command	Argument	Function	Description
SOR	4	Source for Velocity	4: Target current value via analog input with presetting of the direction of rotation via the sign of the set-point.
LPC	Value	Load Peak Current Limit	Load peak current (mA). **Value: 0 … 12 000**

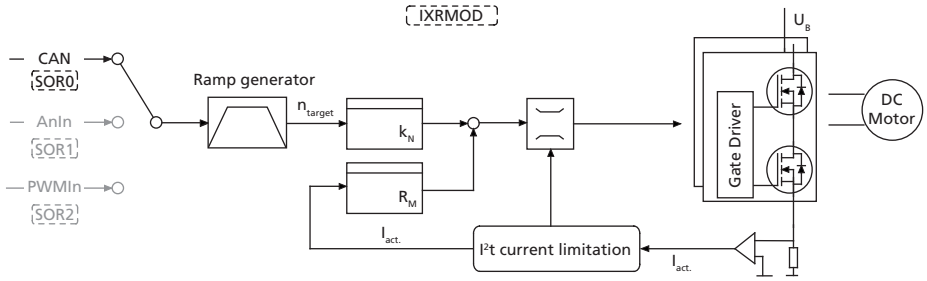
Input

If 10 V are present at the analog input, the current is accordingly limited to the maximum current set with LPC.

SOR4 (AnIn)	I_{max}	n_{max}
-10 V	LPC	-SP
0 V	0	SP
10 V	LPC	SP

3.4.5 IxR control for MCDC

Controller structure in IxR mode



For speed-controlled applications with DC motors without an encoder, an IxR control is available on the MCDC. In this mode, the motor speed is determined via an internal motor model. Consequently, the encoder and the associated wiring can be omitted.

However, control quality and accuracy are considerably restricted. This mode is mainly suited for higher speeds and larger motors in the FAULHABER range.

Basic settings

Command	Argument	Function	Description
IXRMOD	-	Set IxR Mode	Activate IxR control (MCDC only)
RM	Value	Load Motor Resistance	Load motor resistance RM according to specification in data sheet. Unit: mΩ
KN	Value	Load Speed Constant	Load speed constant kn in accordance with information in the data sheet. Unit: rpm/V

In stationary mode the following formula applies to the voltage at the DC motor: $U_M = R_M \times I_A + k_N \times n$.

As a result, at constant terminal voltage U_M the speed falls under load.

Vice versa, if R_M and k_N are known, the voltage applied to the motor can be increased depending on the target velocity and the measured motor current so that the voltage drop is approximately compensated at the winding resistor.

Setting rules

Synchronisation of the no-load speed via k_N.

Synchronisation of the velocity under load via R_M.

- Velocity increases under load: R_M is set too high
- Velocity drops too far under load: R_M is set too low

3.5 Special fault output functions

The error connection (fault pin) can be configured as input or output for different tasks:

Command	Function	Description
ERROUT	Error Output	Fault pin as error output (default)
ENCOUT	Encoder Output	Fault pin as pulse output (not MCDC)
DIGOUT	Digital Output	Fault pin as digital output. The output is set to low level.
DIRIN	Direction Input	Fault pin as rotational direction input ▶ Velocity control (see Chapter 3.2 “Velocity control”), ▶ Stepper motor mode (see Chapter 3.4.1 “Stepper motor mode”), ▶ Gearing mode (see Chapter 3.4.2 “Gearing mode (electronic gear)”), ▶ Voltage regulator mode (see Chapter 3.4.3 “Voltage regulator mode”). ▶ Current control with analog current presetting (see Chapter 3.4.4 “Current control with analog current presetting”).
REFIN	Reference Input	Fault pin as reference or limit switch input ▶ Homing and limit switches (see Chapter 3.3 “Homing and limit switches”)
POSOUT	Position Output	Fault pin as output for display of the condition: “target position reached".

Fault pin as error output

In ERROUT mode the output is set as soon as one of the following errors occurs:

- One of the set current limitation values (LPC, LCC) is exceeded
- Set maximum permissible speed deviation (DEV) is exceeded
- Overvoltage detected
- Maximum coil or MOSFET temperature exceeded

Additional settings

Delayed signalling

In order to hide the transient occurrence of errors during the acceleration phase, for example, an error delay can be set which specifies how long an error must be present before it is displayed at the error output:

Command	Argument	Function	Description
DCE	Value	Delayed Current Error	Delayed error output with ERROUT Value in 1/100 sec.

Example:

Wait 2 seconds before displaying error:

- `DCE200`

If one of the errors above occurs, a corresponding Emergency Object is sent to the CAN network, provided the Emergency mask in Object 0x2320 for the corresponding error is set to 1. See also Chapter 8.2 “Manufacturer-specific objects” under “FAULHABER Fault Register”.

3 Operation in FAULHABER mode

3.5 Special fault output functions

Fault pin as pulse output (not for MCDC):

In the ENCOUT mode the fault pin is used as pulse output, which outputs an adjustable number of pulses per revolution. The pulses are derived from the Hall sensor signals of the BL motors and are limited to max. 4 000 pulses per second in 2 pole motors, and to max. 2 000 pulses per second in 4 pole motors.

In MCBL 300x CF AES the LPN value is limited to 32.

Command	Argument	Function	Description
LPN	Value	Load Pulse Number	Preset pulse number for ENCOUT. **Value: 1 to 255** **Value: 1 bis 32 bei MCBL AES**

Example:

Output 16 pulses per revolution at the fault pin:

- `LPN16`

In the case of 5 000 rpm, 5 000/60 · 16 = 1 333 pulses per second are output.

NOTE

Bei Drehzahlen, die bei eingestelltem LPN-Wert mehr als die maximal mögliche Impulszahl erzeugen würden, wird die maximale Anzahl ausgegeben. The set pulses are precisely achieved, but the timing does not necessarily have to exactly agree (delays possible).

Position determination via pulse counting is therefore possible, provided that no change occurs in the direction of rotation and the maximum possible pulse number is not exceeded.

Fault pin as digital output

In DIGOUT mode, the fault pin can be used as universal digital output. The digital output can be set or cleared using the following commands:

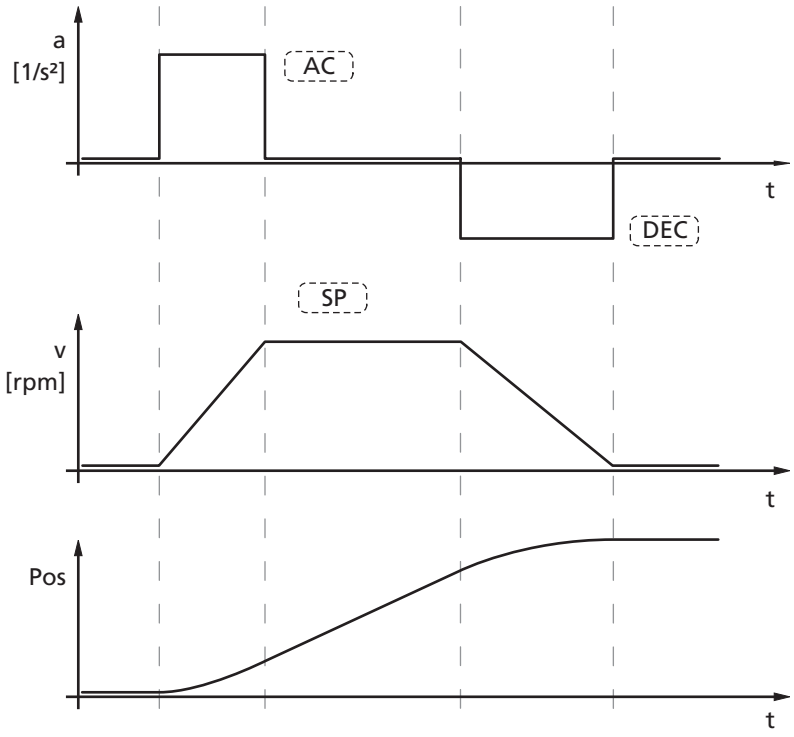
Command	Argument	Function	Description
CO	-	Clear Output	Set digital output DIGOUT to low level
SO	-	Set Output	Set digital output DIGOUT to high level
TO	-	Toggle Output	Toggle digital output DIGOUT

3.6 Technical information

3.6.1 Ramp generator

In all modes, apart from voltage regulator mode and current control, the set-point is controlled by the ramp generator.

Basic ramp generator function



This can be used to separately set the parameters for maximum acceleration (AC), maximum delay (DEC) and maximum speed (SP) for specific applications.

CAUTION!

Overshoot at maximum acceleration / delay

If the acceleration (AC) or delay (DEC) is set to the maximum value of 30 000 mm/s² or higher the effect of the ramp generator is switched off. With this setting, the maximum possible dynamic of the drive system is achieved. At this setting, at times the drive swings clearly beyond the target position.

► *Please note and take into account this fact during use.*

Basic settings

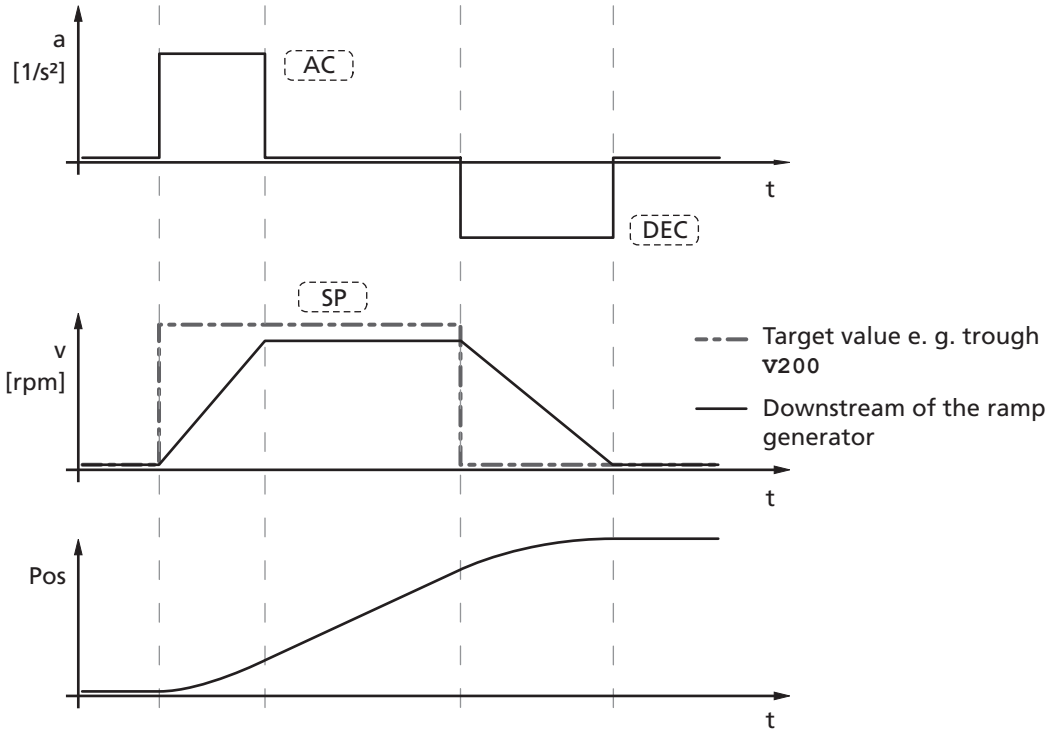
Command	Argument	Function	Description
AC	Value	Load Command Acceleration	Load acceleration value (1/s²). **Value: 0 … 30 000**
DEC	Value	Load Command Acceleration	Load deceleration value (1/s²). **Value: 0 … 30 000**
SP	Value	Load Maximum Speed	Load maximum speed (rpm). **Value: 0 … 30 000**

3 Operation in FAULHABER mode

3.6 Technical information

Ramp generator in velocity mode

Intervention of the ramp generator in velocity mode



In velocity mode the ramp generator acts like a filter on the target velocity. The target value is limited to the maximum speed value (SP) and target value changes are limited according to the deceleration and acceleration ramps (AC and DEC).

Notification of the higher level control

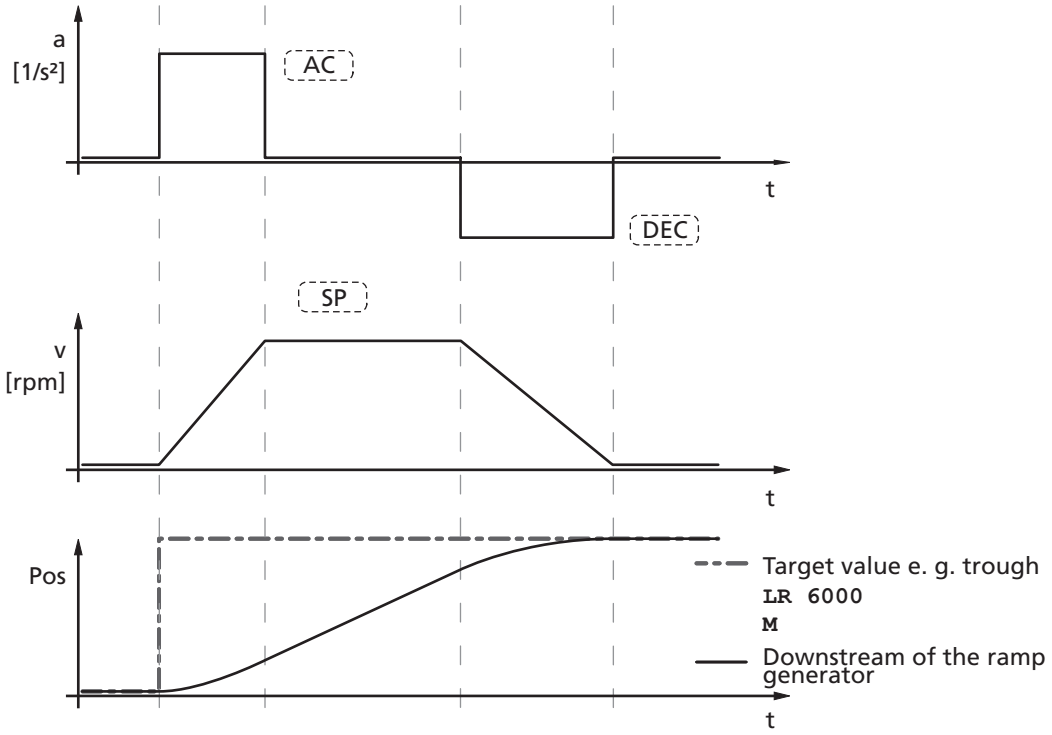
Attainment of the target velocity is signalled by bit 10 “Target reached” in the statusword of the drive. If the transmission type for the TxPDO1 is set to 255, the PDO is transmitted asynchronously, triggered by the change in state.

3 Operation in FAULHABER mode

3.6 Technical information

Ramp generator in positioning mode

Intervention of the ramp generator in positioning mode



In positioning mode a preset speed is determined by the position controller from the difference between the target position and actual position.

In the ramp generator, the preset speed output by the position controller is limited to the maximum speed value (SP) and accelerations are limited according to the acceleration ramp (AC).

In positioning mode the deceleration process is not extended as, before reaching the limit position, the speed has to be reduced so that the target position can be reached without overshooting.

According to the equation of motion:

$2a\,s = v^2 \rightarrow v_{max} = \sqrt{2a\,s}$

a: Acceleration [m/s²]

v: Velocity [m/s]

s: remaining distance [m]

the maximum speed n_{max} must be limited proportional to the remaining distance.

The allowable delay, or rather the technically possible delay depending on the motor and inertia of the load, is set here using the parameter DEC.

Notification of the higher level control

Attainment of the target position is signalled by bit 10 “Target reached” in the statusword of the drive. If the transmission type for the TxPDO1 is set to 255, the PDO is transmitted asynchronously, triggered by the change in state.

Complex motion profiles

More complex motion profiles can be generated through appropriate presetting of new values (maximum speed, acceleration, end position) during positioning.

After a value change, simply execute a new motion start command (M).

The positioning range can be set using the command LL and activated using APL.

Command	Argument	Function	Description
LL	Value	Load Position Range Limits	Load limit positions (the drive cannot be moved out of these limits). Positive values specify the upper limit and negative values the lower. The range limits are only active if APL is 1. **Value: $-1.8 \cdot 10^9$ … $+1.8 \cdot 10^9$**
APL	0 - 1	Activate/Deactivate Position Limits	Activate range limits (LL) (valid for all operating modes except VOLTMOD). 1: Position limits activated 0: Position limits deactivated

3 Operation in FAULHABER mode

3.6 Technical information

3.6.2 Sinus commutation

The outstanding feature of FAULHABER motion controllers for brushless motors is their so-called sinus commutation. This means that the specified rotating field is always ideally positioned relative to the rotor. As a result, torque fluctuations can be reduced to a minimum, even at very low speeds. In addition, the motor runs particularly quietly.

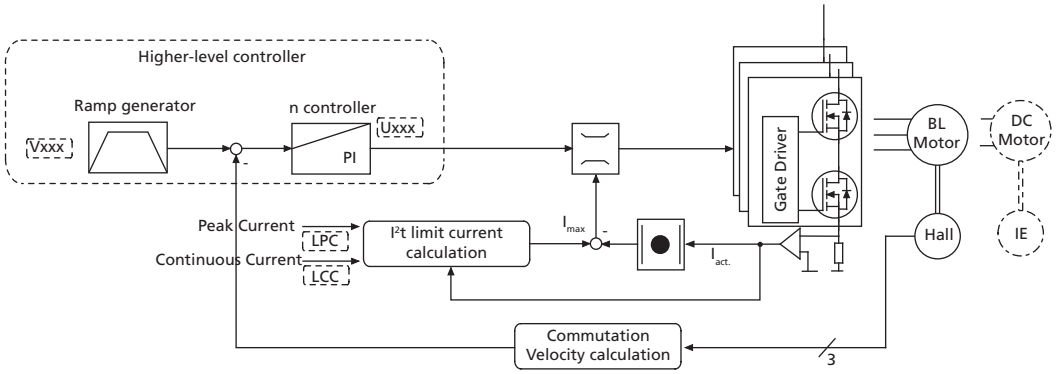
The sinus commutation is further enhanced by so-called flat-top modulation, which enables more modulation. As a result, higher no-load speeds are possible.

The SIN0 command can even be used to set the system so that the sinus commutation switches to block commutation in the upper speed range. This full modulation enables the complete speed range of the motor to be utilised.

Command	Function	Description
SIN	Sinus commutation	0: Full control (block mode with full control) 1: Limited to sinusoidal form (basic setting)

3.6.3 Current controller and I²t current limitation

Intervention of the current limiting controller



The FAULHABER Motion Controllers are equipped with an integral current controller, which enables torque limitation.

The current controller operates as a limitation controller. Depending on the previous loading, the I²t current limitation limits to the allowable peak current or continuous current. As soon as the motor current exceeds the currently allowed maximum value the current controller limits the voltage.

Due to its design as a current limiting controller, current control in the thermally relaxed state has no effect on the dynamic of the velocity control. The time response of this limitation can be adjusted using the parameter CI.

The default values for CI limit the current to the allowable value after around 5ms.

3 Operation in FAULHABER mode

3.6 Technical information

Basic settings

Command	Argument	Function	Description
LPC	Value	Load Peak Current Limit	Load peak current **Value: 0 to 12 000 mA**
LCC	Value	Load Continuous Current Limit	Load continuous current **Value: 0 to 12 000 mA**
CI	Value	Load Current Integral Term	Load integral term for current controller **Value: 1...255**

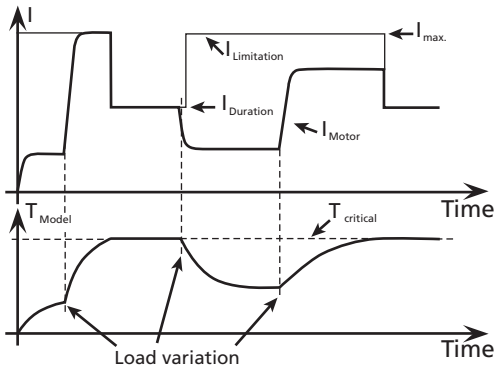
Mode of operation of the current controller

When the motor starts, the peak current is preset as the set-point for the current controller. As the load increases, the current in the motor constantly increases until it finally reaches the peak current. The current controller then comes into operation and limits the current to this set-point.

A thermal current model operating in parallel calculates a model temperature from the actually flowing current. If this model temperature exceeds a critical value, continuous current is switched to and the motor current is regulated to this. Only when the load becomes so small that the temperature falls below the critical model temperature is peak current permitted again.

The aim of this so-called I²t current limiting is not to heat the motor above the thermally allowable temperature by selecting a suitable continuous current. On the other hand, a high load should be temporarily possible in order to enable very dynamic movements.

Function of the I²t current limitation



3.6.4 Overtemperature protection

If the MOSFET temperature of the external controllers or the coil temperature of the drives with integrated controller exceeds a preset limit value, the motor is switched off. The following conditions must be fulfilled in order to reactivate the motor:

- Temperature below a preset limit value
- Target velocity set to 0 rpm
- Actual motor speed less than 50 rpm

NOTE

Determining the coil temperature

The housing temperature is measured and the power loss concluded from the current measurement. The MOSFET or coil temperature is calculated from these values via a thermal model. In most applications, this method represents a thermal motor protection device.

3.6.5 Under-voltage monitoring

If the supply voltage falls below the lower voltage threshold, the power stage is switched off. The Motion Controller remains active. When the voltage returns within the permissible range, the power stage is switched on again immediately.

3.6.6 Overvoltage regulation

If the motor is operated as a generator, it produces energy. Usually power supply units are not able to feed this energy back into the power line. For this reason, the supply voltage at the motor increases, and depending on the speed, the allowable maximum voltage may be exceeded.

In order to avoid irreparable damage to components, FAULHABER motion controllers for brushless motors contain a controller which adjusts the rotor displacement angle if a limit voltage (32 V) is exceeded. Motion controllers for DC motors contain a ballast circuit which is activated if a limit voltage (32 V) is exceeded. As a result, the energy generated in the motor is converted, and the voltage of the electronics remains limited to 32 V. This method protects the drive during generating operation and rapid braking.

3.6.7 Adjustment of the controller parameters

The preset controller parameters must be optimised in order to optimally adjust the controller to the respective application.

NOTE

Controller sampling rate

The digital controller operates at a sampling rate of 100 µs. If necessary the sampling rate can be increased to up to 2 ms via the Sampling Rate Parameter.

3 Operation in FAULHABER mode

3.6 Technical information

Default behaviour:

Without further settings, the gain set in the parameter POR is effective for the velocity controller.

In Positioning Mode the gain set via the parameter POR is increased within the target corridor by the value of the parameter PD. This enables faster adjustment to the stoppage in the target position without having to over-stimulate the controller during the transient phenomena. To this end, the parameter PD must be set carefully and should typically be a maximum of 50% of the base value POR; otherwise there is a risk of instability.

The following controller parameters are available:

Command	Function	Description
POR	Load Velocity Proportional Term	Load velocity controller amplification. **Value: 1 – 255.**
I	Load Velocity Integral Term	Load velocity controller integral term **Value: 1 – 255.**
PP	Load Position Proportional Term	Load position controller amplification. **Value**: 1 – 255.
PD	Load Position D-Term	Load position controller D-term. **Value: 1 – 255.**
SR	Load Sampling Rate	Sampling rate setting. **Value: 1 … 20 ms/10**

In the case of integrated units these values are already preset, however, they can be adjusted to the driving load using the Motion Manager's Motor Wizard. These values are suitably preassigned for external controls by selecting a motor type in the Motion Manager's Motor Wizard.

The controller tuning Wizard in Motion Manager can be used to further adjust several controller parameters, in order to optimally adjust the controller to the respective application.

Possible procedure

It is recommended that you begin with the default settings of the Motor Wizard and then further optimise the velocity controller first and then the position controller.

1.) Optimise velocity controller:

Use, for example, the controller tuning Wizard to make velocity jumps between 1/3 and 2/3 of the maximum velocity and at the same time increase the controller gain POR gradually, until the controller becomes unstable. The controller gain must then be reduced again until reliable stability exists. Under certain circumstances it may be necessary to optimise the integral term I accordingly.

2.) Optimise position controller:

Specify appropriate motion profiles for the application, e.g. using the controller tuning Wizard. If the system does not function stably with these settings, stability can be achieved by reducing the I term of the velocity controller or reducing the P term of the position controller. Then increase the P term of the position controller gradually up to the system's stability limit. The stability can then be restored, either by increasing the D term of the position controller or by reducing the I term of the velocity controller.

3 Operation in FAULHABER mode

3.6 Technical information

Special mode for position control

The SR command can be used to activate a special position control mode (Gain Scheduling). To this end, the value 100 must be added to the required SR setting.

Example:

Required setting SR10 with special mode: `SR110`.

If this mode is activated, the parameter POR is successively reduced in a position-controlled application as soon as the drive in within the target corridor (can be set using the CORRIDOR command). This enables much "gentler" stoppage to be achieved in the target position. As soon as the drive leaves the target corridor, POR is immediately increased back to the set value.

NOTE

The "Gain Scheduling" function only becomes active at sampling rates with a factor larger than 3 (sampling rate > 3).

4 CANopen protocol description

Guide

Introduction	Page 59
PDOs (process data objects)	Page 61
SDO (service data object)	Page 63
Emergency Object (error message)	Page 65
SYNC object	Page 66
NMT (network management)	Page 67
Entries in the object dictionary	Page 70

4.1 Introduction

- CANopen is a standardised software protocol based on the CAN hardware (Controller Area Network).
- The international CAN Organisation CAN in Automation e.V. (CiA) defines the communication profile in DS301 (Description of the communication structure and methods for parameter access, control and monitoring functions).
- Device profiles are specified for the different devices, such as DSP402 for drives and DS401 for I/O devices (general device description from the view of the user).
- Public data is managed using the object dictionary (Parameter table, access to entries via Index and Subindex).
- There are two data communication objects:
  - PDOs (Process data objects for control and monitoring)
  - SDOs (Service data objects for access to the object dictionary)
- Further objects are available for network management, node monitoring and synchronisation.
- CANopen supports up to 127 nodes per network segment with transmission rates up to 1 MBit/s.
- The communication is message based, each communication object is assigned its own 11 bit identifier.

4 CANopen protocol description

4.1 Introduction

FAULHABER Motion Controllers support the CANopen communication profile in accordance with CiA DS301 V4; the following communication objects are supported:

- 3 transmit PDOs
- 3 receive PDOs
- 1 server SDO
- 1 emergency object
- NMT with Node Guarding
- 1 SYNC object

The identifier configuration of the CANopen objects is defined according to the "Predefined Connection Set" (see Chapter 4.6 "NMT (network management)"). Data assignment of the PDOs is permanently preset (static PDO mapping).

Many manufacturers offer CANopen libraries for PC and PCS systems, via which the individual objects can be conveniently accessed, without having to worry about the internal structure.

The FAULHABER Motion Manager also enables easy access to the individual objects via a graphic user interface.

4 CANopen protocol description

4.2 PDOs (process data objects)

PDOs correspond to a CAN message frame with up to 8 bytes and are used to transmit process data, i.e. to control and monitor the device's behaviour. The PDOs are designated from the point of view of the field device. Receive PDOs (RxPDOs) are received by the field device and contain, e.g. control data, send PDOs (TxPDOs) are sent by the field device and contain, e.g. monitoring data.

PDOs can only be transmitted if the device is in "Operational" state (see Chapter 4.6 "NMT (network management)").

PDO communication types:

- Event controlled: Data are automatically sent following a change to the device.
- Remote Request (RTR): Data are sent following a request message frame.
- Synchronised: Data are sent following receipt of a SYNC object, see Chapter 4.5 "SYNC object".

FAULHABER Motion Controllers provide the following PDOs:

- Receive PDO1: Controlword according to DSP402
- Send PDO1: Statusword according to DSP402
- Receive PDO2: FAULHABER command
- Send PDO2: FAULHABER query data (RTR)
- Receive PDO3: FAULHABER trace configuration
- Send PDO3: FAULHABER trace data (RTR)

RxPDO1: *Controlword*

11 bit identifier	2 bytes user data	
0x200 (512d) + Node ID	LB	HB

Contains the 16 bit controlword according to CiA DSP402, which controls the state machine of the drive unit. The PDO refers to object index 0x6040 in the object dictionary. The bit allocation is described in Chapter 6.1 "Device Control".

TxPDO1: *Statusword*

11 bit identifier	2 bytes user data	
0x180 (384d) + node ID	LB	HB

Contains the 16 bit controlword according to CiA DSP402, which displays the state machine of the drive unit. The PDO refers to object index 0x6041 in the object dictionary. The bit allocation is described in Chapter 6.1 "Device Control".

RxPDO2: FAULHABER Command

11 bit identifier	5 bytes user data				
0x300 (768d) + Node-ID	Cmd	LLB	LHB	HLB	HHB

Is made available by the FAULHABER channel for transmission of manufacturer-specific commands. All the parameters and control commands of the drive unit can be transmitted with the help of this PDO. Transmissions are always 5 bytes long, whereby the first byte gives the command and the following 4 bytes give the argument as a long integer value. A description of the commands is given in Chapter 8.4 "FAULHABER commands".

4 CANopen protocol description

4.2 PDOs (process data objects)

TxPDO2: FAULHABER Data

11 bit identifier	6 bytes user data					
0x280 (640d) + Node-ID	Cmd	LLB	LHB	HLB	HHB	Error

FAULHABER Channel for query commands. A request (RTR) on this PDO returns the data requested with the previously sent command. Transmissions are always 6 bytes long, whereby the first byte gives the command and the following 4 bytes give the required value as a long integer followed by an error code. The error byte can also be used to check whether a send command was successfully executed or not (1 = command successfully executed, for further error codes see Chapter 8.4 “FAULHABER commands”).

RxPDO3: Trace Configuration

11 bit identifier	5 bytes user data				
0x400 (1024d) + node ID	Mode 1	Mode 2	TC	Packets	Period

This PDO is used to set the trace mode, via which the internal parameters can be quickly read out.

The data configuration looks like this:

Byte 0: Mode for Parameter 1

Byte 1: Mode for Parameter 2

Byte 2: Transfer with time code [1/0]

Byte 3: Number of packets to be transmitted per request (Default: 1)

Byte 4: Time interval between packets (Default: 1 ms)

The possible operating modes for Parameters 1 and 2 are described in Chapter 5.2 “Trace”.

TxPDO3: Trace Data

11 bit identifier	3 to 8 byte user data							
0x380 (896d) + Node-ID	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7

A request (RTR) on this PDO returns the trace data according to the setting made via RxPDO3 (see Chapter 5.2 “Trace”).

4 CANopen protocol description

4.3 SDO (service data object)

The service data object can be used to read and describe parameters in the object dictionary (OD). They are accessed via the 16 bit index and the 8 bit subindex. The Motion Controller functions as a server, i.e. it makes data available (upload) at the request of the client (PC, PCS) (Upload) or receives data from the client (download).

Byte0	Byte1-2	Byte3	Byte4
Command Specifier	16 bit index	8 bit subindex	1-4 byte parameter data

➔ Entry in the object dictionary

A differentiation is made between 2 SDO transfer types:

- Expedited transfer: Transfer of 4 bytes maximum
- Segmented transfer: Transfer of more than 4 bytes

As, apart from for query of the version and the device name, only 4 bytes maximum are transferred by the FAULHABER Motion Controllers, only the expedited transfer is described in the following.

The size of the message frames is always 8 bytes and their structure is as follows:

Read OD entries: Client ➔ Server, Upload Request

11 bit identifier	8 bytes user data							
0x600 (1536d) + Node-ID	0x40	Index LB	Index HB	Subindex	0	0	0	0

Server ➔ Client, Upload Response

11 bit identifier	8 bytes user data							
0x580 (1408d) + Node-ID	0x4x	Index LB	Index HB	Subindex	LLB (D0)	LHB (D1)	HLB (D2)	HHB (D3)

Byte0 (0x4x) gives the number of valid data bytes in D0-D3 and the transfer type and is coded for expedited transfer (≤ 4 data bytes) as follows:

- 1 data byte in D0: Byte0 = 0x4F
- 2 data bytes in D0-D1: Byte0 = 0x4B
- 3 data bytes in D0-D2: Byte0 = 0x47
- 4 data bytes in D0-D3: Byte0 = 0x43

Write OD entries: Client ➔ Server, Download Request

11 bit identifier	8 bytes user data							
0x600 (1536d) + Node-ID	0x2x	Index LB	Index HB	Subindex	LLB (D0)	LHB (D1)	HLB (D2)	HHB (D3)

Byte0 (0x2x) gives the number of valid data bytes in D0-D3 and the transfer type and is coded for expedited transfer (≤ 4 data bytes) as follows:

- 1 data byte in D0: Byte0 = 0x2F
- 2 data bytes in D0-D1: Byte0 = 0x2B
- 3 data bytes in D0-D2: Byte0 = 0x27
- 4 data bytes in D0-D3: Byte0 = 0x23

If it is not necessary to specify the number of data bytes: Byte0 = 0x22

Server ➔ Client, Download Response

11 bit identifier	8 bytes user data							
0x580 (1407d) + node ID	0x60	Index LB	Index HB	Subindex	0	0	0	0

4 CANopen protocol description

4.3 SDO (service data object)

Termination of the SDO protocols in the event of an error:

Client ➔ Server

11 bit identifier	8 bytes user data							
0x600 (1536d) + Node-ID	0x80	Index LB	Index HB	Subindex	Error0	Error1	Error2	Error3

Server ➔ Client

11 bit identifier	8 bytes user data							
0x580 (1408d) + Node-ID	0x80	Index LB	Index HB	Subindex	Error0	Error1	Error2	Error3

Error3 Error class

Error2: Error code

Error1: Additional error code HB

Error0: Additional error code LB

Error class	Error code	Additional code	Description
0x05	0x03	0x0000	Toggle bit unchanged
0x05	0x04	0x0001	SDO Command Specifier invalid or unknown
0x06	0x01	0x0000	Access to this object is not supported
0x06	0x01	0x0002	Attempt to write on a Read_Only parameter
0x06	0x02	0x0000	Object does not exist in the object dictionary
0x06	0x04	0x0041	Object cannot be mapped in PDO
0x06	0x04	0x0042	Number and/or length of the mapped objects would exceed PDO length
0x06	0x04	0x0043	General parameter incompatibility
0x06	0x04	0x0047	General internal error in the device
0x06	0x06	0x0000	Access cancelled due to hardware fault
0x06	0x07	0x0010	Data type or parameter length do not match or are unknown
0x06	0x07	0x0012	Data type does not match, parameter length is too large
0x06	0x07	0x0013	Data type does not match, parameter length is too short
0x06	0x09	0x0011	Subindex not available
0x06	0x09	0x0030	General value range error
0x06	0x09	0x0031	Value range error: Parameter value too large
0x06	0x09	0x0032	Value range error: Parameter value too small
0x06	0x0A	0x0023	Resource not available
0x08	0x00	0x0021	Access not possible due to local application
0x08	0x00	0x0022	Access not possible due to current device status

4 CANopen protocol description

4.4 Emergency Object (error message)

The emergency object informs other bus devices of errors that have occurred.

The size of the Emergency Object is always 8 bytes and its structure is as follows:

11 bit identifier	8 bytes user data							
0x80 (128d) + Node-ID	Error0 (LB)	Error1 (HB)	Error reg.	0	0	0	0	0

The first two bytes contain the 16 bit error code, the third byte contains the error register (content of object 0x1001), bytes 4 and 5 contain the 16 bit FAULHABER error register (content of object 0x2320), the remaining bytes are unused (always 0).

The error register identifies the error type. The individual error types are bit coded and are assigned the respective error codes in the following table. The object 0x1001 can be used to query the last value of the error register.

The following error code table lists all errors reported by emergency message frames, provided the corresponding error is set in the emergency mask for the FAULHABER error register. Only those errors for which an emergency mask is given in this table are reported.

Emergency error codes

Error code	Meaning	Emergency mask	Error register bit
0x0000	no error		
0x1000	generic error		0
0x2000	current		
0x2300	current, device output side		
0x2310	continuous over current	0x0001	1
0x3000	voltage		
0x3200	voltage inside the device		
0x3210	over voltage	0x0004	2
0x4000	temperature		
0x4300	drive temperature		
0x4310	over temperature	0x0008	3
0x5000	device hardware		
0x5500	data storage		
0x5530	flash memory error	0x0010	5
0x6000	device software		
0x6100	internal software	0x1000	5
0x8000	monitoring		
0x8100	communication		
0x8130	life guard error	0x0100	4
0x8140	recovered from bus off	0x0200	4
0x8400	velocity speed controller (deviation)	0x0002	5
0x8600	positioning controller		
0x8611	following error (deviation)	0x0002	5

Example:

If 0x2320 is set in the FAULHABER error register under Subindex 2 Bit 1, an emergency message frame with 8 data bytes 0x10 0x23 0x01 0x00 0x00 0x00 0x00 0x00 is sent if the current limitation value set using LCC was exceeded for longer than the error delay time set with DCE.

Error Register

Bit	Meaning
0	Generic error
1	Current
2	Voltage
3	Temperature
4	Communication error (overrun, error state)
5	Device profile specific
6	Reserved (always 0)

4 CANopen protocol description

4.5 SYNC object

The SYNC object is a short message frame without data content, which is used to trigger synchronous PDOs and therefore enables quasi simultaneous starting of processes on different devices.

The identifier of the SYNC object can be set in the object dictionary under Index 0x1005 (Default: 0x80).

11 bit identifier	No user data
0x80	

Whether a PDO is to be triggered by a SYNC object or not can be set using the transmission type in the Communication Parameter objects of the corresponding PDO (see Chapter 8.1 "Communication objects according to CiA 301").

A differentiation is made between the following PDO transmission types:

Transmission type	Meaning
255	asynchronous (event controlled)
253	asynchronous, only on request (RTR)
252	synchronous, only on request (RTR) PDO is only sent on request following a SYNC object
1 – 240	synchronous, cyclical PDO is repeatedly sent following a SYNC object The given value simultaneously represents the number of SYNC objects which have to have been received before the PDO is sent again (1 = PDO is sent with each SYNC object).
0	synchronous, acyclic PDO is sent or executed once following a SYNC object, if it has changed its content (new parameter query or status change)

Synchronous receive PDO:

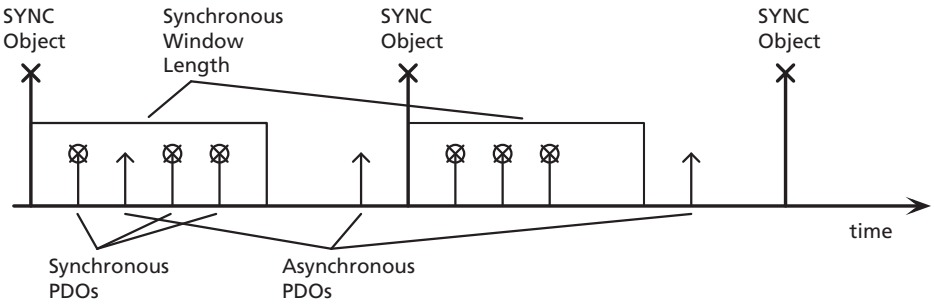
The command transmitted with the PDO is not executed until the SYNC objects is received. In this way, e.g. several axles can be synchronised with each other.

NOTE

In the case of RxPDOs, the transmission types 1-240 are identical to transmission type 0.

Synchronous transmit PDO:

After receiving a SYNC object, the PDO is sent as quickly as possible with the current data (Synchronous Window Length = 0):



NOTE

Transmission types 1-240 can also be used to group nodes.

4 CANopen protocol description

4.6 NMT (network management)

After switching on and initialisation has been successfully performed, the FAULHABER Motion Controllers are automatically in the "Pre-Operational" state. Apart from via NMT messages, in this state it is only possible to communicate with the device via service data objects (SDOs), to make or query parameter settings. FAULHABER Motion Controllers are delivered complete with useful default settings for all objects; therefore, in general it is not necessary to assign parameters with the system start. Necessary parameter settings are usually performed once, e.g. with the help of the FAULHABER Motion Manager and are then permanently stored in the data flash. These settings are then immediately available following the system start.

A single CAN message is sufficient to start a CANopen device:

Start Remote Node:

11 bit identifier	2 bytes user data	
0x000	0x01	Node-ID

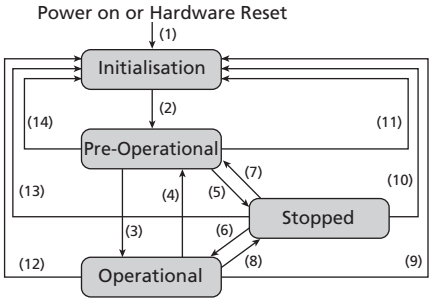
Or, to start the whole network:

Start All Remote Nodes:

11 bit identifier	2 bytes user data	
0x000	0x01	0x00

The devices are then in "Operational" state. The device is now fully functional and can be operated via PDOs.

The state diagram is given in the following:



(1)	At Power on the initialisation state is entered autonomously
(2)	Initialisation finished – enter PRE-OPERATIONAL automatically
(3), (6)	Start_Remote_Node indication
(4), (7)	Enter PRE-OPERATIONAL_State indication
(5), (8)	Stop_Remote_Node indication
(9), (10), (11)	Reset_Node indication
(12), (13), (14)	Reset_Communication indication

In the "Stopped" ("Prepared") state, the device is in the error state and can no longer be operated using SDO and PDOs. Only NMT messages are received, to cause a state change. State changes can be performed with the help of the NMT services:

An NMT message frame always consists of 2 bytes on the identifier 0x000:

11 bit identifier	2 bytes user data	
0x000	CS	Node-ID

CS: Command Specifier

Node-ID: Node address (0 = all nodes)

4 CANopen protocol description

4.6 NMT (network management)

The possible values for the Command Specifier CS are listed in the following table:

State transition	Command Specifier CS	Explanation
(1)	–	The initialisation status is reached autonomously on switching on.
(2)	–	Following initialisation the pre-operational status is reached automatically, at the same time the boot-up message is sent.
(3), (6)	CS = 0x01 (1d)	Start_Remote_Node. Starts the device and releases the transmission of PDOs.
(4), (7)	CS = 0x80 (128d)	Enter_Pre-Operational. Stops the PDO transmission, SDO continues to be active.
(5), (8)	CS = 0x02 (2d)	Stop_Remote_Node. Device changes to error state, SDO and PDO are switched off.
(9), (10), (11)	CS = 0x81 (129d)	Reset_Node. Performs a reset. All objects are reset to power-on defaults.
(12), (13), (14)	CS = 0x82 (130d)	Reset_Communication. Resets the communication functions.

Boot-Up Message:

Following the initialisation phase, the FAULHABER Motion Controller sends the Boot-Up Message, a CAN message with one data byte (Byte0 = 0x00) on the identifier of the node guarding message (0x700 + Node ID):j

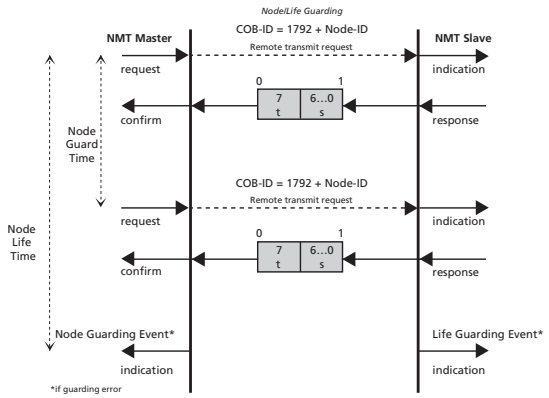
11 bit identifier	1 bytes user data
0x700 (1792d) + Node ID	0x00

The boot-up message signals the end of the initialisation phase of a newly activated module, which can then be configured or started.

Node guarding / life guarding:

The node guarding object can be used to query the momentary state of the device. To do this, by setting a remote frame, the master sends a request (request message frame) on the guarding identifier of the node to be monitored. This then replies with the guarding message, which contains the current status of the node and a toggle bit.

The following diagram describes the node guarding protocol:



t: Toggle bit. Initially 0, changes its value in each guarding message frame.

s: Status:

s = 0x04 (4d): Stopped

s = 0x05 (5d): Operational

s = 0x7F (127d): Pre-operational

If a node life time > 0 is set (objects 0x100C and 0x100D), a life-guarding-error is set, if no more node guarding queries of the master arrive within the given life time (life-guarding).

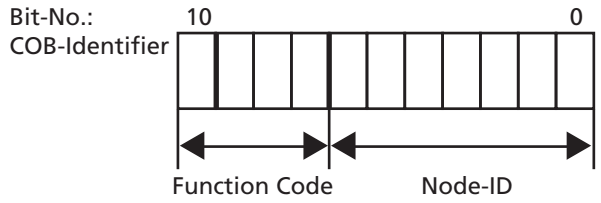
The response to a Life Guarding error can be set via the FAULHABER error register (Object 0x2320). Otherwise, by default, no action is performed.

4 CANopen protocol description

4.6 NMT (network management)

Identifier distribution:

CANopen provides default identifiers in the "Predefined Connection Set" for the most important objects. These are made up of a 7 bit node address (Node-ID) and a 4 bit function code in accordance with the following schema:



FAULHABER Motion Controller only operate with these default identifiers:

Object	Function code (binary)	Resulting COB-ID	Communication Parameters at Index
NMT	0000	0	–
SYNC	0001	128 (80h)	1005h

Object	Function code (binary)	Resulting COB-ID	Communication Parameters at Index
EMERGENCY	0001	129 (81h) – 255 (FFh)	1014h
PDO1 (tx)	0011	385 (181h) – 511 (1FFh)	1800h
PDO1 (rx)	0100	513 (201h) – 639 (27Fh)	1400h
PDO2 (tx)	0101	641 (281h) – 767 (2FFh)	1801h
PDO2 (rx)	0110	769 (301h) – 895 (37Fh)	1401h
PDO3 (tx)	0111	897 (381h) – 1023 (3FFh)	1802h
PDO3 (rx)	1000	1025 (401h) – 1151 (47Fh)	1402h
SDO (tx)	1011	1409 (581h) – 1535 (5FFh)	1200h
SDO (rx)	1100	1537 (601h) – 1663 (67Fh)	1200h
NMT Error Control	1110	1793 (701h) – 1919 (77Fh)	

4 CANopen protocol description

4.7 Entries in the object dictionary

The configuration parameters are managed in the CANopen object dictionary. The object dictionary is divided into three areas:

1. Communication parameters (Index 0x1000 – 0x1FFF)
2. Manufacturer specific area (Index 0x2000 – 0x5FFF)
3. Standardised device profiles (0x6000 – 0x9FFF)

The 1st area contains the objects according to DS301, the 2nd area is reserved for manufacturer-specific objects and the 3rd area contains the objects according to DSP402 supported by the FAULHABER Motion Controllers.

Each object can be referenced via its index and subindex (SDO protocol).

Overview of the available objects:

a.) Communication objects according to DS301:

Index	Object	Name	Type	Attribute
0x1000	VAR	device type	UNSIGNED32	ro
0x1001	VAR	error register	UNSIGNED8	ro
0x1003	ARRAY	pre-defined error field	UNSIGNED32	rw
0x1005	VAR	COB-ID SYNC	Unsigned32	rw
0x1008	VAR	manufacturer device name	Vis-String	const
0x1009	VAR	manufacturer hardware version	Vis-String	const
0x100A	VAR	manufacturer software version	Vis-String	const
0x100C	VAR	guard time	UNSIGNED16	rw
0x100D	VAR	life time factor	UNSIGNED8	rw
0x1010	ARRAY	store parameters	UNSIGNED32	rw
0x1011	ARRAY	restore default parameters	UNSIGNED32	rw
0x1014	VAR	COB-ID EMCY	UNSIGNED32	ro
0x1018	RECORD	identity object	Identity (23h)	ro
		Server SDO Parameter		
0x1200	RECORD	1st server SDO parameter SDO	SDOParameter	ro
		Receive PDO Communication Parameter		
0x1400	RECORD	1st receive PDO parameter PDO	PDOCommPar	rw
0x1401	RECORD	2nd receive PDO parameter PDO	PDOCommPar	rw
0x1402	RECORD	3rd receive PDO Parameter PDO	PDOCommPar	rw
		Receive PDO Mapping Parameter		
0x1600	RECORD	1st receive PDO mapping PDO	PDOMapping	ro
0x1601	RECORD	2nd receive PDO mapping PDO	PDOMapping	ro
0x1602	RECORD	3rd receive PDO mapping PDO	PDOMapping	ro
		Transmit PDO Communication Parameter		
0x1800	RECORD	1st transmit PDO parameter PDO	PDOCommPar	rw
0x1801	RECORD	2nd transmit PDO parameter PDO	PDOCommPar	rw
0x1802	RECORD	3rd transmit PDO parameter PDO	PDOCommPar	rw
		Transmit PDO Mapping Parameter		
0x1A00	RECORD	1st transmit PDO mapping PDO	PDOMapping	ro
0x1A01	RECORD	2nd transmit PDO mapping PDO	PDOMapping	ro
0x1A02	RECORD	3rd transmit PDO mapping PDO	PDOMapping	ro

4 CANopen protocol description

4.7 Entries in the object dictionary

b.) Drive profile objects according to DSP402:

Index	Name	Type	Attribute	Meaning
0x6040	controlword	Unsigned16	rw	Drive control
0x6041	statusword	Unsigned16	ro	Status display
0x6060	modes of operation	Integer8	wo	Operating mode changeover
0x6061	modes of operation display	Integer8	ro	Set operating mode
0x6062	position demand value	Integer32	ro	Last target position scaled
0x6063	position actual value	Integer32	ro	Actual position in increments
0x6064	position actual value	Integer32	ro	Actual position scaled
0x6067	position window	Unsigned32	rw	Target position window
0x6068	position window time	Unsigned16	rw	Time in target position window
0x6069	velocity actual sensor value	Integer32	ro	Current speed value
0x606B	velocity demand value	Integer32	ro	Target velocity
0x606C	velocity actual value	Integer32	ro	Current speed value
0x606D	velocity window	Unsigned16	rw	End speed window
0x606E	velocity window time	Unsigned16	rw	Time in end speed window
0x606F	velocity threshold	Unsigned16	rw	Speed threshold value
0x6070	velocity threshold time	Unsigned16	rw	Time below speed threshold value
0x607A	target position	Integer32	rw	Target position
0x607C	homing offset	Integer32	rw	Reference point offset
0x607D	software position limit	ARRAY Integer32	rw	Area limits
0x607E	polarity	Unsigned8	rw	Polarity (direction of rotation)
0x607F	max profile velocity	Unsigned32	rw	Maximum speed
0x6081	profile velocity	unsigned32	rw	Maximum speed
0x6083	profile acceleration	Unsigned32	rw	Acceleration value
0x6084	profile deceleration	Unsigned32	rw	Braking ramp value
0x6085	quick stop deceleration	Unsigned32	rw	Quick stop braking ramp value
0x6086	motion profile type	Integer16	ro	Motion profile
0x608F	position encoder resolution	ARRAY Unsigned32	rw	Resolution of the external encoder
0x6093	position factor	ARRAY Unsigned32	rw	Position factor
0x6096	velocity factor	ARRAY Unsigned32	rw	Speed factor
0x6097	acceleration factor	ARRAY Unsigned32	rw	Acceleration factor
0x6098	homing method	Integer8	rw	Homing method
0x6099	homing speed	ARRAY Unsigned32	rw	Homing speed
0x609A	homing acceleration	Unsigned32	rw	Homing acceleration
0x60F9	velocity control parameter set	ARRAY Unsigned16	rw	Parameters for speed controller
0x60FA	control effort	Integer32	ro	Controller output
0x60FB	position control parameter set	ARRAY Unsigned16	rw	Parameters for position controller
0x60FC	position demand value	Integer32	ro	Last target position in increments
0x60FF	target velocity	Integer32	rw	Target velocity
0x6510	drive data	RECORD	rw	Drive information

A detailed description of the individual objects is given in Chapter 8 “Parameter description”.

5 Extended CAN functions

5.1 The FAULHABER channel

A special FAULHABER channel is available on, which can be used to execute all the Motion Controller's commands in a simple way.

A corresponding CAN message frame is available for each FAULHABER command, with which the CAN unit can be operated analogous to the serial version. All the functions and parameters of the drive unit can be activated using this channel.

Chapter 8.4 "FAULHABER commands" contains the complete description of the FAULHABER commands.

5.2 Trace

PDO3 can be used to trace operating data, i.e. to read it out online with a resolution of up to 1 ms. After the required trace type has been set using RxPDO3, the values can be requested consecutively by requests on TxPDO3 (see Chapter 4.2 "PDOs (process data objects)").

Trace configurations:

RxPDO3:

Byte	Function
0	Mode for Parameter 1
1	Mode for Parameter 2 255 = No second parameter
2	Transmission with time code 1 = with time code 0 = without time code
3	Number of data packets to be transmitted per request Default: 1
4	Time interval between packets [ms] Default: 1

The following values are available for Parameter 1 and 2:

0: Actual velocity [Integer16, rpm]
1: Target velocity [Integer16, rpm]
2: Controller output [Integer16]
4: Motor current [Integer16, mA]
44: Housing temperature [Unsigned16, °C]
46: Coil temperature [Unsigned16, °C]
200: Current position [Integer32, Inc]
201: Target position [Integer32, Inc]

Data request:

Depending on the mode set for Parameter 1 and 2, following a request (RTR) on TxPDO3, 3 to 8 bytes are returned on TxPDO3:

1.) Mode 1 between 0 and 15,
Mode 2 at 255 (inactive)

➔ 3 byte … 1st byte: Low byte data
2nd byte: High byte data
3rd byte: Timer code

The data are in Integer16 format.

2.) Mode 1 between 16 and 199,
Mode 2 at 255 (inactive)

➔ 3 bytes … Coding as for 1.)

The data are in Unsigned16 format.

3.) Mode 1 between 200 and 255,
Mode 2 at 255 (inactive)

➔ 5 byte … 1st byte: Lowest byte data
2nd byte: Second byte data
3rd byte: Third byte data
4th byte. Highest byte data
5th byte: Time code

The data are in Integer32 format.

4.) Mode1 corresponding to 1.), 2.) or 3.) and
Mode 2 less than 255:

➔ 5 to 8 bytes ...	Byte 1 to 2 (4):	Data bytes of Mode1
	Byte 3 (5) to 4 (6) (8):	Data bytes of Mode2
	Byte 5 (7):	Time code

The data bytes of Mode2 are coded as for Mode1.

The time code corresponds to a multiple of the time basis of 1 ms and defines the time interval to the last transmission. If two Integer32 parameters are requested, there is no more space for the time code in the CAN message frame; configuration parameter 2 must therefore be set to 0 (transmission without time code). The time must then be measured in the master.

NOTE

The trace data can also be read out using a SYNC object instead of via RTR. To do this, the transmission type of TxPDO3 in object 0x1802 must be set to a synchronous transmission type (see Chapter 4.5 "SYNC object"). In this way, data can be simultaneously read out from several nodes.

Example:

Record the actual position and motor current of node 1.

a.) Set trace configuration
Transmit ID 401: C8 04 01 01 01

b.) Data request
Request ID 381

c.) Answer
Receive ID 381: 10 27 00 00 32 00 03

➔	Byte 1 to 4:	Position 10 000
	Byte 5 to 6:	Motor current 50 mA
	Byte 7:	Timecode 3 ms

Further data can be requested by means of renewed requests on ID 381.

6 Functional description of the CANopen CiA 402

Guide

Device Control Page 75
Factor Group Page 80
Profile Position Mode and Position Control Function Page 82
Homing Mode Page 87
Profile Velocity Mode Page 91
Drive parameters/Common entries Page 94
Inputs/Outputs Page 95
Error handling Page 97

The CANopen device profile for drives and Motion Control applications (CiA 402) of the CANopen user organisation CAN in Automation (CiA) is based on the general CANopen protocol description CiA 301 as described in Chapter 4.

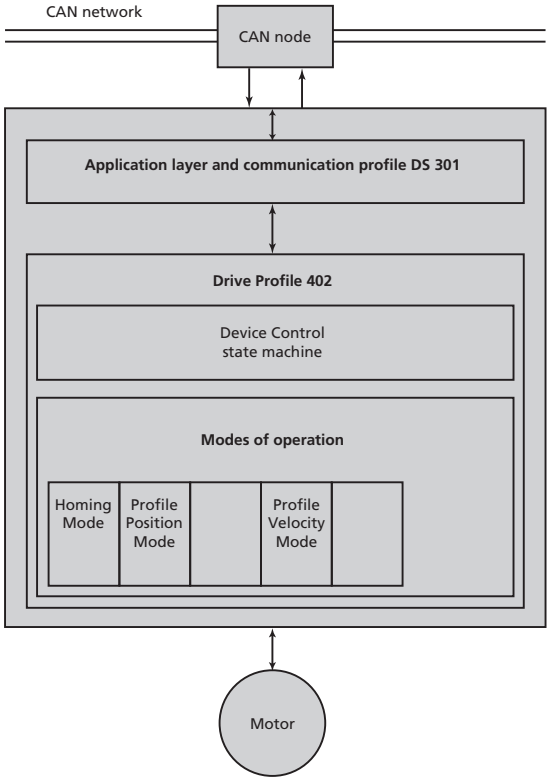
Communication with the drive takes place via the mechanisms described there. Before the drive can be addressed the baud rate must be set and a node number assigned to the CAN node. In addition, the underlying CANopen node must be activated using the network management (NMT) (see Chapter 4.6 “NMT (network management)”).

6 Functional description of the CANopen CiA 402

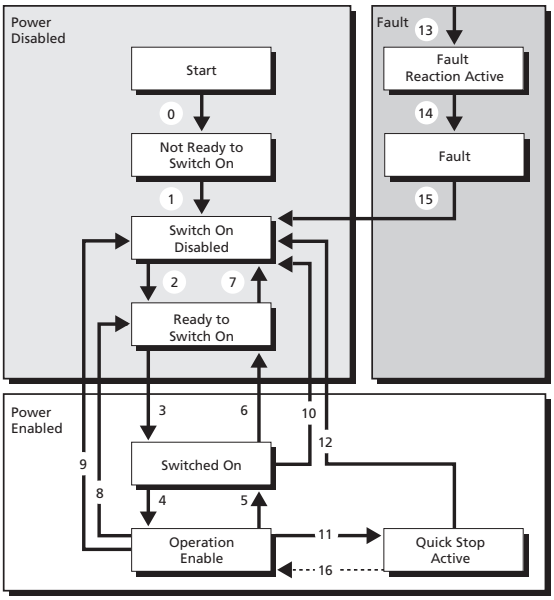
6.1 Device Control

FAULHABER Motion Control systems support "Device Control" from the CiA 402 profile and the "Profile Position Mode", "Profile Velocity Mode" and "Homing Mode" operating modes.

6.1.1 State machine of the drive



The drive behaviour is mapped in CANopen via a state machine. The states can be controlled with the controlword and displayed with the statusword:



After switching on and the initialisation has been successfully performed, the FAULHABER drive is immediately in "Switch On Disabled" state. At the same time, transitions 0 and 1 are run through autonomously.

A change in state within the state machine of the drive according to CiA 402 cannot be made until the underlying CANopen node is in the "Operational" state (see Chapter 4.6 "NMT (network management)").

The "Shutdown" command places the drive in "Ready to Switch On" state (transition 2).

The "Switch On" command then switches on the power stage. The drive is now enabled and is in "Switched On" state (transition 3).

The "Enable Operation" command places the drive in "Operation Enabled" state, the drive's normal operating mode (transition 4). The "Disable Operation" command places the drive back in "Switched On" state and is used, e.g. to terminate a running operation (transition 5).

6 Functional description of the CANopen CiA 402

6.1 Device Control

The state changes shown in the diagram are executed by the following commands:

Command	Transitions
Shutdown	2, 6, 8
Switch on	3
Disable Voltage	7, 9, 10, 12
Quick Stop	7, 10, 11
Disable Operation	5
Enable Operation	4, 16
Fault Reset	15

Controlword (0x6040)

The commands for executing state changes are executed by a combining bits 0 – 3 in the controlword. The controlword is located in the object dictionary under Index 0x6040 and is usually transmitted with PDO1.

Index	Subindex	Name	Type	Attrb.	Default value	Meaning
0x6040	0	controlword	Unsigned16	rw	0	Drive control

The bits in the controlword have the following meaning:

Bit	Function	Commands for Device Control State Machine						
		shut-down	Switch on	Disable Voltage	Quick Stop	Disable Operation	Enable Operation	Fault Reset
0	Switch on	0	1	X	X	1	1	X
1	Enable Voltage	1	1	0	1	1	1	X
2	Quick Stop	1	1	X	0	1	1	X
3	Enable Operation	X	0	X	X	0	1	X
4	New set-point/Homing operation start							
5	Change set immediately							
6	abs/rel							
7	Fault reset							0->1
8	Halt							
9	0							
10	0							
11	0							
12	0							
13	0							
14	0							
15	0							

Meaning of the other bits in the controlword:

Function	Description
New set-point	0: No new target position specified 1: New target position specified
Change set immediately	0: Start new positioning when current positioning has finished. 1: Interrupt current positioning and start a new one
abs/rel	0: Target position is an absolute value 1: Target position is a relative value
Fault reset	0->1: Reset fault
Halt	0: Movement can be made 1: Stop drive

The command sequences for starting a positioning, a speed control operation or a homing sequence are explained in the following sections.

6 Functional description of the CANopen CiA 402

6.1 Device Control

Example

Step sequence of the transitions in order to set a drive in Enable Operation state:

1. **Shutdown:**

   Controlword = 0x00 06

2. **Switch on:**

   Controlword = 0x00 07

   The drive is then in “Switched On” status. Operation must then be released to enable drive commands to be executed:

3. **Enable Operation:**

   Controlword = 0x00 0F

   The drive is then in “Operation Enabled” state, in which it can be operated using the relevant objects of the set operating mode.

Example

Step sequence of the transitions to get a drive from the error state:

1. **Fault reset:**

   Controlword = 0x00 80

2. **Shutdown:**

   Controlword = 0x00 06

3. **Switch on:**

   Controlword = 0x00 07

   The drive is then in “Switched On” status. Operation must then be released to enable drive commands to be executed:

4. **Enable Operation:**

   Controlword = 0x00 0F

   The drive is then in “Operation Enabled” state, in which it can be operated using the relevant objects of the set operating mode.

6 Functional description of the CANopen CiA 402

6.1 Device Control

Statusword (0x6041)

The current state of the drive is displayed in bits 0 – 6 of the statusword. In the event of state changes, the FAULHABER Motion Controller in its default setting automatically sends the current statusword on PDO1. The current state can also be queried at any time using a remote request on PDO1. The statusword is located in the object dictionary under Index 0x6041.

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6041	0	statusword	Unsigned16	ro	0	Status display

The bits of the statusword have the following meaning:

Bit	Function	State of the Device Control State Machine							
		Not Ready to Switch On	Switch On Disabled	Ready to Switch On	Switched On	Operation Enabled	Quick stop active	Fault reaction active	Fault
0	Ready to Switch On	0	0	1	1	1	1	1	0
1	Switched On	0	0	0	1	1	1	1	0
2	Operation Enabled	0	0	0	0	1	1	1	0
3	Fault	0	0	0	0	0	0	1	1
4	Voltage Enabled	X	X	X	X	X	X	X	X
5	Quick Stop	X	X	1	1	1	0	X	X
6	Switch On Disabled	0	1	0	0	0	0	0	0
7	Warning								
8	0								
9	Remote								
10	Target Reached								
11	Internal limit active								
12	Set-point acknowledge/Speed/Homing attained								
13	Homing Error								
14	Hard Notify								
15	0								

Meaning of the other bits in the statusword:

Function	Description
Warning	not used
Remote	not used
Target Reached	0: Target position or target velocity not yet reached 1: Target position or target velocity reached. (Halt = 1: Drive has reached speed 0)
Set-point acknowledge	0: New target position not yet adopted (Profile Position Mode) 1: New target position adopted
Homing attained	0: Homing sequence not yet completed 1: Homing sequence successfully completed
Speed	0: Speed not equal to 0 (Profile Velocity Mode) 1: Speed 0
Homing Error	0: No error 1: Error
Hard Notify	0: No limit switch has switched 1: A notify switch has switched (see Object 0x2311 for which input has switched)

Bit 10 (Target Reached) is set if the drive has reached its target position in Profile Position Mode or has reached its target velocity in Profile Velocity Mode. Specification of a new target value deletes the bit.

Bit 11 (Internal Limit active) indicates that a internal range limit has been reached.

Bit 12 (Setpoint acknowledge/Speed) is set after receiving a new positioning command (control word with new setpoint) and is reset when the target position is reached or the new setpoint has been reset in the control word (handshake for positioning command). In Profile Velocity Mode the bit is set at velocity 0.

6 Functional description of the CANopen CiA 402

6.1 Device Control

6.1.2 Selection of the operating mode

The Modes of Operation parameter is used to select the active drive profile, the Modes of Operation Display entry can be used to read back the current mode of operation.

Modes of Operation (0x6060)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6060	0	modes of operation	Integer8	wo	1	Operating mode changeover

FAULHABER Motion Control systems support the following operating modes:

1 CiA 402 Profile Position Mode (position control)

3 CiA 402 Profile Velocity Mode (velocity control)

6 CiA 402 Homing Mode (homing)

-1 FAULHABER specific operating mode

The operating modes according to CiA 402 are described in the following sections. The operating modes of the FAULHABER-specific mode of operation are described in Chapter 3 “Operation in FAULHABER mode”.

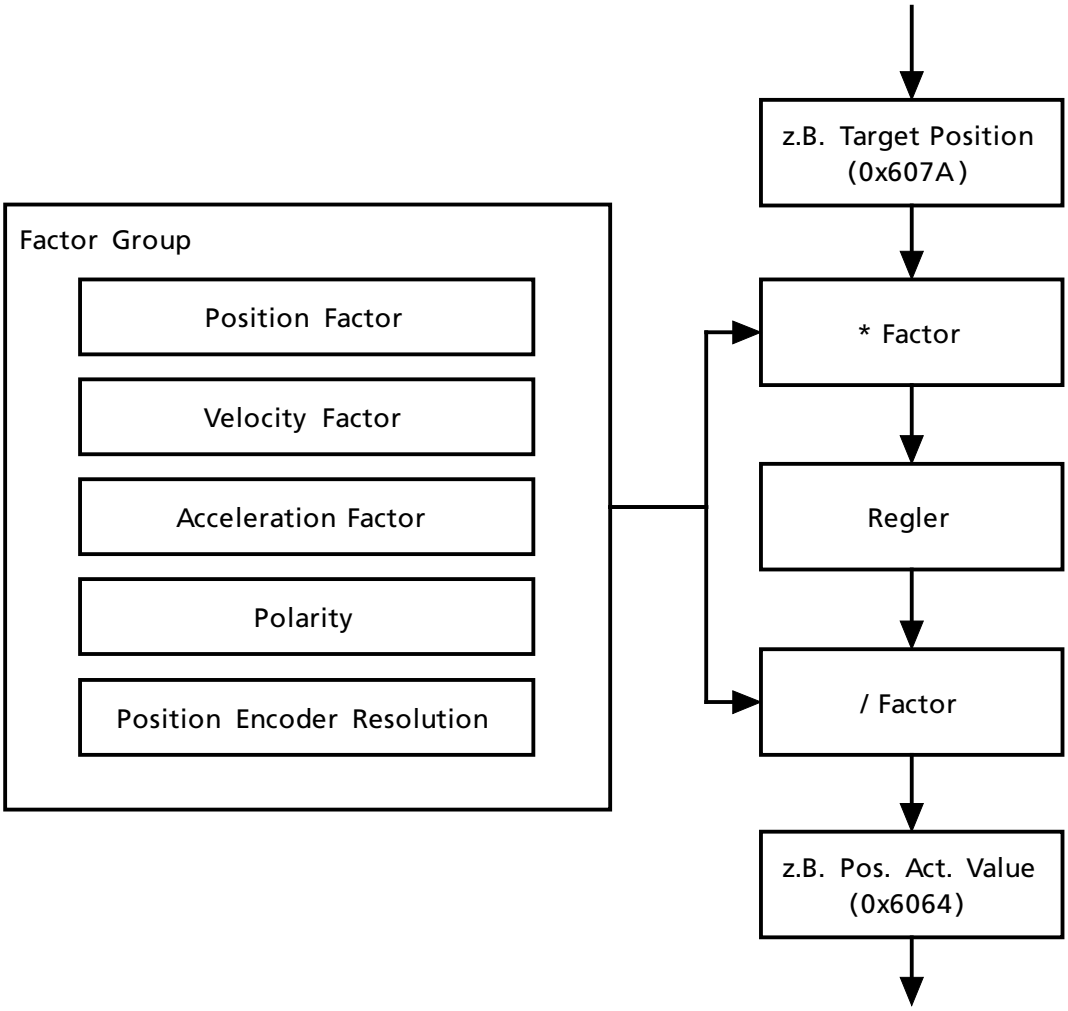
Modes of Operation Display (0x6061)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6061	0	modes of operation display	Integer8	ro	1	Display of the set operating mode

The set operating mode can be queried here, the meaning of the return values corresponds to the values of the object 0x6060.

6.2 Factor Group

Effects of the factor group on the set-point and actual values of the controller



The objects of this area are used to convert between internal variables and user-defined physical variables. The effective factors are each determined via a quotient:

$$\text{Factor} = \frac{\text{Numerator}}{\text{Denominator (divisor)}}$$

FAULHABER Motion Control systems support the conversion of the position, the velocity and acceleration at the interface in user-defined variables.

6 Functional description of the CANopen CiA 402

6.2 Factor Group

Position Factor (0x6093)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6093	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	numerator	Unsigned32	rw	1	Numerator of the position factor
	2	feed_constant	Unsigned32	rw	1	Denominator (divisor) of the position factor

The position factor can be used to set the required position unit for the profile position mode. Internally, the encoder resolution or the resolution of the analog hall signals of BL motors without encoder are used.

Velocity Factor (0x6096)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6096	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	numerator	Unsigned32	rw	1	Numerator of the velocity factor
	2	divisor	Unsigned32	rw	1	Denominator (divisor) of the velocity factor

The required velocity unit can be set using the velocity factor. Internally, the velocity is used in 1/min (rpm).

Acceleration Factor (0x6097)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6097	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	numerator	Unsigned32	rw	1	Numerator of the acceleration factor
	2	divisor	Unsigned32	rw	1	Denominator (Divisor) of the acceleration factor

The required acceleration unit can be set using the acceleration factor. Internally, accelerations are displayed in $1/s^2$.

Polarity (0x607E)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x607E	0	polarity	Unsigned8	rw	0	Direction of rotation

The entries in this object can be used to change the direction of rotation of the connected encoder for the supported operating modes:

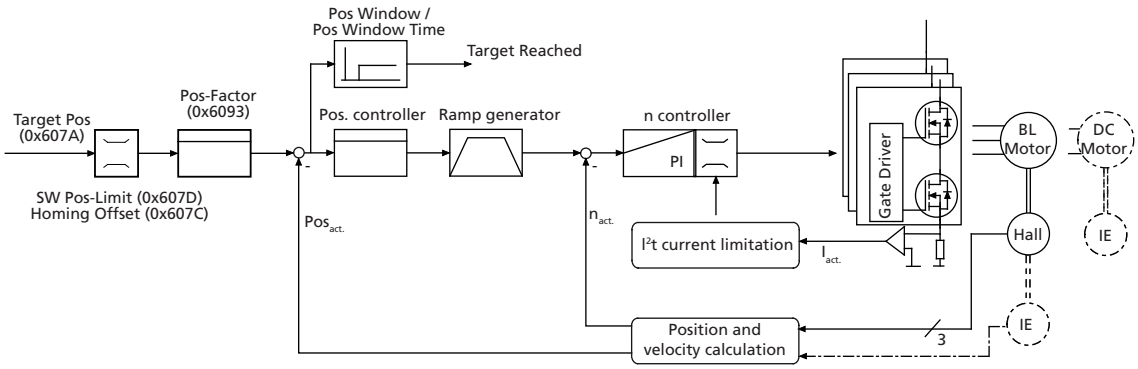
Bit 7 = 1 ➔ negative direction of rotation in positioning mode

Bit 6 = 1 ➔ negative direction of rotation in velocity mode

6 Functional description of the CANopen CiA 402

6.3 Profile Position Mode and Position Control Function

Controller structure for position control in Profile Position Mode



In this operating mode the target position and the controller settings are specified by the entries in the object dictionary.

Operating mode overview

In profile position mode the drive is positioned in the transferred target position.

In order for the drive to be operated in profile position mode, this operating mode must be set in the modes of operation parameter (0x6060). In addition, the drive must be in operation enabled state via its state machine.

In general, after switching on a homing sequence must be performed via homing mode in order to reset the position value to zero at the homing limit switch (see Chapter 6.4 “Homing Mode”).

A position setpoint value is specified via the target position object (0x607A). The positioning process is started by a change from 0 to 1 of bit 4 (New setpoint in the controlword). Bit 6 in the controlword can be used to additionally specify whether the setpoint value is to be interpreted in absolute or relative terms.

Operation in profile position mode requires correctly set velocity and position controllers.

In addition to the setpoint value, the software position limit object (0x607D) can be used to specify range limits for the movement range.

The set maximum values for acceleration, deceleration ramp and speed are additionally taken into account.

Notification of the higher level control

Attainment of the target position is signalled by bit 10 “target reached” in the statusword of the drive. If the transmission type for the particular PDO is set to 255, the PDO is transmitted asynchronously, triggered by the change in state.

Basic settings

The position control parameter set object (0x2332) can be used to set the proportional amplification and a differential term for the position controller.

Positioning range limits can be defined relative to the reference position using the software position limit object (0x607D).

The position window object (0x6067) can be used to define a window around the target position. The target position is signalled as being reached using bit 10 (target reached) in the statusword, if the actual position stays within the position window for at least the time set in the position window time object (0x6068).

6 Functional description of the CANopen CiA 402

6.3 Profile Position Mode and Position Control Function

Software Position Limit (0x607D)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x607D	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	min position limit	Integer32	rw	**$-1.8 \cdot 10^9$**	Lower positioning range limit
	2	max position limit	Integer32	rw	**$+1.8 \cdot 10^9$**	Upper positioning range limit

The positioning range limits are specified in the units defined by the user and are converted in the internal display using the position factor.

Position Control ParameterSet (0x60FB)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x60FB	0	number of entries	Unsigned16	ro	2	Number of object entries
	1	gain	Unsigned16	rw	*)	Position controller P term
	2	D constant	Unsigned16	rw	*)	Position controller D term

*) Dependent on the factory configuration of the motion controller

Position Window (0x6067)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6067	0	position window	Unsigned32	rw	40	Target position window

Symmetrical area around the target position, which is used for the “Target Reached” message. It is specified in user-defined units, according to the given Position Factor.

Position Window Time (0x6068)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6068	0	position window time	Unsigned32	rw	200	Time in target position window

If the drive stays within the range of the Position Window for at least the time set here in milliseconds, bit 10 is set in the statusword (Target Reached).

Query current values / Position Control Function

The last target position can be read back in internal units using the Position Demand Value object on Index 0x60FC and in user-defined units using the entry on Index 0x6062.

The current position can be read back in internal units using the Position Actual Value object on Index 0x6063 and in user-defined units using Index 0x6064. The description of the objects is given in Chapter 8.3 “Drive profile objects according to CiA 402”.

Additional settings

Incremental encoder as position sensor

By default, the position for BL motors is evaluated using the analog hall sensors with a resolution of 3 000 increments per revolution. Alternatively, an incremental encoder can also be used as the position encoder for BL motors in Profile Position mode. To do this, the drive must be configured in ENCMOD. This can be done in FAULHABER mode using the FAULHABER commands on PDO2 or using the Motion Manager. Following the initial configuration, the Profile Position Mode can be selected.

Ramp generator

The output of the position controller is additionally limited by a ramp generator to the permissible acceleration and deceleration values and the maximum speed.

A trapezoidal profile with linear speed ramps only is supported. This setting can be read out in the Motion Profile Type object (0x6086).

Profile Velocity (0x6081) and Max Profile Velocity (0x607F)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6081	0	profile velocity	Unsigned32	rw	*)	Maximum speed
0x607F	0	max profile velocity	Unsigned32	rw	*)	Maximum speed

*) Dependent on the factory configuration of the motion controller

Maximum velocity during positioning. It is specified in user-defined units, according to the given velocity factor. Both objects describe the same internal parameter.

Profile Acceleration (0x6083)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6083	0	profile acceleration	Unsigned32	rw	30 000.	Maximum acceleration

It is specified in user-defined units, according to the given acceleration factor.

Profile Deceleration (0x6084)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6084	0	profile deceleration	Unsigned32	rw	30 000.	Maximum delay

Quick Stop Decelaration (0x6085)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6085	0	quick stop deceleration	Unsigned32	rw	30 000	Braking ramp value for Quick Stop

The acceleration values are specified in user-defined units, according to the value of the given acceleration factor.

Velocity controller / current limitation

The controller parameters of the secondary velocity controller can also be adjusted. In addition, the current limitation values LPC and LCC made available by the FAULHABER channel (PDO2) or the Motion Manager can be used to protect the drive against overload (see Chapter 6.5 “Profile Velocity Mode”).

6 Functional description of the CANopen CiA 402

6.3 Profile Position Mode and Position Control Function

Motion control commands

A position set-point is specified using the Target Position object (0x607A). The positioning process is started using bit 4 in the controlword. Bit 6 in the controlword can also be used to additionally specify whether the set-point should be interpreted as being absolute or relative.

Target Position (0x607A)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x607A	0	target position	Integer32	rw	0	Target position

The target position is specified in the units defined by the user and is converted in the internal display using the position factor.

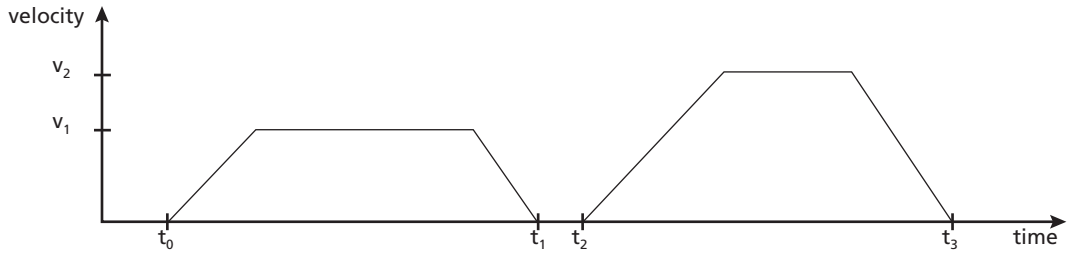
Adoption of a new target position is acknowledged by the drive via the statusword with set bit 12 (acknowledge setpoint). The drive signals that the target position has been reached via the statusvalue with set bit 10 (target reached). "Target Reached" remains set until new positioning is started or the output stage is switched off.

If a new setpoint value is specified during positioning (new setpoint), this is accepted immediately and the drive moves to the new target position. In this way, motion profiles can be run through continuously without having to decelerate the drive to velocity 0 in between times.

Individual positioning sequence:

Prerequisite: NMT state "Operational", drive state "Operation Enabled" and modes of operation (0x6060) set to Profile Position Mode (1).

1. Set Target Position (0x607A) to the required value.
2. In the Controlword, set Bit 4 (New Setpoint) to "1" and set Bit 6 (abs / rel) depending on whether absolute or relative positioning is required.
3. The drive responds with Bit 12 (Setpoint Acknowledge) set in the Statusword and starts the positioning.
4. The drive signals that the target position has been reached via the status value with set bit 10 (target reached). A new positioning job can now be started (New Setpoint).



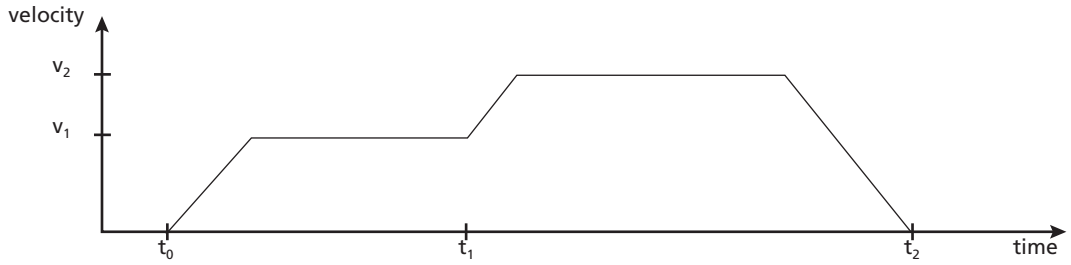
6 Functional description of the CANopen CiA 402

6.3 Profile Position Mode and Position Control Function

Procedure for a sequence of set-points:

Prerequisite: NMT state “Operational”, drive state “Operation Enabled” and modes of operation (0x6060) set to Profile Position Mode (1).

1. Set Target Position (0x607A) to the required value.
2. In the Controlword, set Bit 4 (New Setpoint) to “1” and set Bit 6 (abs / rel) depending on whether absolute or relative positioning is required.
3. The drive responds with Bit 12 (Setpoint Acknowledge) set in the Statusword and starts the positioning.
4. A new positioning job can already be started now (New Setpoint), with relative positioning the new target position is added to the last target position. The drive then moves immediately to the new target position.
5. The end of the movement sequence is signalled by the Statusword with set Bit 10 (Target reached).



6 Functional description of the CANopen CiA 402

6.4 Homing Mode

The objects within this range are available for homing mode. In general, after switching on a homing sequence must be performed to reset the position value at the homing limit switch. Object 0x2310 can be used to set which inputs are to be used as homing limit switches (see Chapter 8.2 "Manufacturer-specific objects").

Homing Offset

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x607C	0	homing offset	Integer32	rw	0	Zero point displacement from the reference position

Homing Method

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6098	0	homing method	Integer8	rw	0	Homing method

All homing methods defined in DSP402 V2 are supported:

1 to 14: Homing with index pulse (if available)

17 to 30: Homing without index pulse

33, 34: Homing at index pulse (if available)

35: Homing at the current position

NOTE

Limit switches limit the movement range (negative/positive limit switch), but at the same time can also be used as reference switches for the zero position. A homing switch is a separate reference switch for the zero position.

Method 1 and 17

Homing at the lower limit switch (Negative Limit Switch)

If the limit switch is inactive, the drive moves in the direction of the lower limit switch first, until its positive edge has been detected. If the limit switch is active, the drive moves upward out of the limit switch until the negative edge has been detected. With Method 1, the drive then continues moving on the next index pulse at which the home position is set.

Method 2 and 18

Homing at the upper limit switch (Positive Limit Switch)

If the limit switch is inactive, the drive moves in the direction of the upper limit switch first, until its positive edge has been detected. If the limit switch is active, the drive moves downward out of the limit switch until the negative edge has been detected. With Method 2, the drive then continues moving on the next index pulse at which the home position is set.

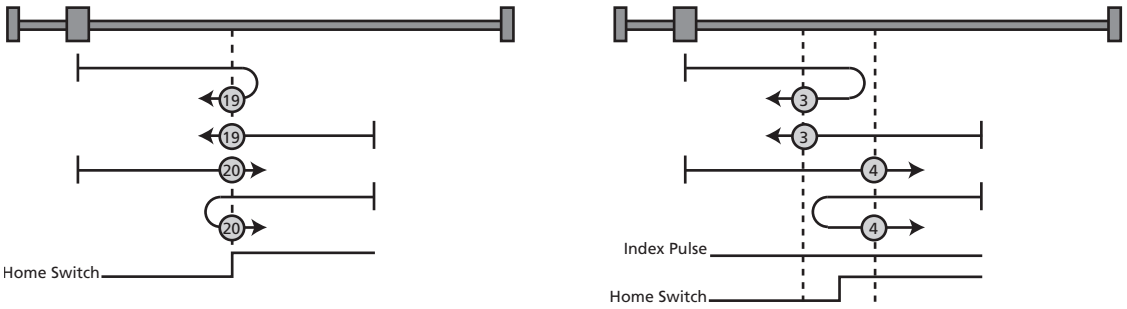
6 Functional description of the CANopen CiA 402

6.4 Homing Mode

Method 3, 4 and 19, 20

Homing at a positive Homing switch (Positive Home Switch)

Depending on the state of the homing switch, the drive moves in one direction or the other up to the falling (3, 19) or rising (4, 20) edge. There is only one rising edge of the homing switch in the direction of the upper limit switch. The FAULHABER HP parameter for the limit switch used is simultaneously set to 1 here (rising edge).



Method 5, 6 and 21, 22

Homing at a negative Homing switch (Negative Home Switch)

Depending on the state of the homing switch, the drive moves in one direction or the other up to the falling (5, 21) or rising (6, 22) edge. There is only one falling edge of the homing switch in the direction of the upper limit switch. The FAULHABER HP parameter for the limit switch used is simultaneously set to 0 here (falling edge).

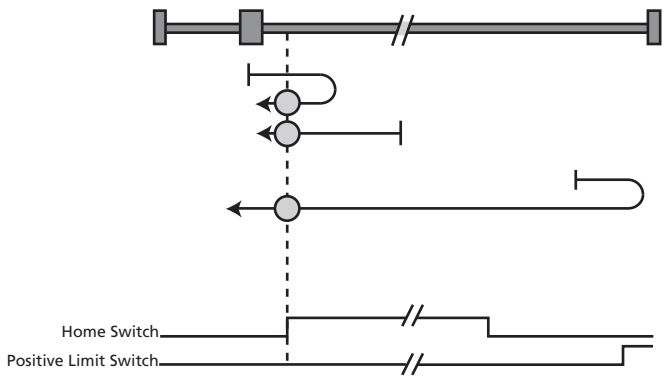
Method 7 to 14 and 23 to 30

Homing at Homing switch (Home Switch)

These methods use a limit switch which is only active within a defined path range. A differentiation is made between the response to the two edges.

With Methods 7 to 14, after the edge has been detected, the drive continues moving up to the index pulse at which the homing position is then set.

Method 7 and 23 Homing at bottom of falling edge.
Start in positive direction, if switch inactive



Method 8 and 24 Homing at bottom of rising edge.
Start in positive direction, if switch inactive.

6 Functional description of the CANopen CiA 402

6.4 Homing Mode

Method 9 and 25	Homing at top of rising edge. Always start in positive direction.
Method 10 and 26	Homing at top of falling edge. Always start in positive direction.
Method 11 and 27	Homing at top of falling edge. Start in negative direction, if switch inactive.
Method 12 and 28	Homing at top of rising edge. Start in negative direction, if switch inactive.
Method 13 and 29	Homing at bottom of rising edge. Always start in negative direction.
Method 14 and 30	Homing at bottom of falling edge. Always start in negative direction.
Method 33 and 34	Homing at the index pulse Drive moves in negative (33) or positive (34) direction up to the index pulse.
Method 35	The position counter is reset at the current position.

NOTE

Limit switches and homing switches are approached in velocity mode, an index pulse in positioning mode. At the same time the set range limits (0x607D) are taken into account.

6 Functional description of the CANopen CiA 402

6.4 Homing Mode

Homing Speed

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6099	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	speed during search for switch	Unsigned32	rw	400	Speed during search for switch
	2	speed during search for home	Unsigned32	rw	100	Speed during search for zero

The specifications are given in user-defined units, according to the given velocity factor.

Homing Acceleration

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x609A	0	homing acceleration	Unsigned32	rw	50	Acceleration during homing

It is specified in user-defined units, according to the given acceleration factor.

Procedure for a homing sequence:

Prerequisite: NMT state “Operational”, drive state “Operation Enabled” and modes of operation (0x6060) set to Homing Mode (6).

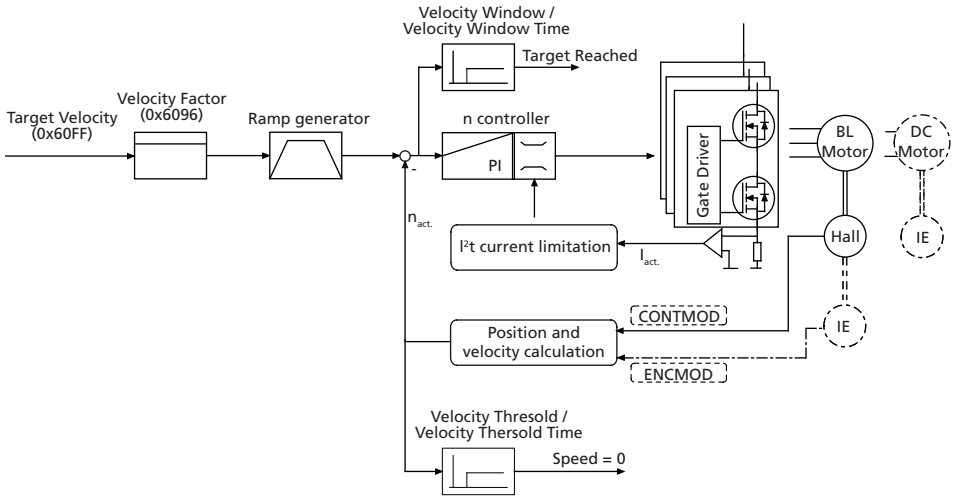
1. Set Homing Limit Switch (0x2310), Homing Method (0x6098), Homing Speed (0x6099) and Homing Acceleration (0x609A) to the required value.
2. In the controlword, set Bit 4 (Homing operation start) to “1”, to start the homing sequence.
3. The drive responds with Bit 12 (Homing attained) set in the statusword when the homing sequence is finished. If an error occurs during the homing sequence, Bit 13 (Homing Error) is set in the statusword.

An on-going homing sequence can be interrupted by writing a “0” on Bit 4 in the controlword.

6 Functional description of the CANopen CiA 402

6.5 Profile Velocity Mode

Controller structure in Profile Velocity Mode



Operating mode overview

In the profile velocity mode the speed of the drive is controlled by a PI controller. This ensures that the drive is operated without deviation from the specified values, provided it is not overloaded.

In order for the drive to be operated in profile velocity mode, this operating mode must be set in the modes of operation parameter (0x6060). In addition, the drive must be in operation enabled state via its state machine.

The target velocity is set via the target velocity object (0x60FF) in the object dictionary. In profile velocity mode the drive directly follows each new transferred setpoint value. At the same time, the set maximum values for acceleration, deceleration ramp and speed are also taken into account.

Notification of the higher level control

Attainment of the target velocity is signalled by bit 10 “target reached” in the statusword of the drive. A stopped drive is signalled via bit 12 “Speed = 0”. If the transmission type for the particular PDO is set to 255, the PDO is transmitted asynchronously, triggered by the change in state.

Operation in profile velocity mode requires a velocity controller correctly adjusted to the application.

6.5 Profile Velocity Mode

Basic settings

The Velocity Control Parameter Set object (0x60F9) can be used to set the proportional amplification and the I term for the position controller.

Velocity Control Parameter Set (0x60F9)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x60F9	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	gain	Unsigned16	rw	*)	P term
	2	integration time constant	Unsigned16	rw	*)	I term

*) Dependent on the factory configuration of the motion controller

The sampling rate can be set between 1 and 20 as a multiple of the internal sampling rate using the SR command. The internal sampling rate is 0.2 ms.

Actual velocity value

In BL motors the current velocity is determined by evaluating the analog hall sensor signals. The ECN-MOD can also be configured for BL motors using the Motion Manager or the ENCMOD command via the FAULHABER channel on PDO2.

In DC motors, the velocity is determined using the incremental encoder.

The resolution of the encoder is configured using the Position Encoder Resolution object (0x608F) (see Chapter 6.2 “Factor Group”).

Additional settings

Ramp generator

After specifying a new target speed using the Target Velocity object (0x60FF), the drive is accelerated or braked to the new speed in the Profile Velocity Mode using the acceleration deposited in the Profile Acceleration object (0x6083). The parameter is valid in both directions!

Current limitation

The parameter LPC (allowable peak current) and LCC (allowable continuous current) can be used to protect the drive against overload. The parameters can be set using the configuration dialogue of the Motion Manager or the FAULHABER commands LPC and LCC via PDO2.

Motion control commands

A velocity set-point is specified using the Target Velocity object (0x60FF). Provided the drive is in Operation Enable state (see Chapter 6.1 “Device Control”), the drive is accelerated directly to the new target velocity.

The Parameter Velocity Window (0x606D) is used to define a window around the target velocity, within which the target velocity is signalled as being reached, if the velocity remains within the target window for at least the time in using the parameter Velocity Window Time (0x606E).

The attained target velocity is signalled in the statusword by bit 10 “Target Reached”.

The Parameter Velocity Threshold (0x606F) is used to define a threshold value for the velocity, below which the drive is signalled as being at a standstill, if the velocity remains below the threshold value for at least the time defined using the parameter and Velocity Threshold Time (0x6070).

Stoppage is signalled in the statusword by bit 12 “Speed=0”.

Target Velocity (0x60FF)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x60FF	0	target velocity	Integer32	rw	20	Target velocity

The target velocity is specified in the units defined by the user and is converted in the internal display (1/min) using the velocity factor.

The last set target velocity can be queried in user-defined units using the Velocity Demand Value object (0x606B).

The current velocity value can be queried using the Velocity sensor actual value (0x6069) or Velocity actual value (0x606C) objects, each in user-defined units.

The description of the objects is given in Chapter 8.3 “Drive profile objects according to CiA 402”.

Complex motion profiles

Evaluation of bits 10 “Target Reached” and 12 “Speed = 0” in the statusword can be used to deliberately shut down specific velocity profiles. The acceleration is defined using the Profile Acceleration object.

6 Functional description of the CANopen CiA 402

6.6 Drive parameters / Common entries

Basic properties of the drive system are stored in the encoder and motor type objects.

Drive Data (0x6510)

The velocity constant and the motor resistance are required as parameters for the motor monitoring models. These values are already set for integrated units. These values are suitably preassigned for external controls by selecting a motor type in the Motion Manager's Motor Wizard.

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6510	0	number of entries	Unsigned8	ro	3	Number of object entries
	1	motor type	Signed32	ro	*)	Set motor type 0 BL motor –1 DC motor
	2	speed constant KN	Unsigned16	rw	*)	Speed constant Kn of the motor Unit: rpm/V
	3	motor resistance RM	Unsigned32	rw	*)	Motor resistance RM Unit: mΩ

*) Dependent on the factory configuration of the motion controller

Position Encoder Resolution (0x608F)

If the linear Hall sensors of the brushless motors are used as position transducers, 3 000 pulses per revolution are supplied. If using an incremental sensor as position sensor, its resolution must be set using the Position Encoder Resolution object.

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x608F	0	number of entries	Unsigned8	ro	2	Number of entries
	1	encoder increments	Unsigned32	rw	2 048	Resolution of the external encoder for 4 edge evaluation
	2	motor revolution	Unsigned32	rw	1	Number of motor revolutions with the pulse number named in subindex 1

6 Functional description of the CANopen CiA 402

6.7 Inputs / Outputs

The connections described in Chapter 3.3 “Homing and limit switches” are available. Chapter 3.3.1 “Limit switch connections and switching level”” describes configuration of the switching level. Chapter 3.5 “Special fault output functions” describes the special function of the fault pin.

Limit switch and homing switch setting

The available digital inputs can each be configured as limit switches or homing switches for use within a DSP402 homing method. The upper and lower limit switches are additionally used as range limit switches, beyond which the drive cannot move (hard blocking).

If lower and upper limit switches are not used for a DSP402 homing method, their switch polarity can be defined using the switch polarity parameter (rising or falling edge valid). By default, homing methods 1, 2, 17 and 18 assume a positively switching limit switch. If, on the other hand, a negative switching limit switch is to be used the required polarity must be set here accordingly and in addition the polarity parameter for the homing limit must be set to 1.

NOTE

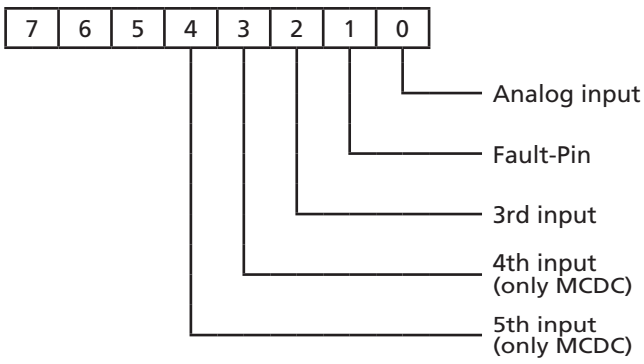
The input configuration cannot be changed in homing mode. For this you must switch to profile position or profile velocity mode!

Limit switch setting

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x2310	0	Number of entries	Unsigned8	ro	6	Number of object entries
	1	Negative limit switch	Unsigned8	rw	0	Lower limit switch
	2	Positive limit switch	Unsigned8	rw	0	Upper limit switch
	3	Homing switch	Unsigned8	rw	0x07/0x1F *)	Homing switch
	4	Notify switch	Unsigned8	rw	0	Notify switch
	5	Switch polarity	Unsigned8	rw	0x07/0x1F *)	Polarity of the switches 1: Pos. edge valid 0: Neg. Edge valid
	6	Polarity for homing limit	Unsigned8	rw	0	Use the polarity of the switches in DSP402 Homing Mode too

*) BL-Controller / MCDC

The function of the digital inputs can be set here in accordance with the following bit mask:



Explanation

- Subindex 1 (negative limit):

  Here the input is given, at which the lower limit switch for homing methods 1 and 17 or for a hard blocking function is connected.

  If the limit switch is activated the drive is stopped and can now only be moved back out of the limit switch in the opposite direction (Hard Blocking).

- Subindex 2 (positive limit):

  Here the input is given, at which the upper limit switch for homing methods 2 and 18 or for a hard blocking function is connected.

  If the limit switch is activated the drive is stopped and can now only be moved back out of the limit switch in the opposite direction (Hard Blocking).

- Subindex 3 (homing):

  Here the input is given, at which the homing switch for homing methods 3 to 14 and 19 to 30 is connected. Polarity and notify (subindex 5 and 4) cannot be used here.

- Subindex 4 (notify):

  Here the input is given, at which a notify switch is connected, which reports with the status word and set bit 14, if it has been activated. Object index 0x2311 can then be used to query which switch has switched. This function cannot be used simultaneously with a homing method. The object's setting is not saved, as soon as a switch has been activated the value is reset.

- Subindex 5 (polarity):

  The polarity of the notify switch and the hard blocking limit switch can be set here. If the polarity is to be changed with homing methods 1, 2, 17 and 18 too, subindex 6 must be set to 1 beforehand.

- Subindex 6 (DSP402 polarity):

  Here it is possible to give whether the polarity settings under subindex 5 are to be used for the homing methods 1, 2, 17 and 18. In general, the setting can only be set for all inputs (no bitmask coding).

For a description of the homing methods, see Chapter 6.4 "Homing Mode".

The settings of this object simultaneously change the settings of the FAULHABER parameters HB, HD, HA, HN and HP!

Notify switch

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x2311	0	triggered switch	Unsigned8	ro	0	Switched switches

This object can be used to query which switch switched according to the above bit mask if a status-word message with set bit 14 has arrived. Reading the object resets bit 14 in the statusword.

6.8 Error handling

FAULHABER Fault Register

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x2320	0	number of entries	Unsigned8	ro	4	Number of object entries
	1	internal fault register	Unsigned16	ro	0	Current internal fault 0 = No fault
	2	emergency mask	Unsigned16	rw	0x00FF	Faults which trigger an emergency message frame
	3	fault mask	Unsigned16	rw	0	Faults which are treated as DSP402 faults and affect the state machine (fault state)
	4	errout mask	Unsigned16	rw	0x00FF	Faults which set the error output

This object describes how internal faults are treated.

The errors are coded as follows and can be masked by adding the required error types:

Error-Bit	Error	Beschreibung
0x0001	Continuous over current	Set continuous current limiting exceeded
0x0002	Deviation	Set maximum allowable velocity deviation exceeded
0x0004	Over voltage	Overvoltage detected
0x0008	Over temperature	Maximum coil or MOSFET temperature exceeded
0x0010	Flash memory error	Memory error
0x0100	Life guard or heartbeat error	CAN monitoring error
0x0200	Recovered from bus off	Exit CAN bus error “Bus off”
0x1000	Internal software	Internal software error

NOTE

Set SubIndex 3 of Object 0x2320 to 1, to switch off the drive in the event of overcurrent and place it in error status. A value of 0x0101 switches the drive off even if a CAN error exists.

NOTE

Set SubIndex 4 of Object 0x2320 to 0, if the error output (Fault Pin) is not to display errors or to 0xFFFF, if all errors (including CAN errors) are to be displayed.

See also Chapter 3.5 “Special fault output functions” for further information on the error types and Chapter 4.4 “Emergency Object (error message)” for the coding of the emergency error codes.

Set Baud Rate

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x2400	0	Baud rate	Unsigned8	ro	0xFF	Set baud rate

This object can be used to query which baud rate is set. The index of the set baud rate is returned or 0xFF, if AutoBaud is set.

Baud rate	Index
1 000 kBit	0
800 kBit	1
500 kBit	2
250 kBit	3

Baud rate	Index
125 kBit	4
50 kBit	6
20 kBit	7
10 kBit	8
AutoBaud	0xFF

7 Commissioning

Guide

Node number and baud rate	Page 98
Basic settings	Page 100
Configuration using the Motion Manager	Page 101
Configuration in FAULHABER mode	Page 105
Configuration in a drive profile according to CIA 402	Page 113
Data set management	Page 120
Status display	Page 121

The drive unit must be connected to a PC via a CAN adapter or a host control with CANopen interface in order to make the basic settings for commissioning.

NOTE

Connection of the CAN interface is described in the technical manual. For the communication setup, ensure that the same transfer rate is set for all nodes (see Chapter 2.1 "Set node number and baud rate") and the terminating resistances are used!

FAULHABER Motion Manager provides a convenient device configuration option using graphic dialogues.

The configuration can also be carried out using your own programming or other CANopen configuration tools.

7.1 Node number and baud rate

The node address and transfer rate are set using the network in accordance with the LSS protocol according to CiA DSP305 V1.1 (Layer Setting Services and Protocol).

A configuration tool which supports the LSS protocol is therefore required for the setting, e.g. FAULHABER Motion Manager.

The configuration tool is the LSS Master, and the drives act as LSS Slaves.

LSS Slaves can be configured in two ways:

1. "Switch Mode Global" switches all connected LSS Slaves to configuration mode. However, only one LSS Slave may be connected to set the baud rate and Node-ID.
2. "Switch Mode Selective" switches precisely one LSS Slave in the network to configuration mode. For this, the Vendor ID, Product code, Revision number and Serial number of the node to be addressed must be known.

7 Commissioning

7.1 Node number and baud rate

The following baud rates (Bit Timing Parameters) can be set:

Baud rate	Index
1 000 kBit/s	0
800 kBit/s	1
500 kBit/s	2
250 kBit/s	3
125 kBit/s	4
50 kBit/s	6
20 kBit/s	7
10 kBit/s	8

In addition, Index 0xFF can be used to activate automatic baud rate detection.

The following node numbers can be set:

1 – 127.

Node-ID 255 (0xFF) indicates that the node has not yet been configured, which causes the node to retain in LSS-Init status after it is switched on until a valid node number is transferred to it. Only then is the NMT initialisation continued.

The LSS protocol also supports the reading of of LSS addresses, consisting of the Vendor ID, Product code, Revision Number and Serial number of connected units and reading out of the set Node-ID.

Identifiers 0x7E5 (from the Master) and 0x7E4 (from the Slave), on which the protocol is worked through, are used for LSS communication.

Following configuration, the set parameters are backed up in the Flash memory, so that they are available again after switching off and on.

FAULHABER controllers use Vendor ID, Product code and Serial number only to activate the “Switch Mode Selective”. 0.0 can always be transferred for the revision number, as this value is ignored in the protocol.

Vendor ID: 327

Product code: 3 150

Please refer to the CiA document DSP 305 for a detailed description of the LSS protocol.

If automatic baud rate detection is activated, the drive can be used in a network with any transmission rate in accordance with the table above and after 3 message frames on the bus line at the latests, the baud rate of the network is detected and the drive has adjusted itself to it. Here it must be noted that the initial message frames cannot be processed and booting takes a little longer.

7 Commissioning

7.2 Basic settings

In the case of external motion controllers, several basic settings have to be made during the initial start-up to adjust the controller to the connected motor.

If drive units are integrated, these basic settings are made in the factory so it is only necessary to adjust to the respective application.

CAUTION! ***Risk of destruction!***

Failure to observe these basic settings can result in destruction of components!

► *The basic settings described in the following must be noted and observed*

The following basic settings must be made for external motion controllers:

- Motor type or motor data (KN, RM) of the connected motor
- Resolution of an external encoder (ENCRES), if used
- Current limitation values (LCC, LPC), adjusted to the motor type and application
- Controller parameters (POR, I, PP, PD), adjusted to the motor type and application

In addition, FAULHABER Motion Manager can be used to synchronise the Hall sensor signals for smooth start-up and optimisation of the phase angle for the best efficiency.

The configuration must then be adjusted to the respective application for all motion controllers (integrated and external). In particular, the following basic settings are important:

- Operating Mode
- Current limiting values
- Controller parameters
- Function of the digital inputs/outputs

Warning! ***Risk of destruction***

If using the Fault Pin as input (REFIN, DIRIN), the desired function must be programmed before applying external voltage!

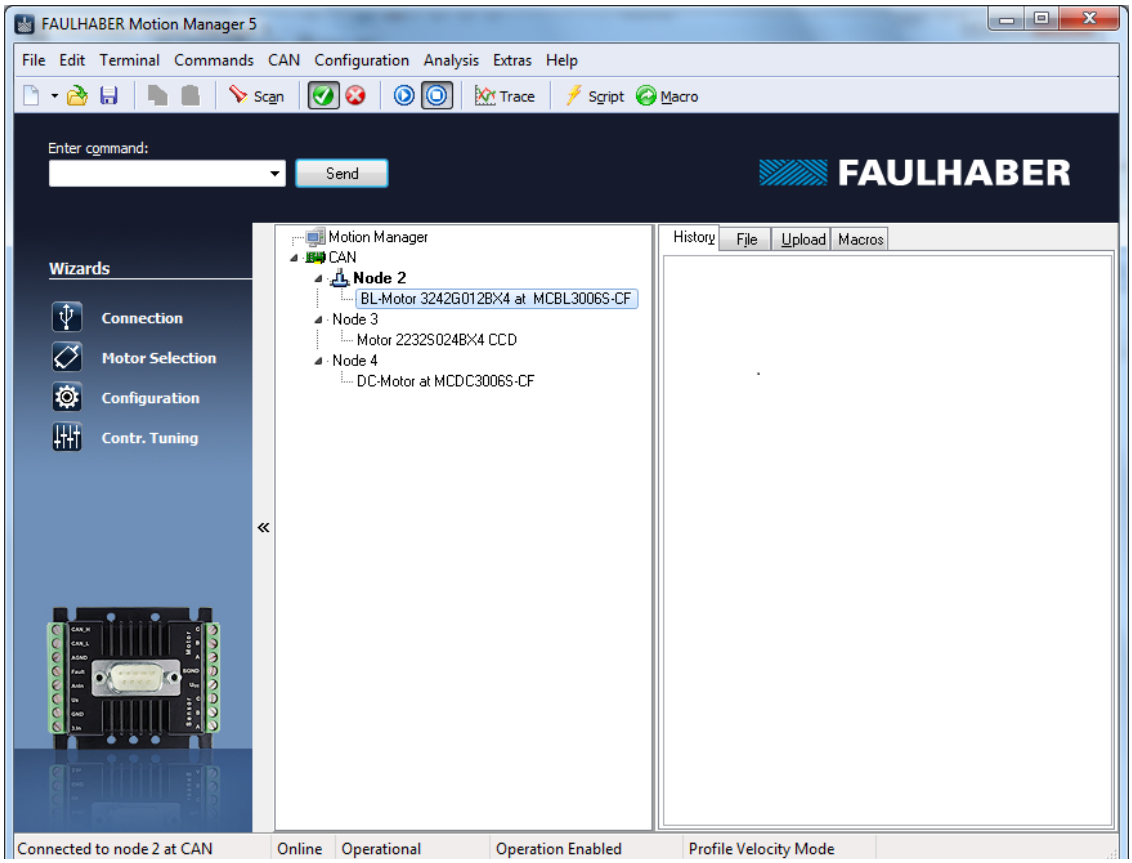
Configuration of these parameters with the help of the FAULHABER Motion Manager is explained in greater detail in the following chapter.

7 Commissioning

7.3 Configuration using the Motion Manager

"FAULHABER Motion Manager" PC software provides a simple option for configuring the drive unit and for performing initial tests and optimisation.

The software is available for Microsoft Windows and can be downloaded free of charge from the FAULHABER internet site: www.faulhaber.com.



Motion control systems with electronics built onto the motor are already pre-parameterised in the factory.

Motion controllers with an externally connected motor must be equipped with current limitation values suitable for the motor and suitable controller parameters before being started up.

The motor selection Wizard is available for selecting the motor and the suitable basic parameters.

Other settings, e.g. for the function of the fault pin, can be made under the "configuration – drive functions" menu item, where a convenient dialog is provided (Chapter 7.3.3 "Drive configuration"). The configuration dialog is also available for direct access in the wizard bar of the Motion Manager.

A tuning wizard, with which the controller parameters of the speed and positioning controller can be adjusted to the application, is also provided.

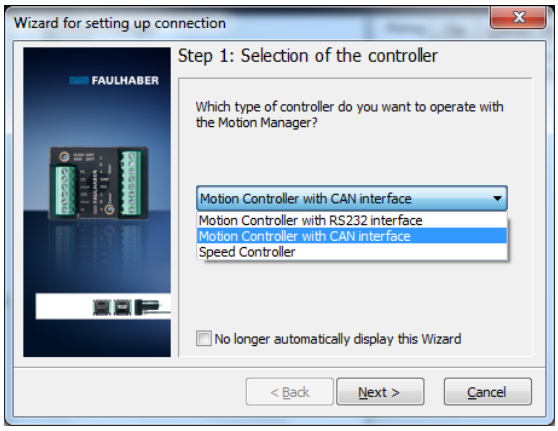
7 Commissioning

7.3 Configuration using the Motion Manager

7.3.1 Connection setting

If no drive nodes are found when the Motion Manager is started, a connection wizard appears. In the first step, the “Motion Controller with CAN-interface” product group must be selected. The connection wizard can also be started at any time via the wizard bar.

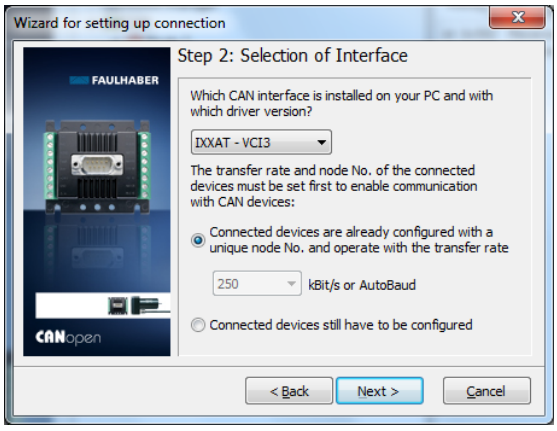
Connection wizard (Step 1: Selection of the controller)



In the second step, the CAN interface used and, if applicable, the baud rate can be set. Information on the supported CAN interfaces is given in the instruction manual of the Motion Manager or you can contact FAULHABER for information.

The interface found by the driver must then be explicitly adopted again as a once-off action.

Connection wizard (Step 2: Selection of Interface)



Devices which are already set to a baud rate are then found by the Motion Manager and are displayed in the Node Explorer.

Devices which have not yet been configured can be assigned a node number and baud rate in a further step.

7 Commissioning

7.3 Configuration using the Motion Manager

7.3.2 Motor selection

External motion controllers must be adjusted to the connected motor.

The Motor Wizard is provided for this purpose; it can be opened via the Wizard bar of the Motion Manager.

After selecting the required FAULHABER motor from a list and setting the sensor type used, as well as entering an inertia factor for the load to be operated, in addition to the motor and current limiting values, suitable controller parameters are also determined and transferred to the drive.

Refer to the Motion Manager instruction manual for details of how to use the Motor Wizard.

7.3.3 Drive configuration

The Motor Wizard has already set sensible default settings for the motor/sensor combination selected.

A configuration dialog with several pages for further drive configuration and adjustment to the required application is available in the Motion Manager's Wizard bar or under the menu item: "Configuration – Drive functions... ".

NOTE

The CAN node must be in OPERATIONAL state (see Chapter 4 "CANopen protocol description") in order to configure the drives.

The node can be started using the context menu in the NodeExplorer: "CANopen Network Management - Start Remote Node".

No settings are transferred to the drive until the "Send" button is pressed. The current state of the drive is also read back and the dialog is updated accordingly. Invalid combinations of settings are corrected at the same time, as they are not accepted by the drive.

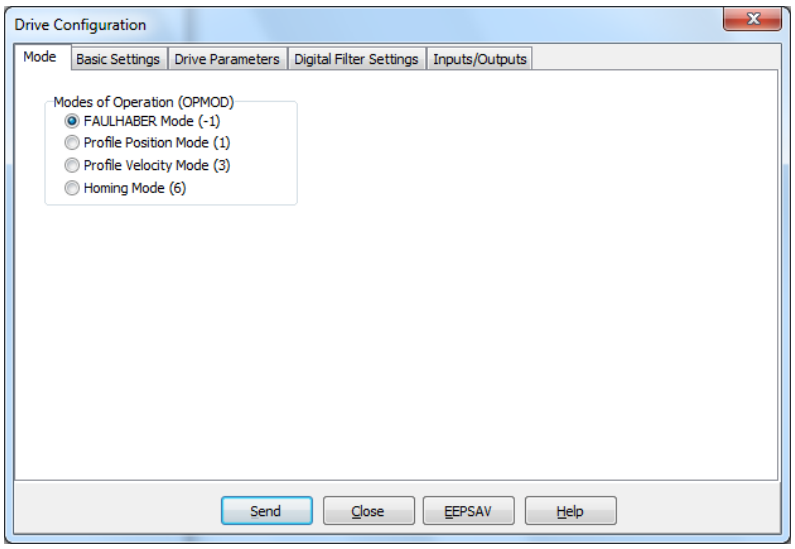
The settings are permanently saved in the drive using the "EEPSAV" button.

7 Commissioning

7.3 Configuration using the Motion Manager

7.3.4 Selection of the operating mode

Configuration of the Modes of Operation



The basic operating mode must be selected using the “Modes of Operation” or OPMOD on the first page of the configuration dialogue. FAULHABER Mode with the scope of functions described in Chapter 3 “Operation in FAULHABER mode” and the drive profiles described in the Chapter 6 “Functional description of the CANopen CiA 402” are available.

7 Commissioning

7.4 Configuration in FAULHABER mode

7.4.1 Basic settings

Within the scope of the commissioning,

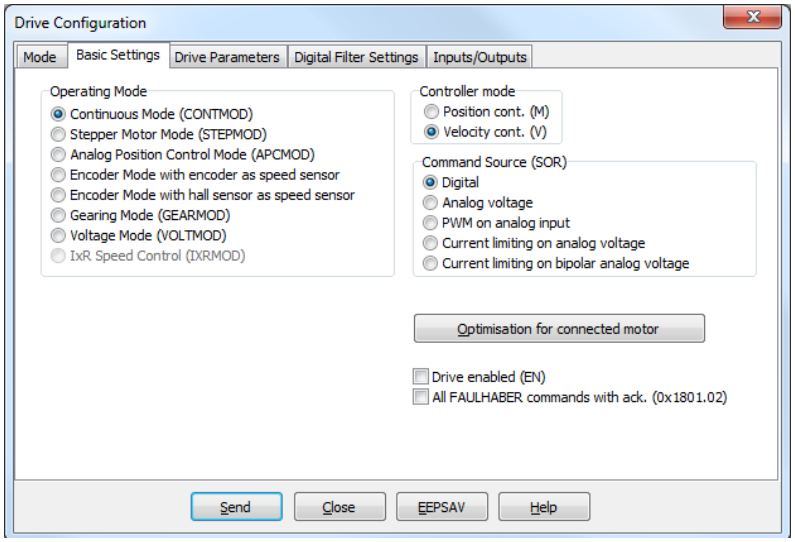
- the operating mode
- and the type of set-point presetting (command source)

are set in the Basic Settings tab:

NOTE

The"Basic settings" dialog page is only shown in FAULHABER mode.

Basic settings for the motor and encoder type



7 Commissioning

7.4 Configuration in FAULHABER mode

Encoder type and optimisation

If an incremental encoder attached to the motor is to be evaluated its effective resolution must be given for 4 edge evaluation. If using the internal encoder, no further inputs are necessary.

A button, with which the Optimisation Wizard can be started, is available for adjusting Hall sensor signals and phase angles to the connected motor for externally connected BL motors with analog Hall sensors.

NOTE

Ensure that the motor can freely rotate before starting the encoder optimisation.

Controller mode

FAULHABER motion controllers support both main types of operation

- Position control as servo drive.
- Velocity control

The controller mode is partly automatically selected depending on the chosen operating mode.

Operating mode

In addition to the controller mode, variations of the operation can also be selected.

The following options are available:

CONTMOD

Default setting for the selected controller mode.

For BL motors the actual velocity and actual position in CONTMOD are determined by the motor's Hall sensors.

For DC motors the actual velocity and actual position are determined by the motor's incremental encoder (corresponds to ENCMOD)

CONTMOD for position control: see Chapter 3.1.1 "Set-point presetting via CAN/PDO2"

CONTMOD for velocity control: see Chapter 3.2.1 "Target velocity via CAN/PDO2"

STEPMOD

Position control

The target position is derived from the number of steps at the AnIn input.

STEPMOD see Chapter 3.4.1 "Stepper motor mode"

APCMOD

Position control

The target position is preset by an analog voltage at the AnIn input.

APCMOD see Chapter 3.1.2 "Analog positioning mode (APCMOD)"

ENCMOD with ENCSPEED

Position control with evaluation of an external encoder in BL motors, including for the actual speed.

ENCMOD for position control: see Chapter 3.1.3 "External encoder as actual position value (ENCMOD) - not for MCDC"

7 Commissioning

7.4 Configuration in FAULHABER mode

ENCMOD with HALLSPEED

Position control with evaluation of an external encoder and the Hall signals for the actual speed of BL motors

ENCMOD for position control: see Chapter 3.1.3 “External encoder as actual position value (ENCMOD) - not for MCDC”

GEARMOD

Position control

The target position is determined using the number of steps of an external encoder

GEARMOD see Chapter 3.4.2 “Gearing mode (electronic gear)”

VOLTMOD

Direct presetting of a voltage amplitude at the motor

VOLTMOD see Chapter 3.4.3 “Voltage regulator mode”

IxRMOD

Velocity control without sensors for DC motors

IxRMOD see Chapter 3.4.5 “IxR control for MCDC”

Set-point presetting

The set-value presetting must be chosen to match the selected type of operation and controller mode.

The following are supported:

- Set-point presetting via CAN
- Set-point presetting for position or velocity via an analog voltage
- Set-point presetting for position or velocity via a PWM voltage
- Set-point presetting for the limit current via an analog voltage

Power-on state

In the default state the drive’s power stage is initially inactive after power-on.

The power stage can be activated by selecting the “Drive enabled (EN)” checkbox.

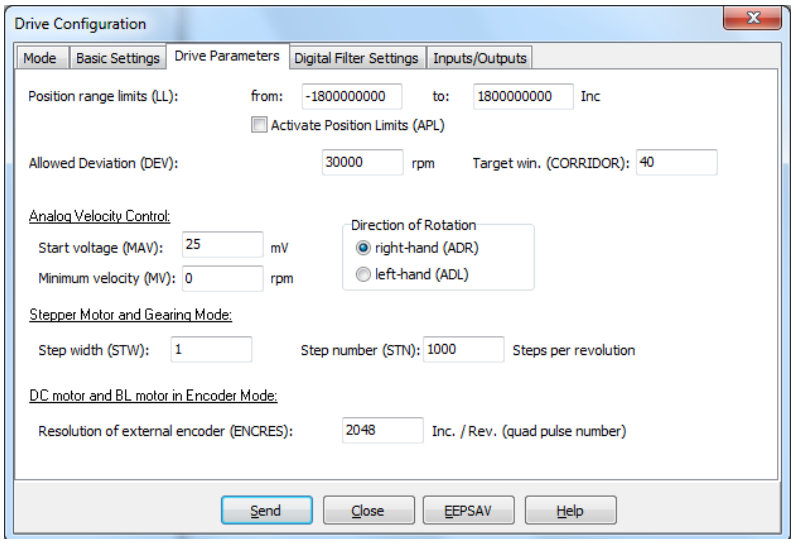
7 Commissioning

7.4 Configuration in FAULHABER mode

7.4.2 Drive parameters

The Drive Parameters tab is used to make additional settings for the encoder and chosen type of operation.

Additional settings for the chosen type of operation



Encoder resolution

If an incremental encoder attached to the motor is to be evaluated its effective resolution for 4 edge evaluation must be given.

Set-point presetting in stepper or gearing mode

For set-point presetting in stepper mode and in gearing mode the conversion from step count of the external presetting to number of motor revolutions must be given.

Example:

Motor has to perform one revolution at 1 000 pulses of the external encoder or at 1 000 steps:

- `STW1`
- `STN1000`

Detailed notes on using these parameters are given in the chapters with the functional description of stepper and gearing mode (Chapter 3.4.1 “Stepper motor mode” and Chapter 3.4.2 “Gearing mode (electronic gear)”).

Velocity presetting via an analog voltage

For presetting a velocity via an analog voltage, a threshold value (MAV) can be preset, from which the target value is evaluated starting with the minimum velocity (MV).

Detailed notes on using these parameters are given in the Chapter 3.2.2 “Velocity presetting via an analog voltage or a PWM signal”.

7 Commissioning

7.4 Configuration in FAULHABER mode

Positioning range limits

In various types of operation the movement range can be monitored and limited. The limits of this movement range can be given in increments of the actual position using the parameter LL

Range monitoring is activated by the APL1 command.

Maximum allowable velocity deviation and target corridor

The parameter CORRIDOR defines a range around the target position within which the “Target position reached” flag is set. If required, the target position is signalled asynchronously by a notify.

Within this corridor the D term of the position controller is active and the ramp generator is inactive.

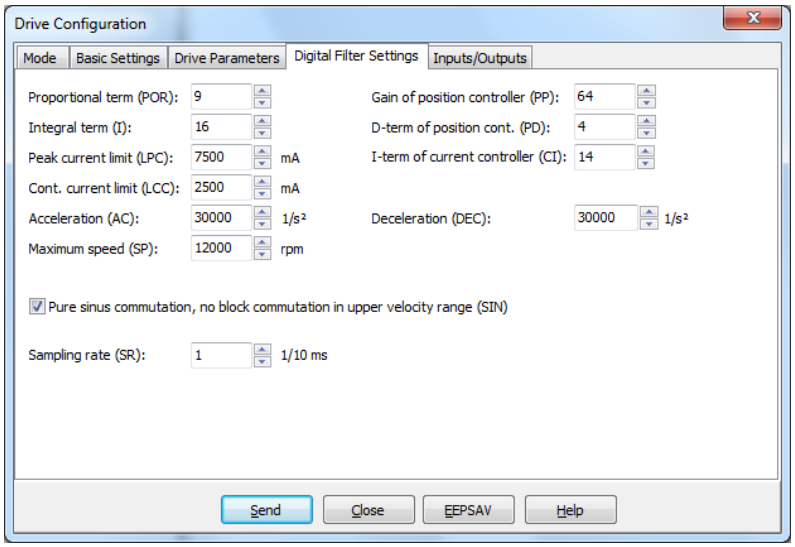
The parameter DEV can be used to preset a maximum allowable controller deviation for the velocity controller. If this barrier is exceeded for longer than set using the parameter DCE in the Inputs and Outputs tab, an error is signalled via the fault pin or via a CANopen Emergency Message.

7.4.3 Controller settings

The changes to the default set controller and current limitation parameters can be made in the “Controller Parameters” tab of the drive configuration dialog.

In addition, under the “Configuration – Digital Filter Settings…” menu item, there is another dialog in which the parameters can be changed online and the result can be observed directly or can be recorded using the trace function in Motion Manager.

Digital Filter Settings



7 Commissioning

7.4 Configuration in FAULHABER mode

Voltage output

By default the Motion Controller for BL motors uses pure sinus commutation. This means the motor runs with the lowest possible losses and noise.

Alternatively, at higher velocities it is possible to also allow overriding of the output signals similar to block commutation. As a result, the whole velocity range of the drive can be used.

NOTE

On changing between pure sinus commutation and operation with block commutation in the upper velocity range the controller amplification is also increased accordingly.

Current controller (LCC, LPC, CI)

The parameter LCC can be used to give the thermally allowable continuous current for the application.

Motors and the motion controller can be overloaded within certain limits. Therefore, higher currents can also be allowed for dynamic processes. The maximum current value is given using the parameter LPC.

Depending on the drive's load, the internal current monitoring limits the output current to the peak current (LPC) or the allowable continuous current (LCC).

CAUTION!

Risk of destruction!

The thermally allowable continuous current (LCC) should never be given above the thermally allowable continuous current of the motor according to the data sheet.

The maximum peak current (LPC) may never be given above the maximum peak output current of the installed electronics.

The current controller of the motion controller operates as a current limiting controller and therefore in an unlimited case has no effect on the dynamics of the velocity control. The speed of the limitation can be set using the parameter CI. If using the default values for your motor, the current is limited to the allowable value after around 5 ms.

If a FAULHABER motor was selected via the Motor Wizard, parameters are already set here with which the motor can be operated safely.

Further details are given in the Chapter 3.6.3 "Current controller and I²t current limitation".

Velocity controller (I, POR, SR)

The velocity controller is implemented as a PI controller. The sampling time SR can be set as multiples of the basic sampling rate, the proportional gain POR and the integral component I can be set.

If a FAULHABER motor was selected via the Motor Wizard, parameters are already set here with which the motor can be operated safely.

If the motor is exposed to additional loads, the inertia of the load must be compensated for by a higher proportional term and if necessary slower sampling; in most applications the integral term can remain unchanged.

Further information on setting and adjustment is given in Chapter 3.6.7 "Adjustment of the controller parameters".

7 Commissioning

7.4 Configuration in FAULHABER mode

Ramp generator (AC, DEC, SP)

The ramp generator limits the velocity change at the input of the velocity controller using the parameters AC and DEC and the maximum preset speed using the parameter SP.

The parameters AC and SP can be freely selected depending on the application; the parameter DEC is used to specify the deceleration behaviour in positioning mode. For large loads, the deceleration ramp must be limited using the parameter DEC to achieve dead beat (overshoot-free) run-in in the target position.

Further information on setting and adjustment is given in Chapter 3.6.1 “Ramp generator”.

Position controller (PP, PD)

The position controller is implemented as a proportional controller. An additional D term also acts within the target corridor only (see Drive Parameters tab).

The proportional term uses the position deviation in increments to calculate the maximum preset velocity for the underlying velocity controller. The ramp generator is used to additionally limit the acceleration and maximum velocity.

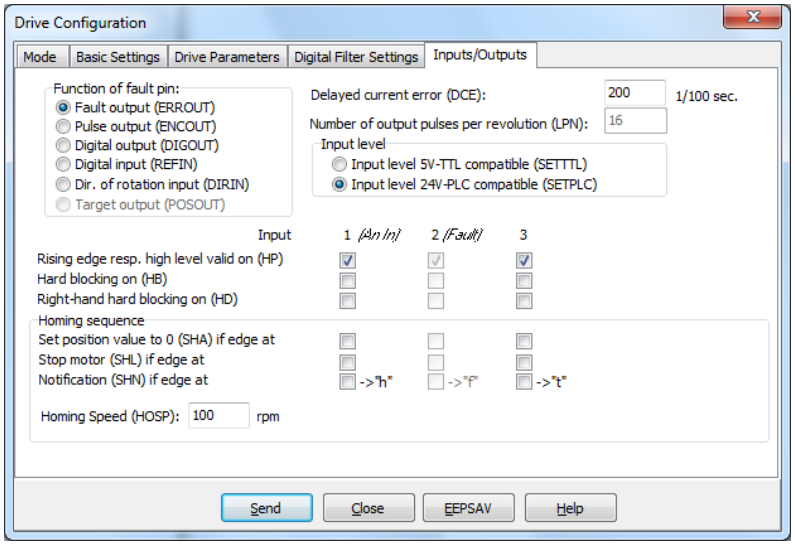
Dead beat run-in in the target position can be preferentially achieved by adjusting the deceleration ramp to the load. For a well-attenuated transient condition in the limit position, the parameter PP must be reduced proportionally to the load inertia.

Further information on setting and adjustment is given in Chapter 3.6.7 “Adjustment of the controller parameters”.

7.4.4 I/O connection and use

The functions of the digital inputs and outputs and homing can be defined in the “Inputs/Outputs” tab of the drive configuration dialog.

Configuration of the inputs and outputs



7 Commissioning

7.4 Configuration in FAULHABER mode

Input level and edge

The switching thresholds of the digital inputs are either directly 5V TTL compatible or are adjusted to the switching level of 24 V PLC outputs.

In addition, it is also possible to select which level is to be used as the active level for each input and if the input is to be used as a limit switch (HB/HD).

Function of the fault pin

The fault pin can be used both as an input and as an output.

CAUTION!

Do not connect 24V to the fault pin, if the fault pin is configured as a digital output (ERROUT / DIG-OUT / ENCOUT)!

The other settings for the 2nd input can only be made if the fault pin is configured as a reference input.

For the default function as a fault output, the parameter DCE can be used to specify a delay time to suppress the response to individual short overcurrent pulses.

For the function as pulse output, the number of pulses per revolution of the motor can be set using the parameter LPN.

In the POSOUT function the output displays the entry into the target corridor as a digital signal (low means target position is reached).

Homing

Use as a reference switch can be set for each of the available inputs.

To this end, either the actual position can be set to 0 by an edge at the selected input (SHA), the motor can be stopped (SHL) or a message can be set to the higher level control (SHN). Notification is given by sending the statusword with bit14 = 1 (Hard Notify) on PDO1. The actions can be combined.

Homing defined in this way can be executed using the GOHOSEQ command.

7 Commissioning

7.5 Configuration in a drive profile according to CIA 402

7.5.1 Basic settings

NOTE

The "Basic settings" dialog page is only shown in FAULHABER mode.

Encoder type and optimisation

If an incremental encoder attached to the motor is to be evaluated its effective resolution must be given for 4 edge evaluation. If using the internal encoder, no further inputs are necessary.

A button, with which the Optimisation Wizard can be started, is available for adjusting Hall sensor signals and phase angles to the connected motor for externally connected BL motors with analog Hall sensors.

NOTE

Ensure that the motor can freely rotate before starting the encoder optimisation.

7 Commissioning

7.5 Configuration in a drive profile according to CIA 402

Profile selection

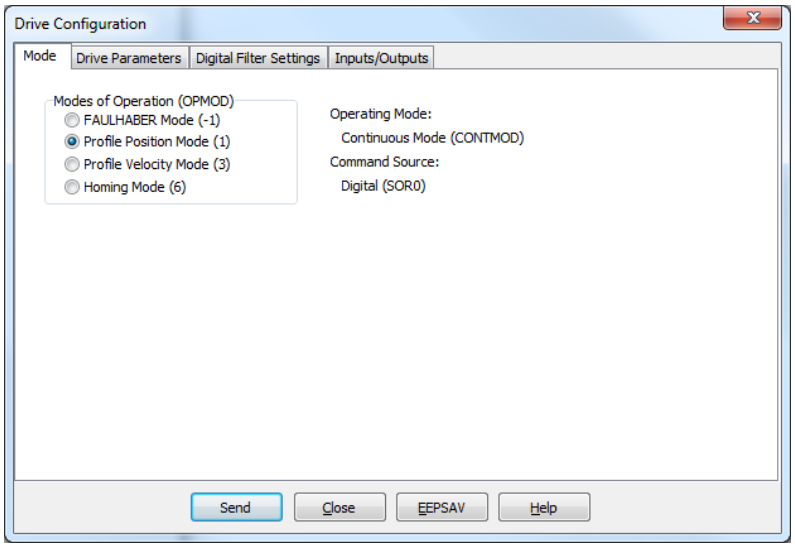
In the “Mode” tab of the drive configuration you can select one of the CiA 402 drive profiles under “Modes of Operation”.

If the drive is to be operated with an incremental encoder as position sensor, you can activate this in the selected profile by entering the ENCMOD command in the command line of the Motion Manager.

NOTE

Save the selected setting via EEPSAV to permanently configure the drive.

Selection of the drive profile



Controller mode

Operation is supported as a positioning drive, as a speed-controlled drive as well as the homing methods.

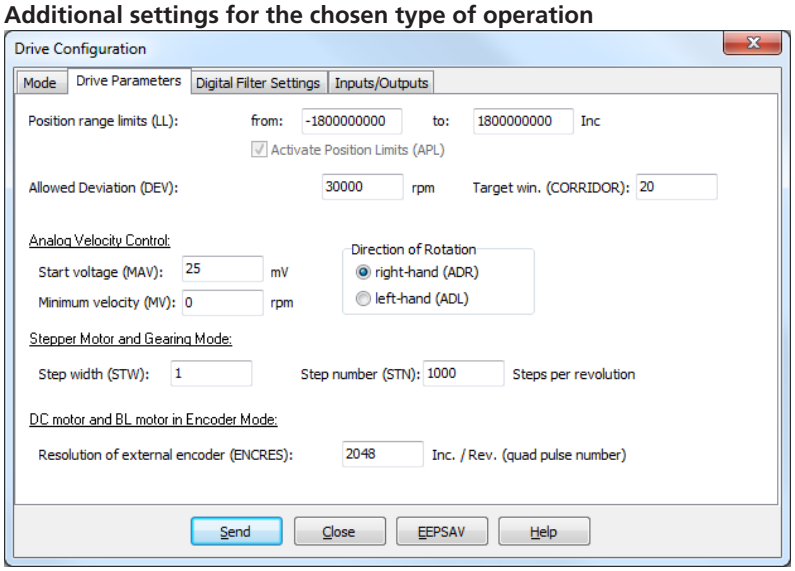
7 Commissioning

7.5 Configuration in a drive profile according to CIA 402

7.5.2 Drive parameters

The “Drive Parameters” tab is used to make additional settings for the encoder and chosen type of operation.

Additional settings for the chosen type of operation



Encoder resolution

If an incremental encoder attached to the motor is to be evaluated its effective resolution for 4 edge evaluation must be given.

Positioning range limits

In various types of operation the movement range can be monitored and limited. The limits of this movement range can be given in increments of the actual position using the parameter LL.

Maximum allowable velocity deviation and target corridor

The parameter CORRIDOR defines a range around the target position within which the “Target reached” flag is set in the statusword. If transmission type 255 is configured for the TxPDO1 (default setting), the target position is signalled by an asynchronously set PDO. Within this corridor the D term of the position controller is active and the ramp generator is inactive.

The parameter DEV can be used to preset a maximum allowable controller deviation for the velocity controller. If this barrier is exceeded for longer than set using the parameter DCE in the Inputs and Outputs tab, an error is signalled via the fault pin or via a CANopen Emergency Message.

7 Commissioning

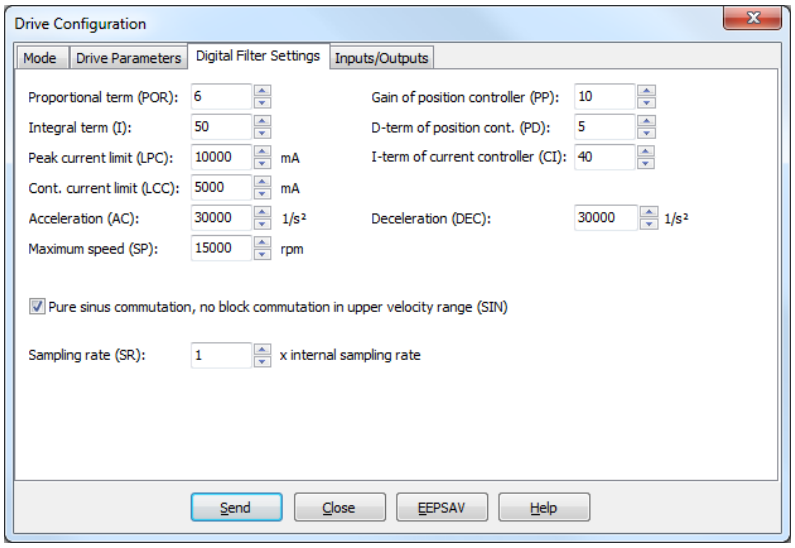
7.5 Configuration in a drive profile according to CIA 402

7.5.3 Controller setting

The changes to the default set controller and current limitation parameters can be made in the “Digital Filter Settings” tab of the drive configuration dialog.

In addition, under the “Configuration – Digital Filter Settings...” menu item, there is another dialog in which the parameters can be changed online and the result can be observed directly or can be recorded using die trace function in Motion Manager.

Digital Filter Settings



Voltage output

By default the motion controller uses pure sinus commutation. This means the motor runs with the lowest possible losses and noise.

Alternatively, at higher velocities it is possible to also allow overriding of the output signals similar to block commutation. As a result, the whole velocity range of the drive can be used.

NOTE

On changing between pure sinus commutation and operation with block commutation in the upper velocity range the controller amplification is also increased accordingly.

7 Commissioning

7.5 Configuration in a drive profile according to CIA 402

Current controller (LCC, LPC, CI)

The parameter LCC can be used to give the thermally allowable continuous current for the application.

Motors and the motion controller can be overloaded within certain limits. Therefore, higher currents can also be allowed for dynamic processes. The maximum current value is given using the parameter LPC.

Depending on the drive's load, the internal current monitoring limits the output current to the peak current (LPC) or the allowable continuous current (LCC).

CAUTION! ***Risk of destruction!***

The thermally allowable continuous current (LCC) should never be given above the thermally allowable continuous current of the motor according to the data sheet.

The maximum peak current (LPC) may never be given above the maximum peak output current of the installed electronics.

The current controller of the motion controller operates as a current limiting controller and therefore in an unlimited case has no effect on the dynamics of the velocity control. The speed of the limitation can be set using the parameter CI. If using the default values for your motor, the current is limited to the allowable value after around 5ms.

If a FAULHABER motor was selected on the basic settings page, parameters are already set here with which the motor can be safely operated.

Further details are given in the Chapter 3.6.3 "Current controller and I²t current limitation".

Velocity controller (I, POR, SR)

The velocity controller is implemented as a PI controller. The sampling time SR can be set as multiples of the drive's basic sampling rate, the proportional amplification POR and the integral term I.

If a FAULHABER motor was selected on the basic settings page, parameters are already set here with which the motor can be safely operated.

If the motor is exposed to additional loads, the inertia of the load must be compensated for by a higher proportional term and if necessary slower sampling; in most applications the integral term can remain unchanged.

Further information on setting and adjustment is given in Chapter 3.6.7 "Adjustment of the controller parameters".

Ramp generator (AC, DEC, SP)

The ramp generator limits the velocity change at the input of the velocity controller using the parameters AC and DEC and the maximum preset speed using the parameter SP.

The parameters AC and SP can be freely selected depending on the application; the parameter DEC is used to specify the deceleration behaviour in positioning mode. For large loads, the deceleration ramp must be limited using the parameter DEC to achieve dead beat (overshoot-free) run-in in the target position.

Further information on setting and adjustment is given in Chapter 3.6.1 "Ramp generator".

7 Commissioning

7.5 Configuration in a drive profile according to CIA 402

Position controller (PP, PD)

The position controller is implemented as a proportional controller. An additional D term also acts within the target corridor only (see Drive Parameters tab).

The proportional term uses the position deviation in increments to calculate the maximum preset velocity for the underlying velocity controller. The ramp generator is used to additionally limit the acceleration and maximum velocity.

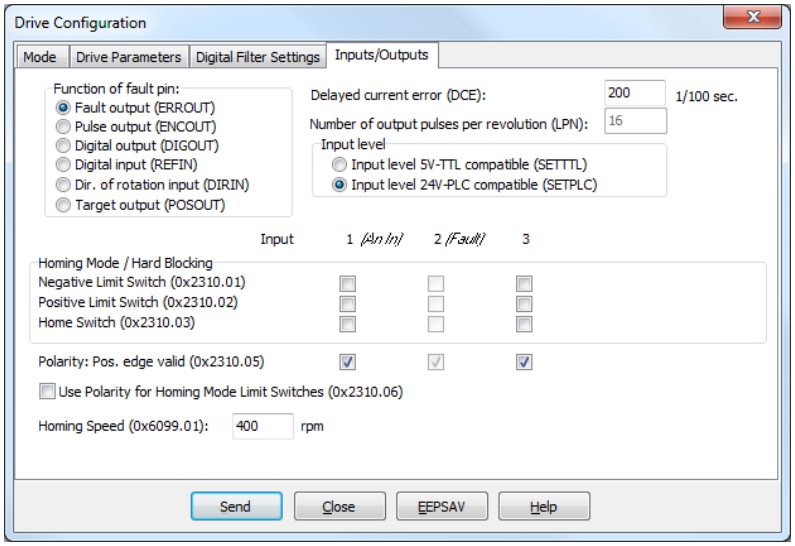
Dead beat run-in in the target position can be preferentially achieved by adjusting the deceleration ramp to the load. For a well-attenuated transient condition in the limit position, the parameter PP must be reduced proportionally to the load inertia.

Further information on setting and adjustment is given in Chapter 3.6.7 “Adjustment of the controller parameters”.

7.5.4 I/O connection and use

The function of the digital inputs and outputs can be defined in the “Inputs/Outputs” tab of the drive configuration dialogue.

Configuration of the inputs and outputs



Input level and edge

The switching thresholds of the digital inputs are either directly 5V TTL compatible or are adjusted to the switching level of 24 V PLC outputs.

Precise information on the thresholds is given in the drive’s data sheet.

7 Commissioning

7.5 Configuration in a drive profile according to CIA 402

Function of the fault pin

The fault pin can be used both as an input and as an output. The basic function can be selected using the radiobuttons.

The other settings for the 2nd input can only be made if the fault pin is configured as a reference input.

For the default function as a fault output, the parameter DCE can be used to specify a delay time to suppress the response to individual short overcurrent pulses.

For the function as pulse output, the number of pulses per revolution can be set using the parameter LPN.

7 Commissioning

7.6 Data set management

Save parameters

The settings of a drive can be saved as a backup or as a file for configuration of other drives.

The Motion Manager offers the option of reading out the current drive configuration and saving it as a parameter file.

Transfer parameters to the drive

Previously saved parameter files can be opened in Motion Manager, edited if necessary and transferred to the drive.

NOTE

Execute the SAVE or EEPSAV command to permanently save a transferred parameter set in the drive.

7 Commissioning

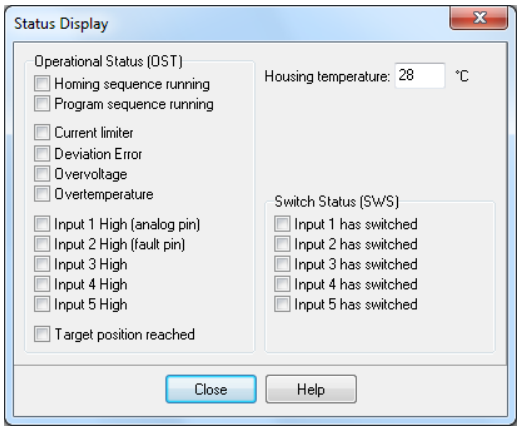
7.7 Status display

The status display is used for continuous checking of the main operating states.

Internal states, error flags and the state of the digital inputs are signalled. In addition, the internally measured housing temperature is also displayed here.

The display is updated by Motion Manager by means of cyclical querying of the internal states.

Display of the operating state



Internal states

Partially autonomous states of the motion controller are displayed. This is the course of homing.

Other internal states are on the one hand the error flag and the housing temperature.

The current limitation flag is set if the maximum current has been set to the continuous current (LCC) by the i²t monitoring.

States of digital inputs

The state of the digital inputs is displayed as On or Off depending on the level setting

Status of the limit switches

The display indicates whether one of the limit switches has switched, even if the assigned input is already back in the idle state.

7.7.1 Trace function

Motion Manager provides a trace function as an additional diagnosis tool with which the internal parameters can be graphically recorded. This enables the dynamic behaviour of the drive to be monitored, which is useful, e.g. for optimisation of the controller parameters.

8 Parameter description

8.1 Communication objects according to CiA 301

Device Type

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1000	0	device type	Unsigned32	ro	0x00020192	Specification of the device type

Contains information on the device type, divided into two 16-bit fields:

byte: MSB	LSB
Additional information	Device Profile Number

Device Profile Number = 0x192 (402d)

Error Register

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1001	0	error register	Unsigned8	ro		Error register

The error register contains, bit coded, the types of errors that have most recently occurred. For a description of the error register, see Chapter 4.4 “Emergency Object (error message)”.

Pre-defined Error Field (error memory)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1003	0	number of errors	Unsigned8	rw		Number of stored errors
	1	standard error field	Unsigned32	ro		Last error
	2	standard error field	Unsigned32	ro		Further errors…

The error memory contains the coding of the last error to occur. The standard error field is divided into two 16-bit fields:

byte: MSB	LSB
Additional information	Error code

The meaning of the individual error codes is described in Chapter 4.4 “Emergency Object (error message)”.

The error memory is deleted by writing “0” on subindex 0.

COB-ID SYNC

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1005	0	COB-ID SYNC	Unsigned32	rw	0x80	CAN object identifier of the SYNC object

8 Parameter description

8.1 Communication objects according to CiA 301

Manufacturer Device Name

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1008	0	manufacturer device name	Vis-String	const		Device name

Use the segmented SDO protocol to read out the device name, as it can be larger than 4 bytes.

Manufacturer Hardware Version

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1009	0	manufacturer hardware version	Vis-String	const		Hardware Version

Use the segmented SDO protocol to read out the hardware version, as it can be larger than 4 bytes.

Manufacturer Software Version

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x100A	0	manufacturer software version	Vis-String	const		Software Version

Use the segmented SDO protocol to read out the software version, as it can be larger than 4 bytes.

Guard Time

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x100C	0	guard time	Unsigned16	rw	0	Monitoring time for Node Guarding

Specification of the Guard Time in milliseconds, 0 switches off the monitoring.

Life Time Factor

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x100D	0	life time factor	Unsigned8	rw	0	Time factor for lifeguarding

The Life Time Factor multiplied by the Guard Time gives the Life Time for the Node Guarding Protocol (see Chapter 5.6 “NMT (Network Management)”). 0 switches off lifeguarding.

Store Parameters

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1010	0	number of entries	Unsigned8	ro	3	Number of object entries
	1	save all parameters	Unsigned32	rw	1	Saves all parameters
	2	save communication parameters	Unsigned32	rw	1	Save communication parameters only
	3	save application parameters	Unsigned32	rw	1	Save application parameters only

This object saves configuration parameters in the non-volatile Flash memory. Read access provides information about the storage options.

8 Parameter description

8.1 Communication objects according to CiA 301

The storage process is triggered by writing the “save” signature on the relevant subindex:

Signature	MSB			LSB
ISO 8859 ("ASCII")	e	v	a	s
hex	65h	76h	61h	73h

The object corresponds to the FAULHABER SAVE command.

CAUTION!

Flash memory

The Flash memory is designed for 10 000 write cycles. If this command is executed more than 10 000 times, the function of the Flash memory can no longer be guaranteed.

► *Do not execute command more than 10 000 times.*

Restore Default Parameters

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1011	0	number of entries	Unsigned8	ro	3	Number of object entries
	1	restore all default parameters	Unsigned32	rw	1	Loads all default parameters
	2	restore default communication parameters	Unsigned32	rw	1	Load default communication parameters only
	3	restore default application parameters	Unsigned32	rw	1	Load default application parameters only

This object loads the default configuration parameters (status on delivery).

Read access provides information about the restore options.

The restore process is triggered by writing the “load” signature on the relevant subindex:

Signature	MSB			LSB
ASCII	d	a	o	l
hex	64h	61h	6Fh	6Ch

The parameters are not set to the default values until the next boot process (reset).

If the default parameters are to be finally saved, a Save command must be executed after the reset.

COB-ID Emergency Message

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1014	0	COB-ID EMCY	Unsigned32	ro	0x80 + Node ID	CAN object identifier of the Emergency Object

8 Parameter description

8.1 Communication objects according to CiA 301

Identity Object

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1018	0	Number of entries	Unsigned8	ro	4	Number of object entries
	1	Vendor ID	Unsigned32	ro	327	Manufacturer ID number (FAULHABER: 327)
	2	Product code	Unsigned32	ro	3 150	Product ID number
	3	Revision number	Unsigned32	ro		Version number
	4	Serial number	Unsigned32	ro		Serial No.

Server SDO Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1200	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	COB-ID Client ➔ server (rx)	Unsigned32	ro	0x600 + Node ID	CAN object identifier of the server RxSDO
	2	COB-ID Server ➔ client (tx)	Unsigned32	ro	0x580 + Node ID	CAN object identifier of the server TxSDO

Receive PDO1 Communication Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1400	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	COB-ID	Unsigned32	ro	0x200 + Node ID	CAN object identifier of the server RxPDO1
	2	transmission type	Unsigned8	rw	255	PDO transmission type

Receive PDO2 Communication Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1401	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	COB-ID	Unsigned32	ro	0x300 + Node ID	CAN object identifier of the server RxPDO2
	2	transmission type	Unsigned8	rw	255	PDO transmission type

Receive PDO3 Communication Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1402	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	COB-ID	Unsigned32	ro	0x400 + Node ID	CAN object identifier of the server RxPDO3
	2	transmission type	Unsigned8	rw	255	PDO transmission type

Receive PDO1 Mapping Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1600	0	number of entries	Unsigned8	ro	1	Number of object entries
	1	1st object to be mapped	Unsigned32	ro	0x60400010	Reference to 16 bit controlword (0x6040)

8 Parameter description

8.1 Communication objects according to CiA 301

Receive PDO2 Mapping Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1601	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	1st object to be mapped	Unsigned32	ro	0x23010108	Reference to 8 bit FAULHABER command
	2	2nd object to be mapped	Unsigned32	ro	0x23010220	Reference to 32 bit command argument

Receive PDO3 Mapping Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1602	0	number of entries	Unsigned8	ro	5	Number of object entries
	1	1st object to be mapped	Unsigned32	ro	0x23030108	Reference to 8 bit trace mode for Parameter 1
	2	2nd object to be mapped	Unsigned32	ro	0x23030208	Reference to 8 bit trace mode for Parameter 2
	3	3rd object to be mapped	Unsigned32	ro	0x23030308	Reference to 8 bit trace time code setting
	4	4th object to be mapped	Unsigned32	ro	0x23030408	Reference to 8 bit trace value "number of packets"
	5	5th object to be mapped	Unsigned32	ro	0x23030508	Reference to 8 bit trace value "time interval"

Transmit PDO1 Communication Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1800	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	COB-ID	Unsigned32	ro	0x180 + Node ID	CAN object identifier of the TxPDO1
	2	transmission type	Unsigned8	rw	255	PDO transmission type Default: asynchronous

Transmit PDO2 Communication Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1801	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	COB-ID	Unsigned32	ro	0x280 + Node ID	CAN object identifier of the TxPDO2
	2	transmission type	Unsigned8	rw	253	PDO transmission type Default: asynchronous on request (RTR)

Transmit PDO3 Communication Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1802	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	COB-ID	Unsigned32	ro	0x380 + Node ID	CAN object identifier of the TxPDO3
	2	transmission type	Unsigned8	rw	253	PDO transmission type asynchronous on request (RTR) or synchronous

8 Parameter description

8.1 Communication objects according to CiA 301

Transmit PDO1 Mapping Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1A00	0	number of entries	Unsigned8	ro	1	Number of object entries
	1	1st object to be mapped	Unsigned32	ro	0x60410010	Reference to 16 bit statusword (0x6041)

Transmit PDO2 Mapping Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1A01	0	number of entries	Unsigned8	ro	3	Number of object entries
	1	1st object to be mapped	Unsigned32	ro	0x23010108	Reference to 8 bit FAULHABER command
	2	2nd object to be mapped	Unsigned32	ro	0x23020120	Reference to 32 bit value
	3	3rd object to be mapped	Unsigned8	ro	0x23020208	Reference to 8 bit error code

Transmit PDO3 Mapping Parameter

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x1A02	0	number of entries	Unsigned8	ro	3	Number of object entries
	1	1st object to be mapped	Unsigned32	ro	0x23040120	Reference to 32 bit trace value of Parameter 1
	2	2nd object to be mapped	Unsigned32	ro	0x23040220	Reference to 32 bit trace value of Parameter 2
	3	3rd object to be mapped	Unsigned32	ro	0x23040308	Reference to 8 bit timecode

8 Parameter description

8.2 Manufacturer-specific objects

FAULHABER Command

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x2301	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	command	Unsigned8	rw	0	Command byte for FAULHABER channel
	2	argument	Unsigned32	rw	0	Argument for FAULHABER command

This object is written by RxPDO2 and always contains the last FAULHABER command to be transmitted.

Return value of FAULHABER Command

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x2302	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	value	Unsigned32	ro	0	Argument for FAULHABER command
	2	error	Unsigned8	ro	0	Error code: 1 = OK, for other errors see

The content of this object is requested by a request (RTR) on TxPDO2 and delivers the return value for commands on the FAULHABER channel.

Trace Configuration

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x2303	0	number of entries	Unsigned8	ro	5	Number of object entries
	1	mode1	Unsigned8	rw	0	Trace mode for Parameter 1
	2	mode2	Unsigned8	rw	0	Trace mode for Parameter 2
	3	time code	Unsigned8	rw	1	Data with time code
	4	packets	Unsigned8	rw	1	Number of packets to be transmitted per request
	5	period	Unsigned8	rw	1	Time interval between packets

This object is written by RxPDO3 and always contains the last trace setting to be sent.

Trace Data

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x2304	0	number of entries	Unsigned8	ro	3	Number of object entries
	1	value1	Unsigned32	ro	0	Last value of Parameter 1
	2	value2	Unsigned32	ro	0	Last value of Parameter 2
	3	time code	Unsigned8	ro	0	Last time code value

The content of this object is requested by a request (RTR) on TxPDO3 and delivers the trace data of the set parameters. The values last requested are always temporarily stored here.

8 Parameter description

8.2 Manufacturer-specific objects

FAULHABER Fault Register

Index	Subindex	Name	Type	Attrb.	Default value	Meaning
0x2320	0	number of entries	Unsigned8	ro	4	Number of object entries
	1	internal fault register	Unsigned16	ro	0	Current internal fault 0 = No fault
	2	emergency mask	Unsigned16	rw	0x00FF	Faults which trigger an emergency message frame
	3	fault mask	Unsigned16	rw	0	Faults which are treated as DSP402 faults and affect the state machine (fault state)
	4	errout mask	Unsigned16	rw	0x00FF	Faults which set the error output

The error coding described in Chapter 6.8 “Error handling” applies to the FAULHABER error register and the error mask.

Set baud rate

Index	Subindex	Name	Type	Attrb.	Default value	Meaning
0x2400	0	Baud rate	Unsigned8	ro	0xFF	Set baud rate

This object can be used to query which baud rate is set. The index of the set baud rate is returned or 0xFF, if AutoBaud is set.

Baud rate	Index
1 000 kBit	0
800 kBit	1
500 kBit	2
250 kBit	3

Baud rate	Index
125 kBit	4
50 kBit	6
20 kBit	7
10 kBit	8
AutoBaud	0xFF

8 Parameter description

8.3 Drive profile objects according to CiA 402

Controlword (0x6040)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6040	0	controlword	Unsigned16	rw		Drive control

The bits in the controlword are described in Chapter 6.1 “Device Control”.

Statusword (0x6041)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6041	0	statusword	Unsigned16	ro		Status display

The bits in the statusword are described in Chapter 6.1 “Device Control”.

Modes of Operation (0x6060)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6060	0	modes of operation	Integer8	wo	1	Operating mode changeover

FAULHABER Motion Control systems support the following operating modes:

1 CiA 402 Profile Position Mode (position control)
3 CiA 402 Profile Velocity Mode (velocity control)
6 CiA 402 Homing Mode (homing)
-1 FAULHABER specific operating mode

Modes of Operation Display (0x6061)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6061	0	modes of operation display	Integer8	ro	1	Display of the set operating mode

The set operating mode can be queried here, the meaning of the return values corresponds to the values of the object 0x6060.

Position Demand Value (0x6062)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6062	0	position demand value	Integer32	ro		Last target position (scaled according to the position factor)

Position Actual Value (0x6063)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6063	0	position actual value	Integer32	ro		Actual position (increments)

8 Parameter description

8.3 Drive profile objects according to CiA 402

Position Actual Value (0x6064)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6064	0	position actual value	Integer32	ro		Actual position (scaled according to the position factor)

Position Window (0x6067)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6067	0	position window	Unsigned32	rw	20	Target position window (scaled according to the position factor)

Position Window Time (0x6068)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6068	0	position window time	Unsigned16	rw	200	Time in target position window in ms

Velocity Sensor Actual Value (0x6069)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6069	0	velocity sensor actual value	Integer32	ro		Actual velocity (scaled according to the encoder resolution)

Velocity Demand Value (0x606B)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x606B	0	velocity demand value	Integer32	ro		Target velocity (scaled according to the velocity factor)

Velocity Actual Value (0x606C)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x606C	0	velocity actual value	Integer32	ro		Actual velocity (scaled according to the velocity factor)

Velocity Window (0x606D)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x606D	0	velocity window	Unsigned16	rw	20	End velocity window (scaled according to the velocity factor)

Velocity Window Time (0x606E)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x606E	0	velocity window time	Unsigned16	rw	200	Time in end velocity window

8 Parameter description

8.3 Drive profile objects according to CiA 402

Velocity Threshold (0x606F)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x606F	0	velocity threshold	Unsigned16	rw	20	Velocity threshold value (scaled according to the velocity factor)

Velocity Thresold Time (0x6070)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6070	0	velocity threshold time	Unsigned16	rw	20	Time below the velocity threshold value in ms

Target Position (0x607A)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x607A	0	Target position	Integer32	rw		Target position (scaled according to the position factor)

Homing Offset (0x607C)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x607C	0	homing offset	Integer32	rw	0	Zero point displacement from the reference position (scaled according to the position factor)

Software Position Limit (0x607D)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x607D	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	min position limit	Integer32	rw	$\mathbf{-1.8 \cdot 10^9}$	Lower positioning range limit
	2	max position limit	Integer32	rw	$\mathbf{+1.8 \cdot 10^9}$	Upper positioning range limit

Each scaled according to the position factor.

Polarity (0x607E)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x607E	0	polarity	Unsigned8	rw	0	Direction of rotation

The entries in this object can be used to change the direction of rotation of the connected encoder for the supported operating modes:

Bit 7 = 1 ➔ negative direction of rotation in positioning mode

Bit 6 = 1 ➔ negative direction of rotation in velocity mode

8 Parameter description

8.3 Drive profile objects according to CiA 402

Max Profile Velocity (0x607F)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x607F	0	max profile velocity	Unsigned32	rw	*)	Maximum velocity (scaled according to the velocity factor)

*) Dependent on the factory configuration of the motion controller

Profile Velocity (0x6081)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6081	0	profile velocity	Unsigned32	rw	*)	Maximum velocity (scaled according to the velocity factor)

*) Dependent on the factory configuration of the motion controller

Profile Acceleration (0x6083)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6083	0	profile acceleration	Unsigned32	rw	30 000	Maximum acceleration (scaled according to the acceleration factor)

Profile Deceleration (0x6084)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6084	0	profile deceleration	Unsigned32	rw	30 000	Maximum delay (scaled according to the acceleration factor)

Quick Stop Decelaration (0x6085)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6085	0	quick stop deceleration	Unsigned32	rw	30 000	Quick stop braking ramp value (scaled according to the acceleration factor)

Motion Profile Type (0x6086)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6086	0	motion profile type	Integer16	ro	0	Type of motion profile 0: Linear Ramp

Position Encoder Resolution (0x608F)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x608F	0	number of entries	Unsigned8	ro	2	Number of entries
	1	encoder increments	Unsigned32	rw	2048	Resolution of the external encoder for 4 edge evaluation
	2	motor revolution	Unsigned32	rw	1	Number of motor revolutions with the pulse number named in subindex 1

8 Parameter description

8.3 Drive profile objects according to CiA 402

Position Factor (0x6093)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6093	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	numerator	Unsigned32	rw	1	Numerator of the position factor
	2	feed_constant	Unsigned32	rw	1	Denominator (divisor) of the position factor

$$\text{Factor} = \frac{\text{Numerator}}{\text{Denominator (divisor)}}$$

Velocity Factor (0x6096)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6096	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	numerator	Unsigned32	rw	1	Numerator of the speed factor
	2	divisor	Unsigned32	rw	1	Denominator (divisor) of the speed factor

Acceleration Factor (0x6097)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6097	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	numerator	Unsigned32	rw	1	Numerator of the acceleration factor
	2	divisor	Unsigned32	rw	1	Denominator (Divisor) of the acceleration factor

Homing Method (0x6098)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6098	0	homing method	Integer8	rw	0	Homing method according to CiA 402

Homing Speed (0x6099)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6099	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	speed during search for switch	Unsigned32	rw	400	Speed during switch search (scaled according to the velocity factor)
	2	speed during search for home	Unsigned32	rw	100	Speed during search for zero (scaled according to the velocity factor)

Homing Acceleration (0x609A)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x609A	0	homing acceleration	Unsigned32	rw	50	acceleration during homing (scaled according to the acceleration factor)

8 Parameter description

8.3 Drive profile objects according to CiA 402

Velocity Control Parameter Set (0x60F9)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x60F9	0	number of entries	Unsigned8	ro	2	Number of object entries
	1	gain	Unsigned16	rw	*)	Velocity controller P term
	2	integration time constant	Unsigned16	rw	*)	Velocity controller I term

*) Dependent on the factory configuration of the motion controller

Control Effort (0x60FA)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x60FA	0	Control Effort	Unsigned32	ro		Controller output

Position Control Parameter Set (0x60FB)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x60FB	0	Number of entries	Unsigned8	ro	2	Number of entries
	1	gain	Unsigned16	rw	*)	Position controller P term
	2	D constant	Unsigned16	rw	*)	Position controller D term

*) Dependent on the factory configuration of the motion controller

Position Demand Value (0x60FC)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x60FC	0	position demand value	Integer32	ro	0	Last target position (increments)

Target Velocity (0x60FF)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x60FF	0	target velocity	Integer32	rw	0	Target velocity

The target velocity is specified in the units defined by the user and is converted in the internal display (1/min) using the velocity factor.

Drive Data (0x6510)

Index	Sub-index	Name	Type	Attrb.	Default value	Meaning
0x6510	0	number of entries	Unsigned8	ro	3	Number of object entries
	1	motor type	Integer32	rw	*)	Set motor type 0 BL motor –1 DC motor
	2	speed constant KN	Unsigned16	rw	*)	Speed constant Kn of the motor Unit: rpm/V
	3	motor resistance RM	Unsigned32	rw	*)	Motor resistance RM Unit: mΩ

*) Dependent on the factory configuration of the motion controller

8 Parameter description

8.4 FAULHABER commands

The FAULHABER commands can be used to configure and control the drive in a very easy way. All the supported ASCII commands of the serial version are available as a CAN message frame on PDO2, the first byte always contains the HEX value of the command, the following 4 bytes can then contain data:

RxPDO2: FAULHABER command

11 bit identifier	5 bytes user data				
0x300 (768d) + Node-ID	Cmd	LLB	LHB	HLB	HHB

The device must be in NMT "Operational" state for configuration of the drive using the FAULHABER channel.

Part of the parameter can also be set using the object dictionary, but others can only be set using the FAULHABER channel.

Several parameters can only be set and used in FAULHABER operating mode Modes of operation = -1 (Object 0x6060 or OPMOD command), as they have a direct effect on the drive behaviour.

The response behaviour of the FAULHABER commands depends on the setting of the transmission type of TxPDO2 (OD-Index 0x1801):

a.) transmission type = 0-240

The commands are not acknowledged until a SYNC object is received on TxPDO2 (see Chapter 4.5 "SYNC object").

b.) transmission type = 252

The response to a command is not made available until a SYNC object is received and can then be requested with a request (RTR) on TxPDO2.

c.) transmission type = 253 (default)

After sending the command on RxPDO2, a request (RTR) must be performed on TxPDO2, in order to obtain the response to query commands or to check the success of send commands.

d.) transmission type = 255

The commands are immediately acknowledged on TxPDO2.

TxPDO2: FAULHABER data

11 bit identifier	5 bytes user data					
0x280 (640d) + Node-ID	Cmd	LLB	LHB	HLB	HHB	Error

8 Parameter description

8.4 FAULHABER commands

6 bytes must always be returned, whereby the first byte gives the command and the following 4 bytes the required value as a long integer (for pure send commands 0) followed by an error code:

Error	Explanation
1	Command successfully executed
-2	EEPROM writing done
-4	Overtemperature – drive disabled
-5	Invalid parameter
-7	Unknown command
-8	Command not available
-13	Flash defect

Example:

Query of the actual position of node 3 (“POS” command):

```
Transmit ID 303: 40 00 00 00 00
Request ID 283
Receive ID 283: 40 A0 86 01 00 01
```

➔ Actual position = 100000D

8.4.1 Basic setting commands

The commands listed here are used to configure basic setting parameters, which are stored in the flash data memory via the SAVE command, and from there are reloaded again after switching on.

Commands for special FAULHABER operating modes

Only available in FAULHABER mode (Modes of operation = OPMOD = -1)

Command	Hex value	Data	Function	Description
OPMOD	0xFD	0	Operation Mode	CANopen operating mode: -1: FAULHABER mode 1: Profile Position Mode 3: Profile Velocity Mode 6: Homing Mode Corresponds to object 0x6060 (modes of operation)
SOR	0x8E	0-4	Source For Velocity	Source for target velocity: 0: CAN interface (default) 1: Voltage at analog input 2: PWM signal at analog input 3: Current limit value via analog input 4: current limiting value via analog input with evaluation of the sign for presetting the direction of rotation
CONTMOD	0x06	0	Continuous Mode	Switch back to normal mode from an enhanced mode
STEPMOD	0x46	0	Stepper motor mode	Change to stepper motor mode
APCMOD	0x02	0	Analog Position Control Mode	Change to position control via analog voltage
ENCMOD	0x10	0	Encoder Mode	Change to encoder mode (not for MCDC) An external encoder serves as position detector (the current position value is set to 0)
HALLSPEED	0x3B	0	Hall sensor as speed sensor	Speed via Hall sensors in encoder mode (not for MCDC)
ENCSPEED	0x12	0	Encoder as speed sensor	Speed via encoder signals in encoder mode (not for MCDC)
GEARMOD	0x1D	0	Gearing Mode	Change to gearing mode
VOLTMOD	0x49	0	Set Voltage Mode	Activate Voltage Regulator Mode
IXRMOD	0x50	0	Set IxR Mode	Activate IxR control (MCDC only)

8 Parameter description

8.4 FAULHABER commands

Parameters for basic settings

Command	Hex value	Data	Function	Description
ENCRES	0x70	Value	Load Encoder Resolution	Load resolution of external encoder (4 times pulse/rev). **Value: 8 to 65 535** Corresponds to object 0x608F
KN	0x9E	Value	Load Speed Constant	Load speed constant K_n in accordance with information in the data sheet. Unit: rpm/V. **Value: 0…16 383** Corresponds to object 0x6510
RM	0x9F	Value	Load Motor Resistance	Load motor resistance RM according to specification in data sheet. Unit: mΩ. **Value: 10…320 000** Corresponds to object 0x6510
STW	0x77	Value	Load Step Width	Load step width for step motor and gearing mode **Value: 1…65 535**
STN	0x64	Value	Load Step Number	Load number of steps per revolution for step motor and gearing mode **Value: 1…65 535**
MV	0x85	Value	Minimum Velocity	Specification of minimum velocity in rpm for target velocity via analog voltage (SOR1, SOR2) **Value: 0…30 000**
MAV	0x83	Value	Minimum Analog Voltage	Presetting of minimum start voltage in mV for presetting speed via analog voltage (SOR1, SOR2) **Value: 0…10 000**
ADL	0x00	0	Analog Direction Left	Positive voltages at the analog input result in anticlockwise rotation of the rotor (SOR1, SOR2)
ADR	0x01	0	Analog Direction Right	Positive voltages at the analog input result in clockwise rotation of the rotor (SOR1, SOR2)
SIN	0xA0	0 – 1	Sinus commutation	1: No block commutation within the upper velocity range (default) 0: Block commutation within the upper velocity range (full modulation) (not with MCDC)
POLNUM	0xDC	2, 4	Pole Number	Number of magnetic poles of the connected motor (not for MCDC) 2: Two pole motor 4: Four pole motor (e.g. BX4)
SENSTYP	0xDE	4	Load Sensor Type	Setting of the connected AES encoder (only for MCBL AES) 4: AES-4 096 Further types available on request

8 Parameter description

8.4 FAULHABER commands

General parameters

Command	Hex value	Data	Function	Description
LL	0xB5v	Value	Load Position Range Limits	Load limit positions (the drive cannot be moved out of these limits). Positive values specify the upper limit and negative values the lower. The range limits are only active if APL1 is. **Value: $-1.8 \cdot 10^9$ … $+1.8 \cdot 10^9$** Corresponds to object 0x607D
APL	0x03	0-1	Activate/Deactivate Position Limits	Activate range limits (LL) (valid for all operating modes in FAULHABER Mode except VOLTMOD). 1: Position limits activated 0: Position limits deactivated
SP	0x8F	Value	Load Maximum Speed	Load maximum speed (rpm). Setting applies to all modes. **Value: 0…30 000** Corresponds to object 0x607F or 0x6081
AC	0x65	Value	Load Command Acceleration	Load acceleration value (1/s²). **Value: 0…30 000** Corresponds to object 0x6083
DEC	0x6D	Value	Load Command Deceleration	Load deceleration value (1/s²). **Value: 0 to 30 000** Corresponds to object 0x6084 or 0x6081
SR	0xA4	Value	Load Sampling Rate	Load sampling rate of the velocity controller as a multiple of the basic controller sampling rate according to the data sheet.Value: **Value: 1…20**
POR	0x89	Value	Load Velocity Proportional Term	Load velocity controller amplification. **Values: 1…255** Corresponds to object 0x60F9
I	0x7B	Value	Load Velocity Integral Term	Load velocity controller integral term. **Value: 1…255** Corresponds to object 0x60F9
PP	0x9B	Value	Load Position Proportional Term	Load position controller amplification. **Value: 1…255** Corresponds to object 0x60FB
PD	0x9C	Value	Load Position Differential Term	Load position controller D-term. **Value: 1…255** Corresponds to object 0x60FB
CI	0xA2	Value	Load Current Integral Term	Load integral term for current controller. **Value: 1…255**
LPC	0x81	Value	Load Peak Current Limit	Load peak current (mA). **Value: 0…12 000**
LCC	0x80	Value	Load Continuous Current Limit	Load continuous current (mA). **Value: 0…12 000**
DEV	0x6F	Value	Load Deviation	Load maximum permissible deviation of actual velocity from target velocity (deviation) **Value: 0…30 000**
CORRIDOR	0x9D	Value	Load Corridor	Window around the target position. **Value: 1…32 767** Corresponds to object 0x6067

8 Parameter description

8.4 FAULHABER commands

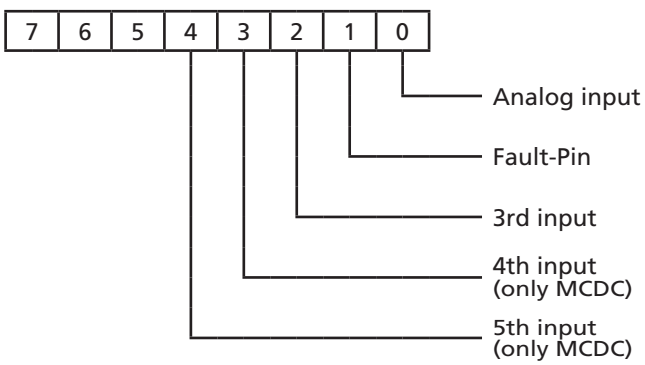
Configuration of fault pin and digital inputs

Command	Hex value	Data	Function	Description
ERROUT	0x14	0	Error Output	Fault pin as error output.
ENCOUT	0x11	0	Encoder Output	Fault pin as pulse output (not for MCDC).
DIGOUT	0x0A	0	Digital Output	Fault pin as digital output. The output is set to low level.
POSOUT	0x4C	0	Position Output	Fault pin as digital output for display of the condition: “target position reached".
DIRIN	0x0C	0	Direction Input	Fault pin as rotational direction input.
REFIN	0x41	0	Reference Input	Fault pin as reference or limit switch input.
DCE	0x6B	Value	Delayed Current Error	Delayed error output for ERROUT in 1/100 sec. **Value: 0…65 535**
LPN	0x82	Value	Load Pulse Number	Preset pulse number for ENCOUT. **Value: 1…255** **Value: 1…32 in MCBL AES**
CO	0x05	0	Clear Output	Set digital output DIGOUT to low level.
SO	0x45	0	Set Output	Set digital output DIGOUT to high level.
TO	0x55	0	Toggle Output	Toggle to digital output DIGOUT.
SETPLC	0x51	0	Set PLC inputs	Digital inputs PLC-compatible (24 V level).
SETTTL	0x52	0	Set TTL inputs	Digital inputs TTL-compatible (5 V level).

Configuring homing and limit switches in FAULHABER mode

Command	Hex value	Data	Function	Description
HP	0x79	0	Hard Polarity	Define valid edge and polarity of respective limit switches: 1: Rising edge and high level valid. 0: Falling edge and low level valid.
HB	0x73	0	Hard Blocking	Activate Hard Blocking function for relevant limit switch.
HD	0x74	0	Hard Direction	Presetting of direction of rotation that is blocked with HB of respective limit switch. 1: Clockwise rotation blocked 0: Anticlockwise rotation blocked
SHA	0x8A	0	Set Home Arming for Homing Sequence	Homing behaviour (GOHOSEQ): Set position value to 0 at edge of respective limit switch.
SHL	0x90	0	Set Hard Limit for Homing Sequence	Homing behaviour (GOHOSEQ): Stop motor at edge of respective limit switch.
SHN	0x9A	Value	Set Hard Notify for Homing Sequence	Homing behaviour (GOHOSEQ): Transmit message to Master for edge at respective limit switch (statusword Bit 14=1).
HOSP	0x78	Value	Load Homing Speed	Load speed and direction of rotation for homing (GOHOSEQ, GOHIX, FHIX). **Value: –32 767 to 32 767 rpm**
HA	0x72	0	Home Arming	Set position value to 0 and delete relevant HA bit at edge of respective limit switch. Setting is not saved.
HL	0x75	0	Hard Limit	Stop motor and delete relevant HL bit at edge of respective limit switch. Setting is not saved.
HN	0x76	0	Hard Notify	Transmit message to respective Master for edge at respective limit switch (Statusword Bit 14=1) and delete corresponding HN bit. Setting is not saved.

Bit mask of the limit switches:



8 Parameter description

8.4 FAULHABER commands

8.4.2 Query commands for basic settings

Operating modes and general parameters

Command	Hex value	Data	Function	Description
GOPMOD	0xFE	0	Get Operation Mode	Display current CANopen operating mode: -1: FAULHABER mode 1: Profile Position Mode 3: Profile Velocity Mode 6 : Homing Mode Corresponds to object 0x6061 (modes of operation display)
CST	0x58	0	Configuration Status	Set operating mode. Return value binary coded (LSB=Bit 0): Bit 0-2, reserved Bit 3-4, Source for target velocity: 0: SOR0 (CAN interface) 1: SOR1 (Analog voltage) 2: SOR2 (PWM signal) 3: SOR3 (current limitation value) Bit 5-6, reserved Bit 7-9, FAULHABER mode: 0: CONTMOD 1: STEPMOD 2: APCMOD 3: ENCMOD/HALLSPEED 4: ENCMOD/ENCSPEED 5: GEARMOD 6: VOLTMOD 7: IXRMOD Bit 10, power amplifier: 0: Disabled (DI) 1: Enabled (EN) Bit 11, position controller: 0: Switched off 1: Switched on Bit 12, analog direction of rotation: 0: ADL 1: ADR Bit 13, Position Limits APL: 0: deactivated 1: activated Bit 14, sinus commutation SIN: 0: Allow block commutation 1: Do not allow block commutation

8 Parameter description

8.4 FAULHABER commands

Command	Hex value	Data	Function	Description
GMOD	0x28	0	Get Mode	Set FAULHABER mode: 0: CONTMOD 1: STEPMOD 2: APCMOD 3: ENCMOD / HALLSPEED 4: ENCMOD / ENCSPEED 5: GEARMOD 6: VOLTMOD 7: IXRMOD
GENCRES	0x1E	0	Get Encoder Resolution	Set encode resolution (ENCRES) Corresponds to object 0x608F
GMOTTYP	0x29	0	Get Motor Type	Set motor type 0: BL motor -1: DC Motor
GKN	0x4D	0	Get Speed Constant	Speed constant in rpm/V (KN) Corresponds to object 0x6510
GRM	0x4E	0	Get Motor Resistance	Motor resistance in mΩ (RM) Corresponds to object 0x6510
GSTW	0x39	0	Get Step Width	Set step width (STW)
GSTN	0x38	0	Get Step Number	Set number of steps per revolution (STN)
GMV	0x2A	0	Get Minimum Velocity	Set minimum velocity in rpm (MV)
GMAV	0x27	0	Get minimum analog voltage	Set minimum start voltage value in mV (MAV)
GPL	0x31	0	Get Positive Limit	Set positive limit position (LL) Corresponds to object 0x607D
GNL	0x2C	0	Get Negative Limit	Set negative limit position (LL) Corresponds to object 0x607D
GSP	0x36	0	Get Maximum Speed	Set maximum speed in rpm (SP) Corresponds to object 0x607F ory 0x6081
GAC	0x15	0	Get Acceleration	Set acceleration value in 1/s² (AC) Corresponds to object 0x6083
GDEC	0x1B	0	Get Deceleration	Set deceleration value in 1/s² (DEC) Corresponds to object 0x6084
GSR	0x56	0	Get Sampling Rate	Set sampling rate of the speed controller ms/10 (SR)
GPOR	0x33	0	Get Velocity Prop. Term	Set amplification value of the speed controller (POR) Corresponds to object 0x60F9
GI	0x26	0	Get Velocity Integral Term	Set integral term of the speed controller (I) Corresponds to object 0x60F9
GPP	0x5D	0	Get Position Prop. Term	Set amplification value of the position controller (PP) Corresponds to object 0x60FB
GPD	0x5E	0	Get Position D-Term	Set D-term of the position controller (PD) Corresponds to object 0x60FB
GCI	0x63	0	Get Current Integral Term	Set integral term of the current controller (CI)
GPC	0x30	0	Get Peak Current	Set peak current in mA (LPC)
GCC	0x18	0	Get Continuous Current	Set continuous current in mA (LCC)
GDEV	0x1C	0	Get Deviation	Set deviation value (DEV)
GCORRIDOR	0x62	0	Get Corridor	Set window around the target position (CORRIDOR) Corresponds to object 0x6067.
GPOLNUM	0xDB	0	Get Pole Number	Number of magnetic poles of the connected motor (not for MCDC)
GSENSTYP	0xDD	0	Get Sensor Type	Setting of the connected AES encoder. (only for MCBL AES)

8 Parameter description

8.4 FAULHABER commands

Configuration of fault pin and digital inputs

Command	Hex value	Data	Function	Description
IOC	0x5C	0	I/O Configuration	Set input/output configuration. Return value binary coded (LSB=Bit 0): Bit 0-7, FAULHABER Hard Blocking: 0-31: Function active for input 1-5 Bit 8-15, FAULHABER Hard Polarity: 0-31: Rising edge at input 1-5 Bit 16-23, FAULHABER Hard Direction: 0-31: Clockwise movement blocked at input 1-5 Bit 24, state of digital output: 0: Low 1: High Bit 25, Level of digital inputs: 0: TTL level (5 V) 1: PLC LEVEL (24 V) Bit 26-28, function of fault pin: 0: ERROUT 1: ENCOUT 2: DIGOUT 3: DIRIN 4: REFIN 5: POSOUT
GDCE	0x1A	0	Get Delayed Current Error	Set value of the error output delay (DCE)
GPN	0x32	0	Get Pulse Number	Set pulse number (LPN)

Configuration of the homing in FAULHABER mode

Command	Hex value	Data	Function	Description
HOC	0x5B	0	Homing Configuration	Set homing configuration. Return value binary coded (LSB = Bit 0): Bit 0-7, SHA setting for input 1-8 Bit 8-15, SHN setting for input 1-8 Bit 16-23, SHL setting for input 1-8 (Input 6-8: Reserved)
GHOSP	0x24	0	Get Homing Speed	Set homing speed in rpm (HOSP).

8.4.3 Miscellaneous commands

Command	Hex value	Data	Function	Description
SAVE EEPSAV	0x53 0x0D	0	Save Parameters	Save current parameters and configuration setting to Flash memory. The drive will also start with these settings when next switched on. Corresponds to object 0x1010. **Important:** Command may not be executed more than 10 000 times, as otherwise the function of the Flash memory can no longer be guaranteed.
RESET	0x59	0	Reset	Restart drive node. Corresponds to NMT reset node.
RN	0x44	0	Reset Node	Set application parameters to original values (ROM values) (current, acceleration, controller parameters, maximum speed, limit positions…); communication parameters, operating mode and hardware configuration are retained.
FCONFIG	0xD0	0	Factory Configuration	All configurations and values are reset to the delivery status. After this command the drive performs a reset. **Attention:** Customer-specific factory settings are also lost. The Node-ID is set to 255 (unconfigured), therefore LSS configuration is necessary to establish a new connection! The command can be executed a maximum 10 000 times.

8 Parameter description

8.4 FAULHABER commands

8.4.4 Motion control commands

The commands listed here are only available in FAULHABER Mode (Modes of operation = -1).

Command	Hex value	Data	Function	Description
DI	0x08	0	Disable Drive	Deactivate drive.
EN	0x0F	0	Enable Drive	Activate drive.
M	0x3C	0	Initiate Motion	Activate position control and start positioning.
LA	0xB4	Value	Load Absolute Position	Load new absolute target position. **Value: -1.8 · 10^9 … 1.8 · 10^9**
LR	0xB6	Value	Load Relative Position	Load new relative target position, in relation to last started target position. The resulting absolute target position must lie between –2.14 10^9 and 2.14 10^9.
V	0x93	Value	Select Velocity Mode	Activate velocity mode and set specified value as target velocity (velocity control). **Value: –30 000 … 30 000 rpm**
U	0x92	Value	Set Output Voltage	Output PWM value in VOLTMOD. **Value: –32 767…32 767 (corresponds to –Uv…+Uv)**
GOHOSEQ	0x2F	0	Go Homing Sequence	Execute FAULHABER homing sequence. A homing sequence is executed (if programmed) irrespective of the current mode.
FHIX	0x35	0	Find Hall Index	Move BL 4 pole motor to Hall zero point (Hall index) and set actual position value to 0. In 4 pole motors, within a revolution, two Hall zero points exist opposite each other. The motor moves to the nearest index. (only for BL 4 pole)
GOHIX	0x2E	0	Go Hall Index	Move BL motor to Hall zero point (Hall index) and set actual position value to 0 (only for BL 2 pole).
GOIX	0xA3	0	Go Encoder Index	Move to the encoder index at the Fault pin and set actual position value to 0 (DC motor or ext. encoder).
HO	0xB8	0/value	Define Home Position	Data = 0: Set actual position to 0. Otherwise: Set actual position to specified value. **Value: –1.8 · 10^9 …1.8 · 10^9**

8 Parameter description

8.4 FAULHABER commands

8.4.5 General query commands

Command	Hex value	Data	Function	Description
POS	0x40	0	Get Actual Position	Current actual position. Corresponds to object 0x6063.
TPOS	0x4B	0	Get Target Position	Target position of the positioning last started. Corresponds to object 0x60FC.
GV	0x3A	0	Get Target Velocity	Current target velocity in rpm. Corresponds to object 0x606B.
GN	0x2B	0	Get Actual Velocity	Current actual speed in rpm. Corresponds to object 0x6069.
GU	0x5F	0	Get PWM Voltage	Set PWM value in VOLTMOD.
GRU	0x60	0	Get Real PWM Voltage	Current controller output value.
GCL	0x19	0	Get Current Limit	Current limitation current in mA.
GRC	0x34	0	Get Real Current	Current actual current in mA.
TEM	0x47	0	Get Temperature	Current housing temperature in °C.
GADV	0xB2	Value	Get Analog Voltage	Read out the voltage applied at the given input (value). Scaling: 1 000 digits = 1 V 1: Voltage at AnIn 3: Voltage at 3rd In 4: Voltage at 4th In (MCDC only) 5: Voltage at 5th In (MCDC only) Return value input 1: -10 000 … 10 000 Return value input 3, 4, 5: 0 … 10 000 **Value: 1, 3, 4, 5 (4 and 5 for MCDC only)**
OST	0x57	0	Operation Status	Display current operating status. Return value binary coded (LSB=Bit 0): Bit 0: Homing running Bit 1-3: Reserved Bit 4: Current limitation active Bit 5: Deviation error Bit 6: Overvoltage Bit 7: Overtemperature Bit 8: Status input 1 Bit 9: Status input 2 Bit 10: Status input 3 Bit 11: Status input 4 Bit 12: Status input 5 Bit 13-15: Reserved for other inputs Bit 16: Position attained
SWS	0x5A	0	Switch Status	Temporary limit switch settings. Return value binary coded (LSB=Bit 0): Bit 0-7: HA setting for input 1-8 Bit 8-15: HN setting for input 1-8 Bit 16-23: HL setting for input 1-8 Bit 24-31: Specifies which limit switch 1-8 has already switched (is reset again when the respective input is reset)

8 Parameter description

8.4 FAULHABER commands

8.4.6 Command overview

Command	Command code	Function	CANopen object
AC	0x65	Load Command Acceleration	0x6083
ADL	0x00	Analog Direction Left	
ADR	0x01	Analog Direction Right	
APCMOD	0x02	Analog Position Control Mode	
APL	0x03	Activate/Deactivate Position Limits	
CI	0xA2	Load Current Integral Term	
CO	0x05	Clear Output	
CONTMOD	0x06	Continuous Mode	
CORRIDOR	0x9D	Load Corridor	0x6067
CST	0x58	Configuration Status	
DCE	0x6B	Delayed Current Error	
DEC	0x6D	Load Command Deceleration	0x6084
DEV	0x6F	Load Deviation	
DI	0x08	Disable Drive	
DIGOUT	0x0A	Digital Output	
DIRIN	0x0C	Direction Input	
EN	0x0F	Enable Drive	
ENCMOD	0x10	Encoder Mode	
ENCOUT	0x11	Encoder Output	
ENCRES	0x70	Load Encoder Resolution	0x608F
ENCSPEED	0x12	Encoder as speed sensor	
ERROUT	0x14	Error Output	
FAULT STATUS	0xDF	Get Fault Pin Status	
FCONFIG	0xD0	Factory Configuration	
FHIX	0x35	Find Hall Index	
GAC	0x15	Get Acceleration	0x6083
GADC	0xB3	Get ADC Value	
GADV	0xB2	Get Analog Voltage	
GCC	0x18	Get Continuous Current	
GCI	0x63	Get Current Integral Term	
GCL	0x19	Get Current Limit	
GCORRIDOR	0x62	Get Corridor	0x6067
GDCE	0x1A	Get Delayed Current Error	
GDEC	0x1B	Get Deceleration	0x6084
GDEV	0x1C	Get Deviation	
GEARMOD	0x1D	Gearing Mode	
GENCRES	0x1E	Get Encoder Resolution	0x608F
GHOSP	0x24	Get Homing Speed	
GI	0x26	Get Velocity Integral Term	0x60F9
GKN	0x4D	Get Speed Constant	0x6510
GMAV	0x27	Get minimum analog voltage	
GMOD	0x28	Get Mode	
GMOTTYP	0x29	Get Motor Type	0x6510
GMV	0x2A	Get Minimum Velocity	
GN	0x2B	Get Actual Velocity	0x6069
GNL	0x2C	Get Negative Limit	0x607D
GOHIX	0x2E	Go Hall Index	
GOHOSEQ	0x2F	Go Homing Sequence	
GOIX	0xA3	Go Encoder Index	
GOPMOD	0xFE	Get Operation Mode	0x6061
GPC	0x30	Get Peak Current	
GPD	0x5E	Get Position D-Term	0x60FB
GPL	0x31	Get Positive Limit	0x607D
GPN	0x32	Get Pulse Number	
GPOLNUM	0xDB	Get Pole Number	
GPOR	0x33	Get Velocity Prop. Term	0x60F9
GPP	0x5D	Get Position Prop. Term	0x60FB
GRC	0x34	Get Real Current	
GRM	0x4E	Get Motor Resistance	0x6510
GRPC	0x61	Get Real Current	
GRU	0x60	Get Real PWM Voltage	0x60FA
GSENSTYP	0xDD	Get Sensor Type (MCBL AES)	
GSP	0x36	Get Maximum Speed	0x6081
GSR	0x56	Get Sampling Rate	
GSTN	0x38	Get Step Number	
GSTW	0x39	Get Step Width	
GU	0x5F	Get PWM Voltage	
GV	0x3A	Get Target Velocity	0x606B

8 Parameter description

8.4 FAULHABER commands

Command	Command code	Function	CANopen object
HA	0x72	Home Arming	
HALLSPEED	0x3B	Hall sensor as speed sensor	
HB	0x73	Hard Blocking	
HD	0x74	Hard Direction	
HL	0x75	Hard Limit	
HN	0x76	Hard Notify	
HO	0xB8	Define Home Position	
HOC	0x5B	Homing Configuration	
HOSP	0x78	Load Homing Speed	
HP	0x79	Hard Polarity	
I	0x7B	Load Velocity Integral Term	0x60F9
IOC	0x5C	I/O Configuration	
IXRMOD	0x50	Set IxR Mode (MCDC)	
KN	0x9E	Load Motor Speed Constant	0x6510
LA	0xB4	Load Absolute Position	
LCC	0x80	Load Continuous Current Term	
LL	0xB5	Load Position Range Limits	0x607D
LPC	0x81	Load Peak Current Limit	
LPN	0x82	Load Pulse Number	
LR	0xB6	Load Relative Position	
M	0x3C	Initiate Motion	
MAV	0x83	Minimum Analog Voltage	
MV	0x85	Minimum Velocity	
OPMOD	0xFD	Operation Mode	0x6060
OST	0x57	Operation Status	
PD	0x9C	Load Position Differential Term	0x60FB
POLNUM	0xDC	Load Pole Number	
POR	0x89	Load Velocity Proportional Term	0x60F9
POS	0x40	Get Actual Position	0x6063
POSOUT	0x4C	Position Output	
PP	0x9B	Load Position Proportional Term	0x60FB
REFIN	0x41	Reference Input	
RESET	0x59	Reset	
RM	0x9F	Load Motor Resistance	0x6510
RN	0x44	Reset Node	
SAVE	0x53	Save Parameters	0x1010
SENSTYP	0xDE	Load Sensor Type (MCBL AES)	
SETPLC	0x51	Set PLC inputs	
SETTTL	0x52	Set TTL inputs	
SHA	0x8A	Set Home Arming for Homing Sequence	
SHL	0x90	Set Hard Limit for Homing Sequence	
SHN	0x9A	Set Hard Notify for Homing Sequence	
SIN	0xA0	Sinus commutation	
SO	0x45	Set Output	
SOR	0x8E	Source for Velocity	
SP	0x8F	Load Maximum Speed	0x607F
SR	0xA4	Load Sampling Rate	
STEPMOD	0x46	Stepper Motor Mode	
STN	0x64	Load Step Number	
STW	0x77	Load Step Width	
SWS	0x5A	Switch Status	
TEM	0x47	Get Temperature	
TO	0x55	Toggle Output	
TPOS	0x4B	Get Target Position	0x60FC
U	0x92	Set Output Voltage	
V	0x93	Select Velocity Mode	0x606B
VER	-	Get Firmware Version	0x100A
VOLTMOD	0x49	Set Voltage Mode	

FAULHABER

MA7000.05030 English, 4th edition, 08.2014

DR. FRITZ FAULHABER GMBH & CO. KG
Antriebssysteme

Daimlerstraße 23/25
71101 Schönaich · Germany
Tel. +49(0)7031/638-0
Fax +49(0)7031/638-100
info@faulhaber.de
www.faulhaber.com