

simover masterdrives

Motion Control

SIEMENS

www.ElectricalPartManuals.com

Definitions and Warnings

Qualified personnel

For the purpose of this documentation and the product warning labels, a "Qualified person" is someone who is familiar with the installation, mounting, start-up, operation and maintenance of the product. He or she must have the following qualifications:

- ◆ Trained or authorized to energize, de-energize, ground and tag circuits and equipment in accordance with established safety procedures.
- ◆ Trained or authorized in the proper care and use of protective equipment in accordance with established safety procedures.
- ◆ Trained in rendering first aid.

DANGER



indicates an **imminently** hazardous situation which, if not avoided, will result in death, serious injury and considerable damage to property.

WARNING



indicates a **potentially** hazardous situation which, if not avoided, could result in death, serious injury and considerable damage to property.

CAUTION



used with the safety alert symbol indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury.

CAUTION

used without safety alert symbol indicates a potentially hazardous situation which, if not avoided, may result in property damage.

NOTICE

NOTICE used without the safety alert symbol indicates a potential situation which, if not avoided, may result in an undesirable result or state.

NOTE

For the purpose of this documentation, "Note" indicates important information about the product or about the respective part of the documentation which is essential to highlight.

WARNING

Hazardous voltages are present in this electrical equipment during operation.

Non-observance of the warnings can thus result in severe personal injury or property damage.

Only qualified personnel should work on or around the equipment

This personnel must be thoroughly familiar with all warning and maintenance procedures contained in this documentation.

The successful and safe operation of this equipment is dependent on correct transport, proper storage and installation as well as careful operation and maintenance.

NOTE

This documentation does not purport to cover all details on all types of the product, nor to provide for every possible contingency to be met in connection with installation, operation or maintenance.

Should further information be desired or should particular problems arise which are not covered sufficiently for the purchaser's purposes, the matter should be referred to the local SIEMENS sales office.

The contents of this documentation shall not become part of or modify any prior or existing agreement, commitment or relationship. The sales contract contains the entire obligation of SIEMENS AG. The warranty contained in the contract between the parties is the sole warranty of SIEMENS AG. Any statements contained herein do not create new warranties or modify the existing warranty.

SIEMENS

SIMOVERT MASTERDRIVES

MOTION CONTROL

Compendium

in Volume 1

System Description

Configuration and Connection
Examples

Instructions for Design of Drives
in Conformance with
EMC Regulations

Function Blocks and Parameters

Parameterization

Parameterizing Steps

Functions

Communication

Technology Option F01

Control Word and Status Word

Engineering Information

in Volume 2

Function Diagrams

Parameter Lists

Faults and Alarms

Lists of Stored Motors
Dimension Drawings

Version AG

6SE7087-6QX70

www.ElectricalPartManuals.com

Contents

1	SYSTEM DESCRIPTION	1-1
1.1	Overview	1-1
1.2	System description	1-2
1.3	Construction sizes	1-3
1.4	Communication	1-4
2	CONFIGURATION AND CONNECTION EXAMPLES	2-1
2.1	Compact PLUS type units	2-1
2.1.1	Single-axis drive	2-1
2.1.2	Multi-axis drive up to 3 axes	2-1
2.1.3	Multi-axis drive	2-2
2.1.4	Configuration and connection examples (Compact PLUS)	2-5
2.2	Compact and chassis-type units	2-9
2.2.1	Water-cooled units	2-9
2.2.2	Single-axis drive with Compact or chassis-type units	2-9
2.2.3	Multi-axis drive with Compact or chassis-type units	2-9
2.2.4	Explanations relating to the configuration examples (Compact and chassis-type units)	2-13
2.3	Points to look out for when using certain option boards and CUPM	2-16
2.3.1	Encoder interface connections	2-16
2.3.2	TB boards	2-17
2.3.3	EB boards	2-17
3	INSTRUCTIONS FOR DESIGN OF DRIVES IN CONFORMANCE WITH EMC REGULATIONS	3-1
3.1	Foreword	3-1
3.2	Principles of EMC	3-2
3.2.1	What is EMC?	3-2
3.2.2	Noise emission and noise immunity	3-2
3.2.3	Industrial and domestic applications	3-3
3.2.4	Non-grounded systems	3-3
3.3	The frequency converter and its electromagnetic compatibility	3-4
3.3.1	The frequency converter as a noise source	3-4
3.3.2	The frequency converter as a noise receiver	3-7

3.4	EMC planning	3-8
3.4.1	The zone concept	3-10
3.4.2	Use of filters and coupling elements	3-12
3.5	Design of drives in conformance with EMC regulations	3-13
3.5.1	Basic EMC rules	3-13
3.5.2	Examples	3-19
3.6	Assignment of SIMOVERT MASTERDRIVES, radio interference suppression filters and line reactors	3-26
3.7	Specified standards	3-26
4	FUNCTION BLOCKS AND PARAMETERS	4-1
4.1	Function blocks	4-1
4.2	Connectors and binectors	4-2
4.3	Parameters	4-4
4.4	Connecting up function blocks (BICO system)	4-8
5	PARAMETERIZATION	5-1
5.1	Parameter menus	5-1
5.2	Changeability of parameters	5-5
5.3	Parameter input via the PMU	5-6
5.4	Parameter input via the OP1S	5-11
5.4.1	General	5-11
5.4.2	Connecting, run-up	5-13
5.4.2.1	Connecting	5-13
5.4.2.2	Run-up	5-14
5.4.3	Operator control	5-16
5.4.3.1	Operator control elements	5-16
5.4.3.2	Operating display	5-17
5.4.3.3	Basic menu	5-18
5.4.3.4	Slave ID	5-19
5.4.3.5	OP: Upread	5-20
5.4.3.6	OP: Download	5-21
5.4.3.7	Delete data	5-22
5.4.3.8	Menu selection	5-23
5.4.3.9	Issuing commands via the OP1S	5-29
5.4.4	Bus operation	5-30
5.4.4.1	Configuring slaves	5-30
5.4.4.2	Changing slaves	5-31
5.4.5	Technical data	5-31

5.5	Parameter input with DriveMonitor	5-32
5.5.1	Installation and connection	5-32
5.5.1.1	Installation	5-32
5.5.1.2	Connection	5-32
5.5.2	Establishing the connection between DriveMonitor and the device	5-33
5.5.2.1	Setting the USS interface	5-33
5.5.2.2	Starting the USS bus scan	5-35
5.5.2.3	Creating a parameter set	5-36
5.5.3	Parameterization	5-38
5.5.3.1	Structure of the parameter lists, parameterization with DriveMonitor	5-38
5.5.3.2	General diagnostics	5-43
6	PARAMETERIZING STEPS	6-1
6.1	Parameter reset to factory setting	6-3
6.2	Detailed parameterization	6-5
6.2.1	Power section definition	6-5
6.2.2	Board configuration	6-11
6.2.3	Drive setting	6-15
6.2.4	Motor identification	6-22
6.2.4.1	Standstill measurement	6-22
6.2.4.2	No-load measurement	6-23
6.2.5	Function adjustment	6-23
6.3	Quick parameterization procedures	6-24
6.3.1	Parameterizing with user settings	6-24
6.3.2	Parameterizing by loading parameter files (download P060 = 6)	6-25
6.3.3	Parameterizing with parameter modules (quick parameterization, P060 = 3) ..	6-28
7	FUNCTIONS	7-1
7.1	Basic functions	7-1
7.1.1	Time slots	7-1
7.1.1.1	Time slots T0 to T20	7-1
7.1.1.2	Processing sequence	7-3
7.1.1.3	Assignment of function blocks to time slots	7-4
7.1.2	Processing sequence of the function blocks	7-6
7.1.2.1	Time monitoring	7-7
7.1.2.2	Influencing the time response	7-8
7.2	Technology functions	7-10
7.2.1	Comfort ramp-function generator	7-10
7.2.2	Technology controller	7-11
7.2.3	Basic positioning	7-12
7.2.3.1	Functions	7-16
7.2.3.2	Normalization	7-24
7.2.3.3	Operating modes	7-28
7.2.3.4	Preprocessing of position setpoint	7-44
7.2.3.5	Application example	7-46
7.2.3.6	Change history	7-50

7.3	Converter functions	7-51
7.3.1	Friction characteristic function (function diagram 399).....	7-51
7.3.1.1	Friction characteristic.....	7-51
7.3.1.2	Friction characteristic recording (automatic procedure).....	7-51
7.3.2	Torque constant adaptation for synchronous motors (function diagram 393)...	7-53
7.3.3	Tr adaptation function (function diagram 394).....	7-55
7.3.4	Position test function.....	7-56
7.3.5	Function "PRBS signal with recording" (function diagram 796)	7-59
7.3.6	Function "speed filter" (function diagram 361)	7-61
7.3.7	"Speed controller characteristic" function (function diagram 360)	7-65
7.3.8	"Field weakening for synchronous motor" function (function diagram 389)...	7-66
7.3.9	Vdmax closed-loop control (function diagram 610).....	7-71
7.3.10	Harmonic compensation.....	7-72
7.4	Special functions.....	7-73
7.4.1	Loading firmware	7-73
8	COMMUNICATION	8-1
8.1	Universal Serial Interface (USS).....	8.1-1
8.1.1	Protocol specification and bus structure.....	8.1-2
8.1.1.1	Protocol specification.....	8.1-2
8.1.1.2	Bus structure.....	8.1-7
8.1.2	The structure of net data	8.1-10
8.1.2.1	General structure of the net-data block.....	8.1-10
8.1.2.2	PKW area	8.1-11
8.1.2.3	Process-data area (PZD).....	8.1-19
8.1.3	Interface overview.....	8.1-20
8.1.4	Connecting-up	8.1-23
8.1.4.1	Bus cable connection	8.1-23
8.1.4.2	Fitting the bus cable.....	8.1-24
8.1.4.3	EMC measures	8.1-25
8.1.4.4	Bus termination, USS protocol	8.1-28
8.1.5	Start-up	8.1-31
8.1.5.1	Parameterization of the USS protocol (1st step).....	8.1-32
8.1.5.2	Parameterizing the parameterizing enable and process-data interconnections (2nd step)	8.1-36
8.2	PROFIBUS	8.2-1
8.2.1	Product description of the CBP communications board.....	8.2-1
8.2.2	Description of the CBP's functions on the PROFIBUS-DP	8.2-3
8.2.2.1	Cyclical data transmission	8.2-5
8.2.2.2	Acyclical data transfer	8.2-10
8.2.2.3	Acyclical master class 1, automation (PLC)	8.2-12
8.2.2.4	Acyclical master class 2 - Configuration (DriveES).....	8.2-16
8.2.2.5	Acyclical master class 2 - Operator control (SIMATIC OP).....	8.2-17
8.2.3	Mechanisms for processing parameters via the PROFIBUS	8.2-18
8.2.4	PROFIdrive V3: Acyclic parameter accessing with data block 47	8.2-26
8.2.4.1	Comparison between parameter tasks to PROFIdrive version 2 and 3.....	8.2-28
8.2.4.2	Example of "Request parameter value", simple	8.2-29
8.2.4.3	Example of "Change parameter value", simple.....	8.2-30
8.2.4.4	Example of "Request parameter value", more than one array element.....	8.2-31

8.2.4.5	Example of "Change parameter value", more than one array element.....	8.2-32
8.2.4.6	Example of "Request parameter value", multi-parameter	8.2-33
8.2.4.7	Example of "Change parameter value", multi-parameter	8.2-35
8.2.4.8	Request description, individual	8.2-37
8.2.4.9	Request description, total	8.2-38
8.2.4.10	Request text, individual	8.2-39
8.2.4.11	Coding in the parameter task according to PROFIdrive Version 3	8.2-40
8.2.5	Mounting methods / CBP slots	8.2-43
8.2.5.1	CBP mounting slots in MC Compact PLUS units	8.2-43
8.2.5.2	CBP slots in Compact units and chassis-type units with the CUs of function classes Motion Control Performance 2 (CUPM), Motion Control (CUMC) and Vector Control (CUVC).....	8.2-44
8.2.5.3	CBP slots in Compact and chassis-type units with the CUs of function classes FC (CU1), VC (CU2) or SC (CU3).....	8.2-46
8.2.6	Connecting up the CBP to the PROFIBUS	8.2-47
8.2.6.1	Assignment of plug-in connector X448.....	8.2-47
8.2.6.2	Connecting up the bus cable by means of the RS485 bus connecting system	8.2-47
8.2.6.3	Connecting the bus cable with the fiber-optic cable system	8.2-52
8.2.6.4	Shielding of the bus cable / EMC measures	8.2-54
8.2.7	Starting up the CBP	8.2-57
8.2.7.1	Basic parameterization	8.2-57
8.2.7.2	Process data interconnection in the units.....	8.2-62
8.2.7.3	Process data interconnection via standard telegrams	8.2-69
8.2.7.4	Process data monitoring.....	8.2-71
8.2.8	Settings for the PROFIBUS-DP master (Class 1)	8.2-73
8.2.8.1	Operating the CBP with a SIMATIC S5.....	8.2-75
8.2.8.2	Operating the CBP with a SIMATIC S7	8.2-77
8.2.8.3	Operating the CBP with a non-Siemens system	8.2-79
8.2.8.4	Operating the CBP2 with extended functions with a SIMATIC S7	8.2-80
8.2.8.5	CBP2 with cross traffic operated with a SIMATIC S7.....	8.2-81
8.2.8.6	CBP2 with clock synchronization operated with a SIMATIC S7	8.2-83
8.2.8.7	CBP2 with clock synchronization on a PROFIBUS master in accordance with PROFIdrive V3	8.2-86
8.2.9	MASTERDRIVES as PROFIdrive V3-Slave	8.2-88
8.2.9.1	Incorporation of drives in automation systems / plant characterization	8.2-90
8.2.9.2	Communication model.....	8.2-94
8.2.9.3	Drive control.....	8.2-95
8.2.9.4	Checkback messages (status words).....	8.2-96
8.2.9.5	Setpoints / Actual values	8.2-97
8.2.9.6	Dynamic Servo Control (DSC).....	8.2-98
8.2.9.7	Communication interface	8.2-106
8.2.9.8	Clock synchronous application	8.2-106
8.2.9.9	Encoder interface (from SW 1.6)	8.2-107
8.2.10	Diagnosis and troubleshooting	8.2-117
8.2.10.1	Evaluating the possibilities of hardware diagnosis	8.2-117
8.2.10.2	Fault and alarm display on the basic unit	8.2-119
8.2.10.3	Evaluating CBP diagnostic parameters	8.2-122
8.2.10.4	Meaning of information in the CBP diagnosis parameter r723.....	8.2-124
8.2.10.5	Additional methods of diagnosis for start-up personnel	8.2-128
8.2.10.6	CBP2 diagnostic parameters.....	8.2-134
8.2.10.7	Extended CBP2 diagnosis for start-up personnel	8.2-137
8.2.11	Appendix.....	8.2-140

8.3	SIMOLINK.....	8.3-1
8.3.1	General principles.....	8.3-1
8.3.2	Peer-to-peer functionality	8.3-5
8.3.3	Application with peer-to-peer functionality.....	8.3-6
8.3.4	Components of the peer-to-peer functionality	8.3-8
8.3.5	Parameterization of the peer-to-peer functionality	8.3-10
8.3.6	Diagnostics of the peer-to-peer functionality	8.3-14
8.3.7	Synchronization of the control circuits by means of the bus cycle time (MC only)	8.3-16
8.3.8	Synchronization diagnostics (MC only)	8.3-18
8.3.9	Switchover of the synchronization source (MC only)	8.3-18
8.3.10	Special data and application flags.....	8.3-20
8.3.11	Configuration (example of peer-to-peer functionality)	8.3-21
8.3.12	Master/slave functionality	8.3-25
8.3.13	Application with master/slave functionality	8.3-26
8.4	CBC Communications Board.....	8.4-1
8.4.1	Product description.....	8.4-1
8.4.2	Mounting methods / CBC slots	8.4-4
8.4.2.1	Mounting positions of the CBC in MC Compact PLUS units.....	8.4-4
8.4.2.2	Mounting positions of the CBC in Compact and chassis units of function classes MC (CUPM, CUMC) and VC (CUVC)	8.4-5
8.4.2.3	Mounting positions of the CBC in Compact type and chassis type units with the CU of the function classes FC (CU1), VC (CU2) or SC (CU3)	8.4-6
8.4.2.4	Mounting positions of the CBC in VC Compact PLUS units	8.4-7
8.4.3	Connecting.....	8.4-8
8.4.3.1	Connection of the bus cable	8.4-9
8.4.3.2	EMC measures	8.4-10
8.4.3.3	Bus termination of the CAN bus (jumper S1.2)	8.4-13
8.4.3.4	Ground connection (jumper S1.1)	8.4-13
8.4.3.5	Interface X458 / X459 with jumper strip S1	8.4-14
8.4.3.6	Recommended circuits.....	8.4-15
8.4.4	Data transfer via the CAN bus.....	8.4-16
8.4.4.1	General	8.4-16
8.4.4.2	Parameter area (PKW)	8.4-17
8.4.4.3	Process data area (PZD).....	8.4-24
8.4.5	Start-up of the CBC	8.4-31
8.4.5.1	Basic parameterization of the units	8.4-32
8.4.5.2	Process-data softwiring in the units.....	8.4-45
8.4.6	Diagnosis and troubleshooting	8.4-52
8.4.6.1	Evaluation of hardware diagnostics.....	8.4-52
8.4.6.2	Fault displays and alarms on the basic unit	8.4-54
8.4.6.3	Evaluation of the CBC diagnostic parameter	8.4-56
8.4.6.4	Meaning of CBC diagnosis	8.4-57
8.4.7	Appendix.....	8.4-60

8.5	CBC CANopen communication board	8.5-1
8.5.1	Object directory	8.5-5
8.5.2	Commissioning the CBC	8.5-16
8.5.2.1	General settings	8.5-16
8.5.2.2	NMT state	8.5-27
8.5.2.3	Relation between PDO/PZD and SDO/PKW	8.5-29
8.5.2.4	PDO mapping	8.5-34
8.5.3	Manufacturer-specific objects	8.5-48
8.5.3.1	Parameter editing	8.5-48
8.5.3.2	Example: Change parameter value with object 4001h	8.5-52
8.5.3.3	Setting factory values (defaults) via CANopen	8.5-53
8.5.3.4	Changing the baud rate and bus address (on MASTERDRIVES MC only)	8.5-54
8.5.4	Faults and alarms	8.5-55
8.5.4.1	Structure of object 1003 _h (pre-defined error field)	8.5-55
8.5.4.2	Error codes	8.5-56
8.5.5	Life guarding / node guarding	8.5-59
8.5.6	The state machine	8.5-60
8.5.6.1	Control word	8.5-61
8.5.6.2	Status word	8.5-62
8.5.6.3	Modes of operation	8.5-64
8.5.7	Description of individual modes	8.5-65
8.5.7.1	Profile Position mode	8.5-65
8.5.7.2	Profile Velocity mode	8.5-68
8.5.7.3	Synchronous mode	8.5-70
8.5.7.4	Homing mode	8.5-71
8.5.7.5	Profile Torque mode	8.5-97
8.5.7.6	Setup mode	8.5-97
8.5.7.7	Automatic Position mode	8.5-98
8.5.7.8	Automatic Single Block mode	8.5-98
8.5.8	Diagnostics and troubleshooting	8.5-99
8.5.8.1	Error and alarm displays on basic unit	8.5-99
8.5.8.2	Evaluation of CBC diagnostic parameter	8.5-103
8.5.8.3	Meaning of CBC diagnostics	8.5-104
8.5.9	CANopen EDS	8.5-106
8.5.10	Parameterization	8.5-107
8.5.10.1	Parameterization for the CBC CANopen with MASTERDRIVES MC_F01 and MASTERDRIVES MC_B-Pos	8.5-107
8.5.11	Logical interconnections for control and status words	8.5-108
8.5.12	General plans of interconnections in MASTERDRIVES MC	8.5-116
8.5.13	Terms and abbreviations	8.5-133

9	TECHNOLOGY OPTION F01	9-1
9.1	Enabling technology option F01	9-1
9.2	Overview of the documentation	9-1
9.3	Application areas	9-3
9.3.1	General functions.....	9-3
9.3.2	Positioning	9-5
9.3.3	Synchronization	9-7
9.3.4	Technology functions already included in the standard software	9-13
9.3.5	Seamless Integration in SIMATIC Automation Solutions	9-15
9.4	Brief description of the technology functions	9-16
9.4.1	Overview of the function diagrams	9-16
9.4.2	Integrating the technology into the basic unit [801].....	9-17
9.4.3	General information on position encoder evaluation [230] ... [270].....	9-17
9.4.4	Resolver evaluation [230]	9-19
9.4.5	Optical sin/cos encoder [240]	9-20
9.4.6	Multiturn encoder evaluation [260, 270]	9-22
9.4.7	Pulse encoder evaluation [250, 255]	9-32
9.4.8	Position sensing system for motor encoder [330]	9-35
9.4.9	Using absolute encoders for positioning of motors with load-side gearing and rotary axis	9-48
9.4.10	Linear axis with absolute encoder when the traversing range is greater than the display range of the encoder	9-52
9.4.11	Position sensing system for external machine encoder [335]	9-53
9.4.12	Position control system [340].....	9-54
9.4.13	Technology overview and mode manager [802]	9-57
9.4.14	Machine data [804]	9-59
9.4.15	Parameter download file POS_1_1 [806]	9-60
9.4.16	Positioning control signals [809]	9-61
9.4.17	Positioning status signals [811]	9-61
9.4.18	Digital I/Os for positioning [813].....	9-62
9.4.19	Evaluation and control of the position sensing system, simulation mode [815]	9-62
9.4.20	Setpoint output and enabling [817]	9-63
9.4.21	Faults, warnings, diagnostics [818]	9-63
9.4.22	Setup mode [819]	9-64
9.4.23	Homing mode [821]	9-65
9.4.23.1	Homing with homing switch only	9-68
9.4.23.2	Homing with encoder zero mark only	9-69
9.4.23.3	Use of a reversing switch during homing	9-69
9.4.24	MDI mode [823]	9-70
9.4.25	Control mode [825]	9-74
9.4.26	Automatic and automatic single-block mode [826, 828].....	9-75
9.4.27	Roll feed [830].....	9-75
9.4.28	Synchronization mode - overview [831]	9-76
9.4.29	Virtual master axis [832]	9-84
9.4.30	Real master with deadtime compensation [833]	9-85
9.4.31	Engaging/disengaging cycle [834].....	9-86
9.4.32	Gearbox function [835]	9-91
9.4.33	Generation of the position setpoint [836].....	9-92

9.4.34	Catch-up function [837]	9-92
9.4.35	Cam [839]	9-93
9.4.36	Synchronization to master value [841]	9-99
9.4.37	Displacement angle setting [841]	9-100
9.4.38	Position correction [843]	9-101
9.4.39	Referencing "on the fly" for synchronization [843]	9-102
9.4.40	Docking point between basic positioner [789b] and synchronism [836]	9-103
9.4.41	Continue synchronism	9-109
9.5	Communication with the technology	9-110
9.5.1	Process data communication (PZD)	9-110
9.5.2	Parameter transfer (PKW)	9-113
9.5.3	Standard function blocks for PROFIBUS-DP and USS	9-114
9.5.4	Additionally available SIMATIC S7 software	9-115
9.5.5	USS interface	9-117
9.5.6	SIMOLINK	9-118
9.6	Configuration	9-120
9.6.1	Encoders for the position sensing system	9-120
9.6.2	Requirements of position encoders for rotary axes	9-121
9.6.3	Brake controller	9-122
9.7	Application examples	9-123
9.7.1	Positioning of a linear axis via PROFIBUS	9-123
9.7.2	Positioning and synchronization with virtual master axis (suitable for self-study)	9-123
9.7.2.1	Task description	9-123
9.7.2.2	Overview diagram	9-124
9.7.2.3	Connection of digital inputs	9-127
9.7.2.4	Connection and parameters of the position sensing system	9-130
9.7.2.5	Velocity normalization P353 [20.5] and P205 [340.2]	9-131
9.7.2.6	Machine data input U501 and U502 [804]	9-132
9.7.2.7	Connecting the technology to the speed and position controllers	9-134
9.7.2.8	Setting the parameters for the positioning modes	9-135
9.7.2.9	Testing the positioning functions of the application example	9-136
9.7.2.10	Setting the parameters for the virtual master axis	9-137
9.7.2.11	Testing the virtual master axis	9-138
9.7.2.12	Configuring the synchronization function	9-138
9.7.2.13	Configuring the SIMOLINK master	9-139
9.7.2.14	Setting the parameters for drive 1 (SIMOLINK slave)	9-140
9.7.2.15	Testing the synchronization in the application example	9-140
9.7.3	Synchronism with the virtual master axis by means of clock-synchronized Profibus (suitable for private study)	9-142
9.7.4	Roll feed	9-151
9.7.5	Application using the SIMATIC S7 GMC software	9-151
9.8	Commissioning the technology	9-151
9.8.1	Measurement and diagnostics resources	9-151
9.8.2	Commissioning the technology	9-152
9.8.3	Checking the speed/position controller	9-153
9.8.4	Defining the actual speed value normalization	9-154
9.8.5	Commissioning the MASTERDRIVES basic functions	9-155
9.8.6	Defining the length unit LU	9-155

9.8.7	Defining the actual value weighting factor (AVWF)	9-156
9.8.8	Defining the maximum traversing velocity	9-158
9.8.9	Procedure for using the "GMC-BASIC" S7 software	9-159
9.8.10	Defining the positioning input signals	9-160
9.8.11	Defining the positioning status signals	9-161
9.8.12	Connection and parameters of the position sensing system	9-162
9.8.13	Machine data input MD1...MD50	9-164
9.8.14	Connecting the technology to the speed and position controller	9-165
9.8.15	Setting the parameters for the positioning modes	9-166
9.8.16	Safety information, hardware limit switches	9-167
9.8.17	Commissioning the positioning modes	9-168
9.8.18	Configuring and testing the virtual master axis	9-170
9.8.19	Setting the parameters for the synchronization block	9-171
9.8.20	Configuring and testing the SIMOLINK drive connection	9-173
9.8.21	Testing the synchronization functions	9-173
9.8.22	Help, my axis won't start!	9-174
9.8.23	General commissioning information	9-175
9.9	Faults, warnings, diagnostics	9-177
9.10	Hardware and software replacement measures	9-178
9.11	Modification history of technology option F01	9-179
9.11.1	Software version V1.0	9-179
9.11.2	Software version V1.1	9-179
9.11.3	Software version V1.2	9-180
9.11.4	Software version V1.3	9-182
9.11.5	Software version V1.4	9-185
9.11.6	Software version V2.1	9-188
9.11.7	Software status V2.2	9-190
9.11.8	Software status V2.3	9-190
9.11.9	Software status V2.4	9-191
9.12	References, software products and accessories	9-193
10	CONTROL WORD AND STATUS WORD	10-1
10.1	Description of the control word bits	10-1
10.2	Description of the status word bits	10-9

11	ENGINEERING INFORMATION	11-1
11.1	Clarification of the type of drive, technical data and other border conditions	11-2
11.2	Specification of the travel curve	11-3
11.3	Calculation of the maximum speed under load and the maximum load torque, selection of the gear	11-4
11.4	Selection of the motor	11-12
11.5	Selection of converters or inverters	11-17
11.6	Selection of the rectifier unit for multi-axis drives	11-19
11.7	Selection of the braking units and braking resistors	11-21
11.8	Selection of other components	11-22
11.9	Calculating example	11-25
11.9.1	Calculation of the x-axis as the travel gear	11-25
11.9.2	Calculating the y-axis as the travel gear	11-34
11.9.3	Calculating the z-axis as the lifting drive	11-35
11.9.4	Selection of the rectifier unit	11-43
11.9.5	Selection of the braking resistor	11-44
11.10	Power Extension PIN F02 (from firmware version 2.20 and higher)	11-45

Annex

Function Diagrams

Parameter Lists

Faults and Alarms

Lists of Stored Motors

Dimension Drawings

The following editions have been published so far:

Edition	Internal Item Number
AA	A5E00192726
AB	A5E00192726
AC	A5E00192726
AD	A5E00192726
AE	A5E00192726
AF	A5E00192726
AG	A5E00856428

Version AG consists of the following chapters:

Chapter		Changes	Pages	Version date
1	System Description	first edition	4	05.2003
2	Configuration and Connection Examples	reviewed edition	17	02.2004
3	Instructions for Design of Drives in Conformance with EMC Regulations	first edition	26	05.2003
4	Function Blocks and Parameters	first edition	9	05.2003
5	Parameterization	reviewed edition	43	01.2006
6	Parameterizing Steps	reviewed edition	44	05.2006
7	Functions	reviewed edition	77	05.2006
8	Communication	first edition	1	05.2003
8.1	Universal Serial Interface (USS)	reviewed edition	43	02.2004
8.2	PROFIBUS	reviewed edition	141	05.2006
8.3	SIMOLINK	reviewed edition	27	05.2006
8.4	CBC Communications Board	reviewed edition	60	10.2004
8.5	CBC CANopen communication board	reviewed edition	133	02.2004
9	Technology Option F01	reviewed edition	195	05.2006
10	Control Word and Status Word	reviewed edition	14	05.2006
11	Engineering Information	reviewed edition	46	06.2005
	Function Diagrams	reviewed edition	211	05.2006
	Parameter Lists	reviewed edition	327	05.2006
	Faults und Alarms	reviewed edition	36	05.2006
	Lists of Stored Motors	reviewed edition	11	05.2006
	Dimension Drawings	reviewed edition	10	05.2003

We reserve the right to make changes to functions, technical data, standards, drawings and parameters.

The reproduction, transmission or use of this document or its contents is not permitted without express written authority. Offenders will be liable for damages. All rights, including rights created by patent grant or registration of a utility model or design, are reserved.

We have checked the contents of this document to ensure that they coincide with the described hardware and software. However, differences cannot be completely excluded, so that we do not accept any guarantee for complete conformance. However, the information in this document is regularly checked and necessary corrections will be included in subsequent editions. We are grateful for any recommendations for improvement.

SIMOVERT® Registered Trade Mark

www.ElectricalPartManuals.com



www.ElectricalPartManuals.com

Siemens AG
Automation and Drives
Motion Control Systems
P.O. Box 3180, D – 91050 Erlangen
Germany

www.siemens.com/motioncontrol

© Siemens AG 2006
Subject to change without prior notice
Order No.: 6SE7087-6QX70

Printed in Germany

1 System Description

1.1 Overview

The SIMOVERT MASTERDRIVES MC (Motion Control) belongs to the SIMOVERT MASTERDRIVES product group. This product group represents an overall modular, fully digital component system for solving all drive tasks posed by three-phase drive engineering. The availability of a high number of components and the provision of various control functionalities enable it to be adapted to the most diversified applications.

Control functionality The control functionality is determined by the software stored in the inverter and converter modules. The following different control versions are provided within the SIMOVERT MASTERDRIVES product group:

- ◆ Vector control (VC)
Vector control for higher demands regarding accuracy and dynamic performance
- ◆ Motion control (MC)
Vector control for servo-drive applications, as an option with higher-level technology functions

Components

The SIMOVERT MASTERDRIVES product group comprises the following components:

- ◆ Accessories Complete converters
- ◆ Inverters
- ◆ Rectifier units (RU)
- ◆ Rectifier/regenerative feedback units (RU, AFE)
- ◆ Braking units and braking resistors
- ◆ Capacitor module (CM)
- ◆ Linking module
- ◆ DC link bus module
- ◆ Line filters
- ◆ Input reactors
- ◆ Fuses
- ◆ Optional boards:
 - Sensor boards (SBx) for speed and position sensing
 - Communication boards (CBx) for field bus interfacing
 - SIMOLINK (SLx) for fast transmission of setpoints and actual values
- ◆ Software options
- ◆ Accessories

1.2 System description

The control functionality of Motion Control is especially tailored to the demands of servo-drive engineering. The vector current control enables fast current injection into the motor windings in conjunction with short sampling times. The related highly dynamic build-up of the torque provides a good basis for higher-level closed-loop control circuits.

With the closed-loop current control, it is possible to operate both synchronous motors and induction motors. Various types of encoders can be used for sensing the necessary speed and positional signals.

The Motion Control functionality is available both in converter and inverter modules which are designed for a system voltage range of 380 V – 15 % to 480 V + 10 %.

All units are provided with a comprehensive basic functionality which can be expanded, if required, by extensive technology and communication functions by the use of software and hardware options. This enables the units to be adapted to the most diversified conditions of service. All closed-loop control functions are implemented with freely assignable function blocks which can be combined as desired. This enables the software to be flexibly adapted to various applications.

Menu structures stored in the unit software simplify start-up and visualization of the drives in conjunction with various operator control panels. PC-assisted tools enable effective parameter setting and data security.

Performance features

The units with Motion Control functionality have the following performance features:

- ◆ Available as a converter and as an inverter module
- ◆ Output range from 0.5 kW to 250 kW
- ◆ Various configurations possible for multi-axis drives
- ◆ Integrated DC link bus module and fusing
- ◆ integrated "Safe STOP" function (optional)
- ◆ Control functionality with servo-characteristics for synchronous and induction motors
- ◆ Interfacing of various position encoders and tachometers
- ◆ Integrated USS interface for the configuration of simple bus systems
- ◆ Interfacing of various field buses
- ◆ Drive networking with up to 200 nodes via SIMOLINK
- ◆ Integrated technology functions for positioning, synchronism and cam disk
- ◆ Definition of the closed-loop control structures by means of freely assignable function blocks
- ◆ User-friendly start-up and diagnostics procedures
- ◆ Menu prompting
- ◆ Graded operator control and visualization by means of an integrated simple standard operator control panel, a user-friendly operator control panel or via PC
- ◆ Uniform PC-capable programming software (DriveMonitor)
- ◆ In accordance with the currently applicable European standards, CE designation
- ◆ UL/CSA approval

1.3 Construction sizes

The power components (converter, inverter, rectifier unit and regenerative feedback unit) used for the Motion Control functionality are available in three types of construction. With reference to the converter/inverter, control versions are available which are assigned to the following output ranges:

- ◆ Compact PLUS 550 W to 18.5 kW
- ◆ Compact 2.2 kW to 37 kW
- ◆ Chassis 45 kW to 250 kW

1.4 Communication

A differentiated communication concept makes it possible to use the correct communication medium depending on the respective requirement. The following communication interfaces are available:

- ◆ Integrated serial interface(s) with USS protocol for parameter setting, operator control and visualization of the units with OP1S or PC
- ◆ Optional boards for various field bus interfaces (e.g. Profibus DP) for integration in the automation
- ◆ Optional board for connecting up SIMOLINK for fast and synchronous data transfer between technologically connected drives (e.g. angular synchronism).

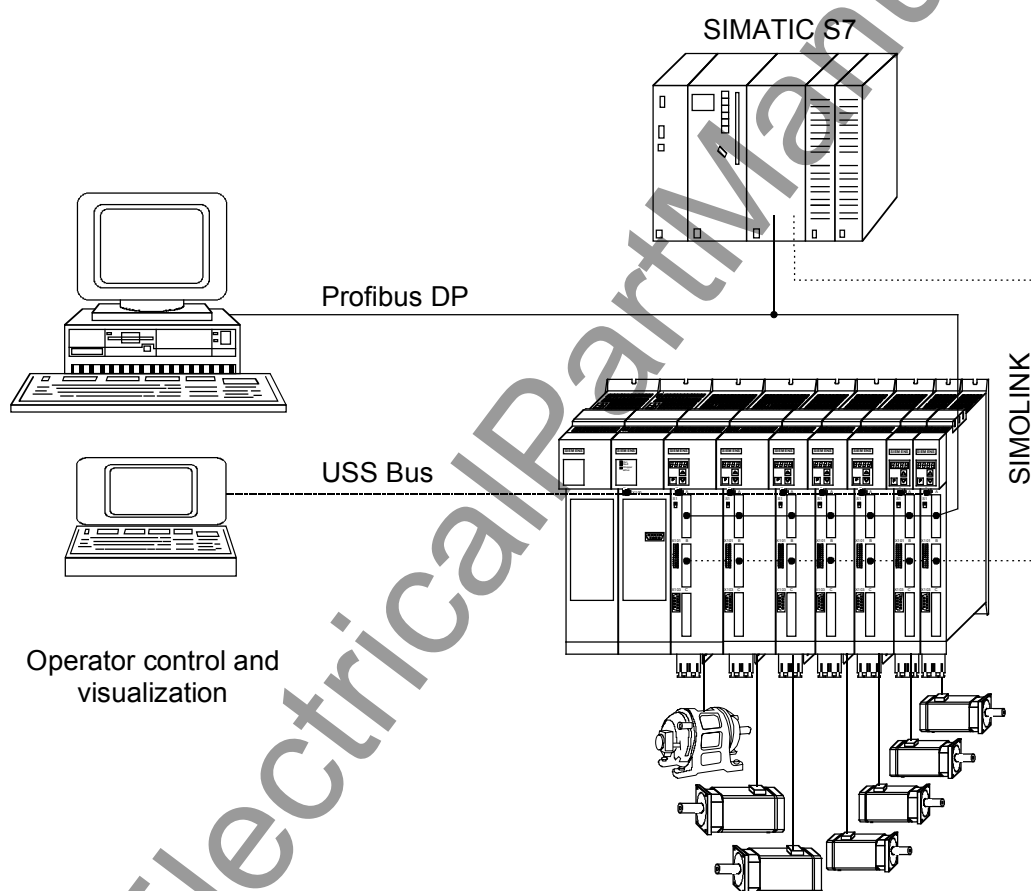


Fig. 1-1

Communication

2 Configuration and Connection Examples

DANGER



The device must be disconnected from its voltage supplies (24 V DC electronics supply and DC link / mains voltage) before the control and encoder leads are connected or disconnected!

2.1 Compact PLUS type units

2.1.1 Single-axis drive

The single-axis drive (see Fig. 2-1 on page 2-2) is used if only single-drive tasks need to be accomplished or if power equalization through several axes is either undesired or not possible.

For this purpose, a converter is used that is directly connected to the 3-phase supply via an external main contactor, a line filter and a line reactor as necessary. Any regenerative energy is stored in the capacitor module or reduced in the braking resistor.

2.1.2 Multi-axis drive up to 3 axes

In the case of multi-axis drives (see Fig. 2-2 on page 2-3) a converter (AC-AC) can be combined with inverters (DC-AC). The converter rectifies the line voltage and supplies the inverters with direct voltage via the DC link bus module. The power supply integrated in the converter further provides the 24 V supply voltage for the electronics of a maximum of 2 inverters.

CAUTION

If more than 2 inverters are connected, the 24 V supply for the electronics must be provided by an external power supply.

The total rated output currents of the inverters supplied by a converter must not exceed the rated output current of the feeding converter (in the case of 6SE7021-0EP50 only half the rated output current).

The regenerative energy generated in one axis can either be used up by the other motors, stored in the capacitor module or reduced in the braking resistor.

2.1.3 Multi-axis drive

In the case of multi-axis drives (see Fig. 2-3 on page 2-4) with more than 3 axes, several inverters are connected to the line voltage via a common rectifier unit.

An external power supply is required for the 24 V supply voltage for the inverter electronics.

The regenerative energy originating in one axis can be used by the other motors, stored in the capacitor module or dissipated in the braking resistor.

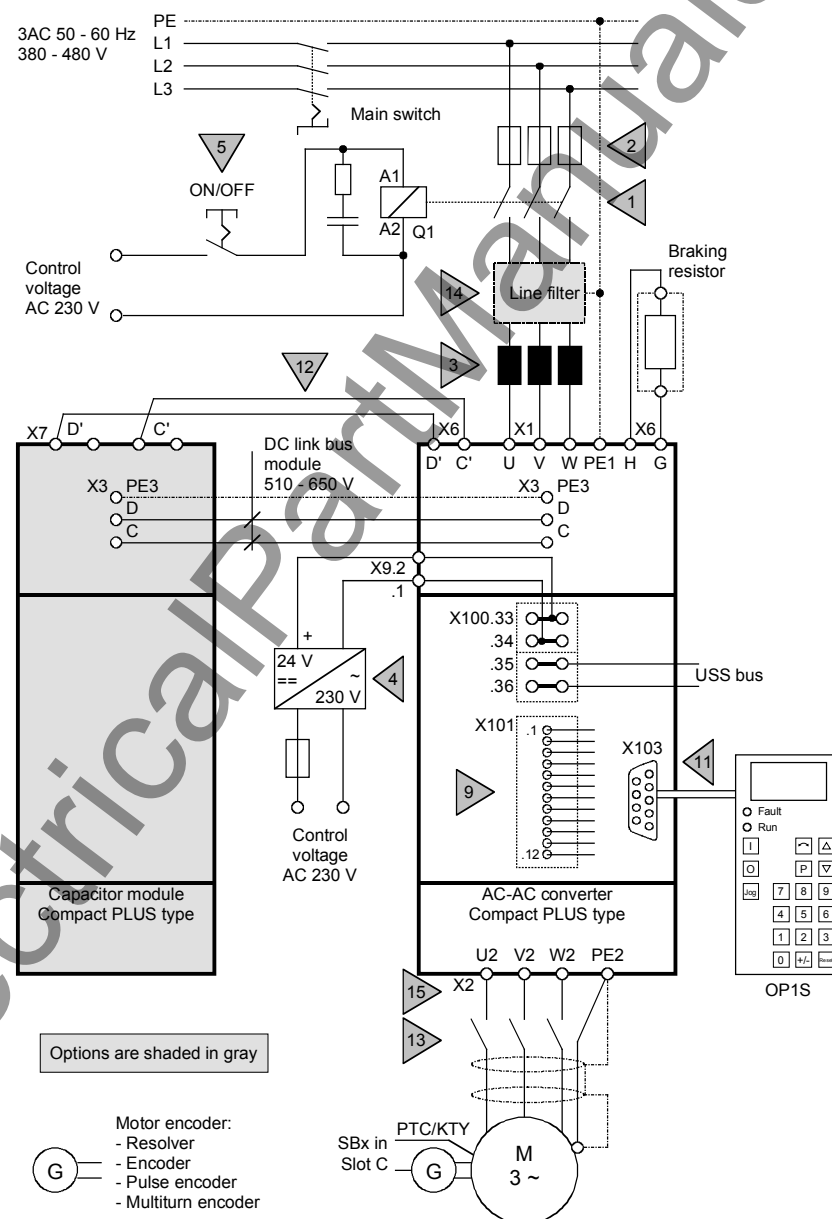


Fig. 2-1 Configuration example of a single-axis drive of the Compact PLUS type

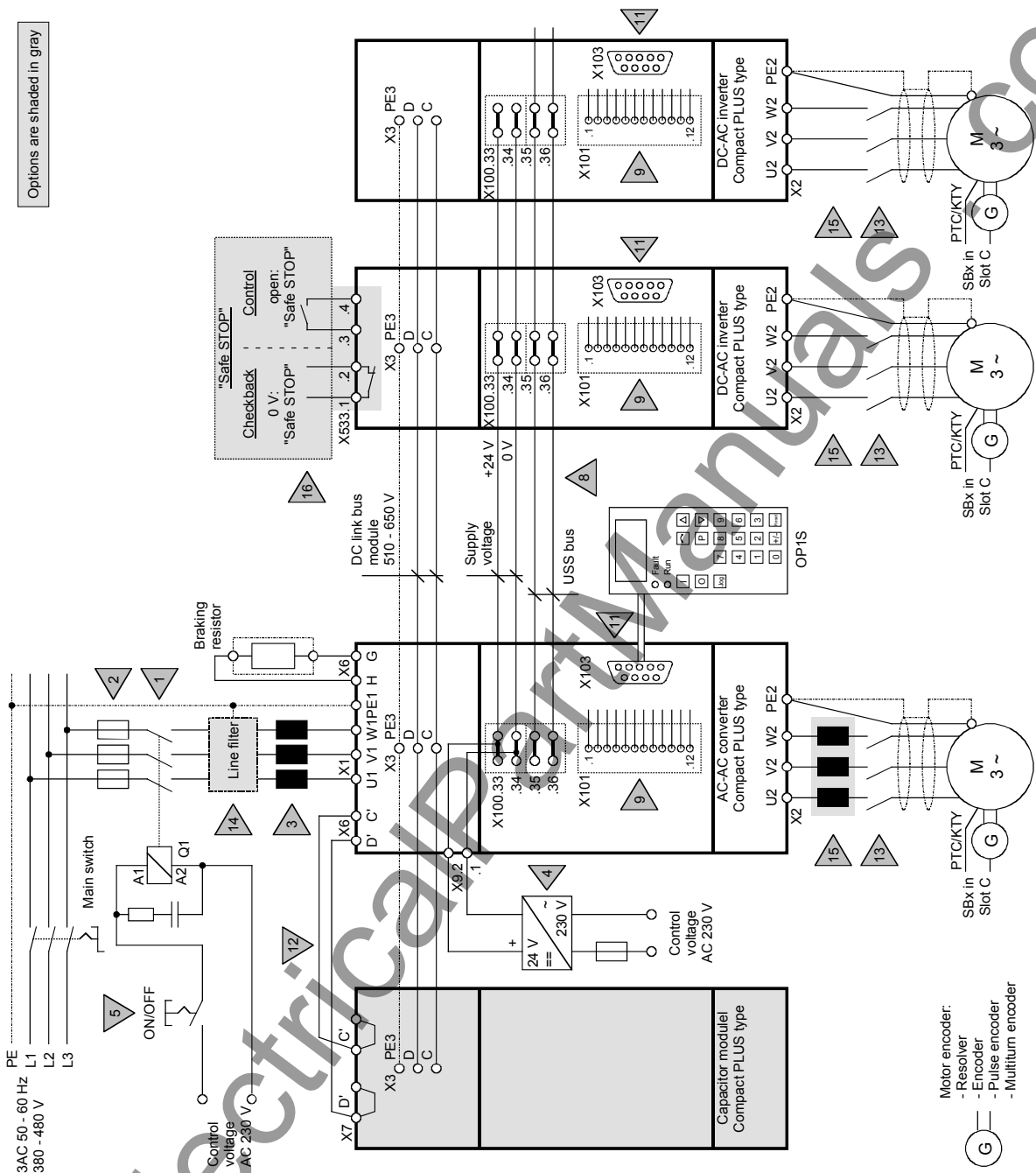


Fig. 2-2

Configuration example of a multi-axis drive with up to 3 axes of the Compact PLUS type

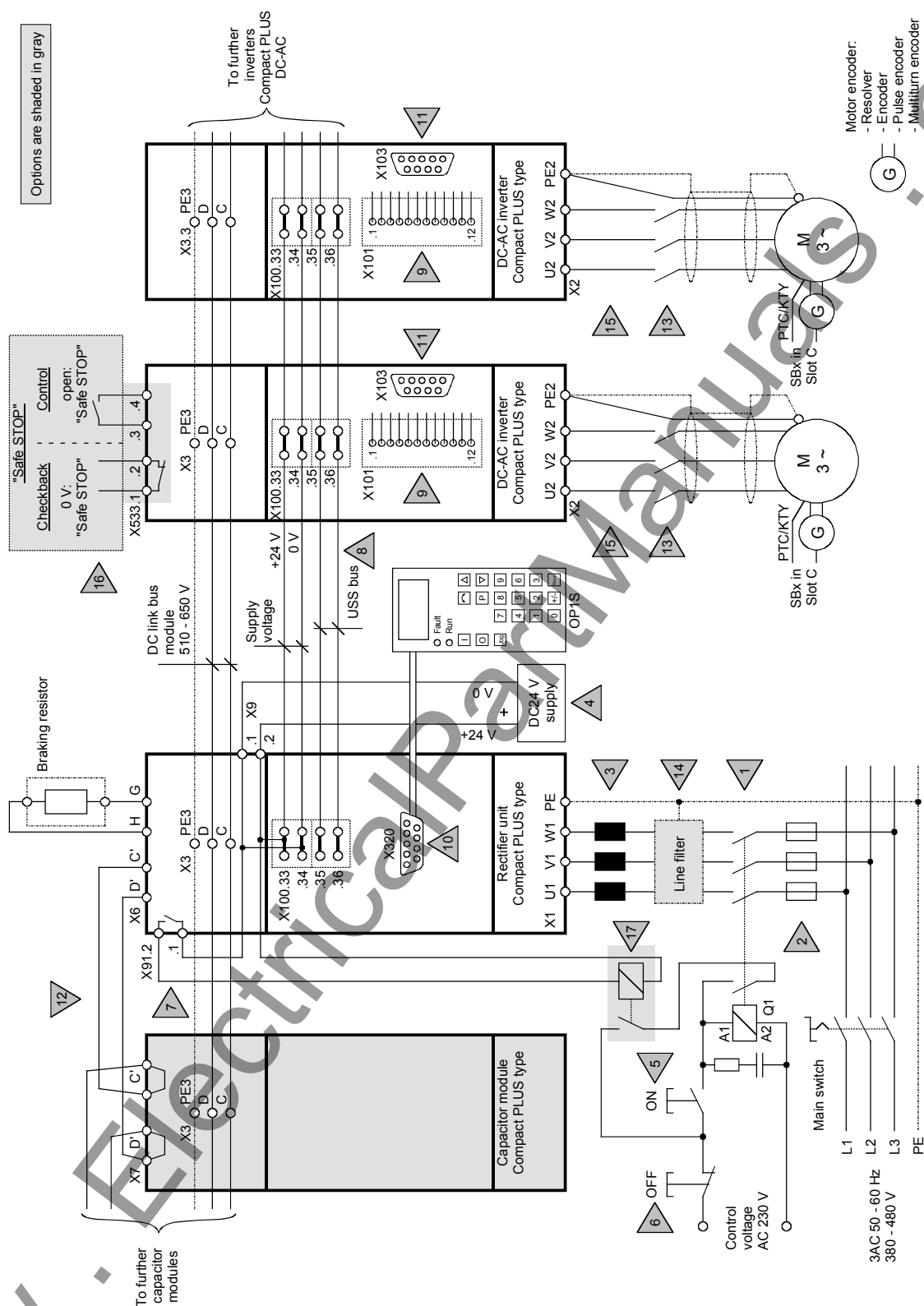


Fig. 2-3

Configuration example of a multi-axis drive with rectifier unit of the Compact PLUS type

2.1.4 Configuration and connection examples (Compact PLUS)

NOTE

The following explanations refer to the numbered gray triangles in Figs. 2-1 to 2-3. These figures are just examples of possible configurations of drives. The necessary individual components have to be clarified according to the specific task.

The information and notes required for dimensioning the individual components and the respective order numbers can be found in the Catalog.

- 1) **Line contactor Q1**
All the equipment is connected to the line via the line contactor, which is used to separate it from the line if required or in the event of a fault. The size of the line contactor depends on the power rating of the connected converter or inverter.
If the line contactor is controlled from the converter, the main contactor checkback time P600 should be set to at least 120 ms.
- 2) **Line fuses**
According to their response characteristic and to suit the requirements, the line fuses protect the connected cables and also the input rectifier of the unit.
- 3) **Line commutating reactor**
The line commutating reactor limits current spikes, reduces harmonics and is necessary for keeping system perturbations to within the limits laid down by VDE 0160.
- 4) **24 V power supply**
The external 24 V supply is used to maintain the communication and diagnostics of the connected-up units even with powered-down line voltage.
The following criteria apply regarding dimensioning:
 - ◆ A current of 1 A must be provided for the rectifier unit, and a current of 2 A for each inverter connected.
 - ◆ When the 24 V supply is powered up, an increased inrush current will be generated that has to be mastered by the power supply.
 - ◆ No controlled power supply unit has to be used; the voltage must be between 20 V and 30 V.
- 5) **ON/OFF**
In the case of a single drive and a multi-axis drive without a rectifier unit, a switch is used to energize or de-energize the line contactor. When they are switched off, the drives are not brought to a controlled standstill, but are braked only by the load.
In the case of a multi-axis drive with a rectifier unit, a pushbutton is used to energize the line contactor. The line contactor is kept energized by means of a lock-type contact connected to the fault signaling relay of the rectifier unit, as long as no fault is detected at the rectifier unit.
- 6) **OFF switch**
Operating the OFF switch causes the line contactor to open immediately.
The drives are not brought to a controlled standstill, but are braked only by the load.

- 7) Fault signaling relay** If a fault occurs in the rectifier unit, a fault message is output via the connecting contacts of the signaling relay.
When the 24 V supply is connected, the relay closes as long as no fault is present.
In the event of a fault, the lock of the line contactor is opened, the contactor drops out and the drives coast down.
- 8) Internal USS bus** The USS bus is used for the internal communication of the units and only has to be connected if it is required.
- 9) X101** The digital inputs and outputs and the analog input and output have to be assigned according to the requirements of the drives.
CAUTION: Terminal X101.1 may **not** be connected with the external 24V supply.
- 10) X320 interface of the rectifier unit** The X320 interface of the rectifier unit serves only for permanently connecting the user-friendly OP1S operator control panel and for connection to the on-line inverters.
Please refer to the relevant operating instructions for the applicable measures and notes for correct operation.
- 11) X103 serial interface** The serial interface is used to connect the user-friendly OP1S operator control panel or a PC. It can be operated either according to the RS232 or the RS485 protocol.
Please refer to the relevant operating instructions for the applicable measures and notes for correct operation.
- 12) Precharging the capacitor module** When a capacitor module is used, the terminals for precharging the capacitors must be connected.
- 13) Output contactor** The use of an output contactor is purposeful if a motor needs to be electrically isolated from the converter/inverter with the DC link charged.
- 14) Line filter** Use of a line filter is necessary if the radio interference voltages generated by the converters or rectifier units need to be reduced.
- 15) Motor supply line** The Siemens cables described in the catalog should be used for connecting the converter and the motor to each other.
The use of output reactors, output filters, sinusoidal and dv/dt filters is not permissible.

The permissible cable lengths are shown in the following tables:

Compact PLUS Inverter DC-AC

Unit MLFB	Pulse frequency 5 kHz		Pulse frequency 10 kHz	
	Shielded cable	Shielded cable	Unshielded cable	Unshielded cable
6SE7012-0TP50	100 m	70 m	50 m	35 m
6SE7014-0TP50	140 m	100 m	70 m	50 m
6SE7016-0TP50	140 m	100 m	70 m	50 m
6SE7021-0TP50	140 m	100 m	70 m	50 m
6SE7021-3TP50	140 m	100 m	70 m	50 m
6SE7021-8TP50	140 m	100 m	70 m	50 m
6SE7022-6TP50	140 m	100 m	70 m	50 m
6SE7023-4TP50	140 m	100 m	70 m	50 m
6SE7023-8TP50	140 m	100 m	70 m	50 m

Compact PLUS Frequency Converter AC-AC

Unit MLFB	Pulse frequency 5 kHz		Pulse frequency 10 kHz	
	Unshielded cable	Shielded cable	Unshielded cable	Shielded cable
6SE7011-5EP50	100 m	70 m	50 m	35 m
6SE7013-0EP50	140 m	100 m	70 m	50 m
6SE7015-0EP50	140 m	100 m	70 m	50 m
6SE7018-0EP50	140 m	100 m	70 m	50 m
6SE7021-0EP50	140 m	100 m	70 m	50 m
6SE7021-4EP50	140 m	100 m	70 m	50 m
6SE7022-1EP50	140 m	100 m	70 m	50 m
6SE7022-7EP50	140 m	100 m	70 m	50 m
6SE7023-4EP50	140 m	100 m	70 m	50 m

The maximum lengths for pulse frequencies between 5 and 10 kHz are obtainable by linear interpolation between the length for 5 kHz and the length for 10 kHz.

16) Safe STOP (Option)

The "Safe Stop" option enables the power supply for the transmission of pulses into the power section to be interrupted by a safety relay. This ensures that the unit will not generate a rotating field in the connected motor.

17) Auxiliary contactor

The auxiliary contactor is used to interrupt the self-holding condition of the main contactor in the event of a fault signal. It must be used if the control voltage for line contactor Q1 is 230 V AC.

The auxiliary contactor is not required if a line contactor with a control voltage of 24 V DC is used.

Braking resistor

The brake choppers are already included in the Compact PLUS rectifier units and converters. Only a suitable external braking resistor has to be connected up, if required.

See also Chapter 11.7.

Encoder cable

You will find preassembled encoder cables in Catalog DA65.11, chapter 3. Please note that different encoder cables are required for encoders and multiturn encoders. If the wrong encoder cable is used for one or the other, fault F051 (during operation) or alarm A018 or A019 is generated.

DANGER

The encoder cable must only be connected and plugged in when the converter is disconnected from the supply (24 V and DC link). Damage to the encoder could result if this advice is not heeded. This especially concerns the multiturn encoder EQN1325. Encoder or encoder cable faults can result in incorrect field orientation and therefore in uncontrolled axis movements.

2.2 Compact and chassis-type units

2.2.1 Water-cooled units

	If you are using water-cooled MASTERDRIVES please note that the permissible operating pressure depends on the construction type.
Type B to G	Operating pressure ≤ 1 bar. Operating pressures above 1 bar not permitted! If the system is to be operated at higher pressure, the pressure on each unit must be reduced to 1 bar initial pressure.
Type $\geq J$	Operating pressure ≤ 2.5 bar. Operating pressures above 2.5 bar not permitted! If the system is to be operated at higher pressure, the pressure on each unit must be reduced to 2.5 bar initial pressure.

2.2.2 Single-axis drive with Compact or chassis-type units

The single-axis drive (see Fig. 2-4 on page 2-9) is used for single-axis applications or where energy balancing over several axes is not wanted or not possible.

In this case, only one converter is used that, where applicable, is connected directly to the three-phase system via an external main contactor, a line filter and a line reactor. Any regenerative energy is dissipated in the braking resistor.

2.2.3 Multi-axis drive with Compact or chassis-type units

In the case of multi-axis drives, several inverters are connected to the line voltage via a common rectifier unit.

A 24 V power supply is needed for the rectifier unit.

The regenerative energy originating in one axis can be used by the other motors, stored in the capacitor module or dissipated in the braking resistor.

◆ Configuration examples:

- Multi-axis drive with Compact units
(see Fig 2-5 on page 2-10)
- Multi-axis drive with chassis-type units
(see Fig. 2-6 on page 2-11)

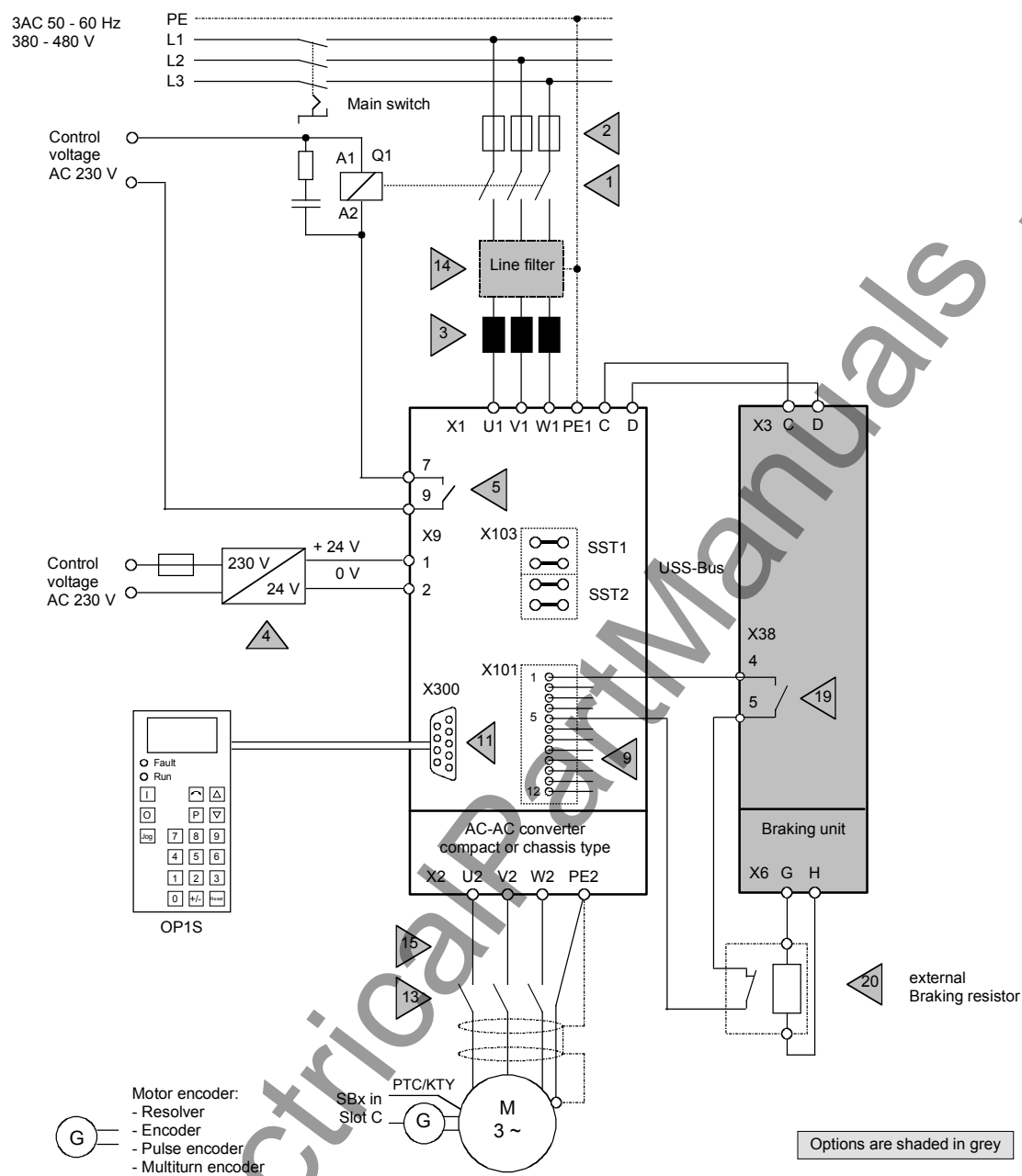


Fig. 2-4

Configuring example: single-axis drive with Compact or chassis-type units

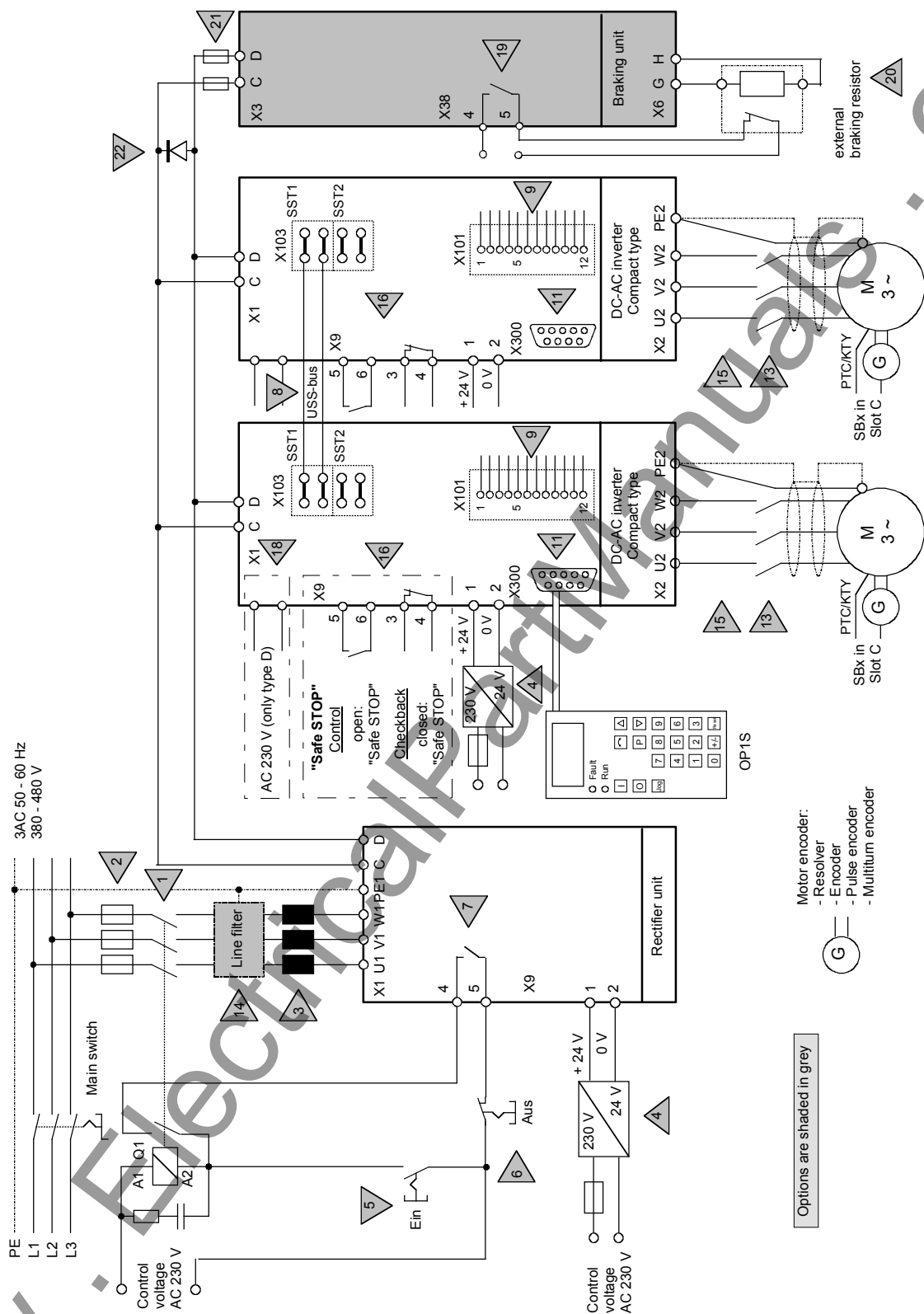


Fig. 2-5

Configuration example: multi-axis drive with Compact units

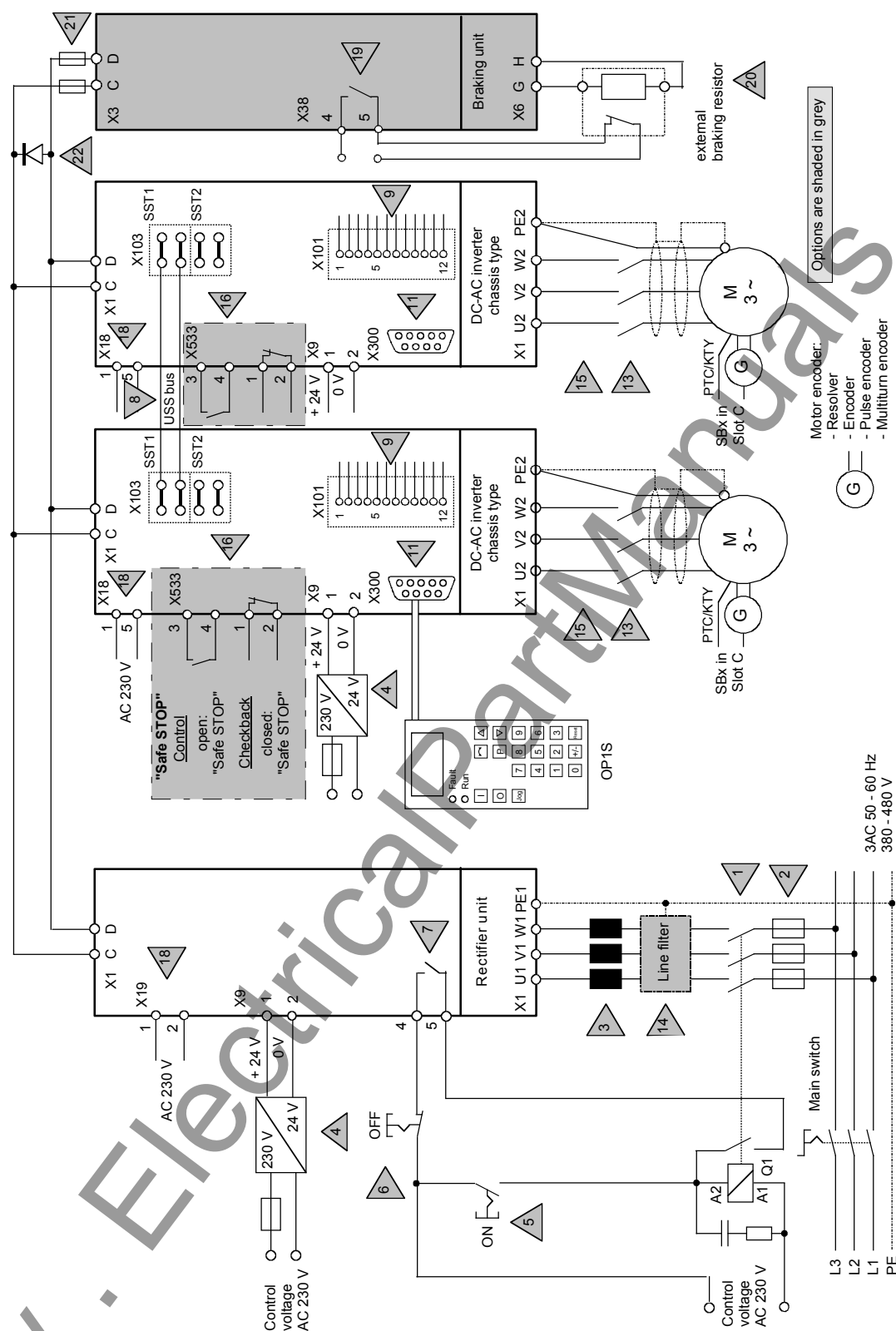


Fig. 2-6

Configuration example: multi-axis drive with chassis-type units

2.2.4 Explanations relating to the configuration examples (Compact and chassis-type units)

NOTE

The following explanations refer to the numbered gray triangles in Figs. 2-4 to 2-6, showing the drives in an example configuration. The application decides which components are required.

In the catalog you will find the necessary information, order numbers and notes concerning the ratings of the individual components.

- 1) **Q1 line contactor** The line contactor is used to connect the entire system up to the supply and disconnect it from the supply if required or in the event of a fault. Dimensioning depends on the output of the connected converters or inverters.
- 2) **Line fuses** According to their response characteristic and to suit the requirements, the line fuses protect the connected cables and also the input rectifier of the unit.
- 3) **Line commutating reactor** The line commutating reactor limits current spikes, reduces harmonics and is necessary for keeping system perturbations to within the limits laid down by VDE 0160.
- 4) **24 V power supply** The external 24 V supply is used to maintain the communication and diagnostics of the connected-up units even with powered-down line voltage. Rectifier units always require an external 24 V supply.
The following criteria apply regarding dimensioning:
 - ◆ Currents (see catalog DA65.11)
 - ◆ When the 24 V supply is powered up, an increased inrush current will be generated that has to be mastered by the power supply.
 - ◆ No controlled power supply unit has to be used; the voltage must be between 20 V and 30 V.
- 5) **ON/OFF** On a single drive, the line contactor is also switched (via X9: 7,9) with the ON/OFF command (e.g. via the control terminal strip). When it is switched off, depending on the parameterization, the drive is brought to a standstill in a controlled manner before the line contactor opens.
If the line contactor (1) is controlled from the converter (via X9:7,9), the main contactor checkback time P600 should be set to at least 120 ms.
In the case of a multi-axis drive with a rectifier unit, a pushbutton is used to energize the line contactor. The line contactor is kept energized by means of a lock-type contact connected to the fault signaling relay of the rectifier unit, as long as no fault is detected at the rectifier unit.
- 6) **OFF switch** The line contactor is opened as soon as the OFF switch is activated.
The drives are not brought to a standstill in a controlled manner; they are braked only by the connected load.

- 7) Fault signaling relay** If a fault occurs in the rectifier unit, a fault message is output via the connecting contacts of the signaling relay.
When the 24 V supply is connected, the relay closes as long as no fault is present.
In the event of a fault, the lock of the line contactor is opened, the contactor drops out and the drives coast down.
- 8) Internal USS bus** The USS bus is used for the internal communication of the units and only has to be connected if it is required.
- 9) X101** The digital inputs and outputs and the analog input and output have to be assigned according to the requirements of the drives.
If the digital inputs are supplied from an external 24 V supply, this must be referenced to frame X101.2.
CAUTION: Terminal X101.1 may **not** be connected with the external 24V supply.
- 11) X300 serial interface** The serial interface is used to connect the user-friendly OP1S operator control panel or a PC. It can be operated either according to the RS232 or the RS485 protocol.
Please refer to the relevant operating instructions for the applicable measures and notes for correct operation.
- 13) Output contactor** The use of an output contactor is purposeful if a motor needs to be electrically isolated from the converter/inverter with the DC link charged.
- 14) Line filter** Use of a line filter is necessary if the radio interference voltages generated by the converters or rectifier units need to be reduced.
- 15) Motor supply line** The use of output reactors, sinusoidal or dv/dt filters is not permissible for MASTERDRIVES MC units.
- 16) Safe STOP (Option for chassis units)** The "Safe STOP" option enables the power supply for the transmission of pulses into the power section to be interrupted by a safety relay. This ensures that the unit will not generate a rotating field in the connected motor.
- 18) Fan supply** An AC 230 V connection is necessary for the fans on all the chassis-type units (via X18: 1, 5).
On Compact units of type D, the voltage has to be connected directly at the fan fuses F101 and F102.
- 19) Monitoring of braking unit** This contact opens if there is a fault in the braking unit. It enables the corresponding converter and the line contactor to be de-energized via a digital input of the X101 control terminal strip and via parameterization at "External fault 2" (P586). The thermal contact can be switched in series if an external braking resistor is used.
On a multi-axis drive, depending on the application, a check has to be made as to where the NC contact has to be looped in. It must be borne in mind that contact X38: 4,5 does not close until the DC link voltage has built up at the braking unit.

20) External braking resistor An external braking resistor can be connected on braking units up to $P_{20} = 20 \text{ kW}$ in order to increase the continuous braking power.

The internal braking resistor has to be disconnected in this case. On braking units where $P_{20} > 20 \text{ kW}$, operation is only possible with an external braking resistor.

21) Fuses for braking unit

The braking units have to be fused on multi-axis drives (see Catalog DA65.11).

22) Free-wheeling diode

If a braking unit is used on multi-axis drives or with strongly varying inverter outputs, a free-wheeling diode has to be used (see Catalog DA65.11).

Encoder cable

You will find preassembled encoder cables in Catalog DA65.11, chapter 3. Please note that different encoder cables are required for encoders and multiturn encoders. If the wrong encoder cable is used for one or the other, fault F051 (during operation) or alarm A018 or A019 is generated.

DANGER



The encoder cable must only be connected and plugged in when the converter is disconnected from the supply (24 V and DC link). Damage to the encoder could result if this advice is not heeded. This especially concerns the multiturn encoder EQN1325. Encoder or encoder cable faults can result in incorrect field orientation and therefore in uncontrolled axis movements.

2.3 Points to look out for when using certain option boards and CUPM

2.3.1 Encoder interface connections

1. **SBM, SBM2 or SBP**

The encoder cable must only be connected or plugged when the converter is de-energized (24 V and DC link). Damage to the encoder can result if this advice is not heeded. This especially applies to the multiturn encoder EQN1325. An encoder or cable fault can cause incorrect field orientation and therefore uncontrolled axis movement on synchronous machines.

2. **SBM2 or SBP**

Only the SBM2 and the SBP are permitted to be used for an **external encoder**. The pulse encoder simulation of the SBM2 is generally switched off (also on the X420 front connector).

3. **SBM2 or SBP or SBR2**

The pulse encoder simulation of the **motor** encoder (always in slot C!) on the backplane bus is always applied for further processing by the technology board T300 or T400.

4. **SBP or SBM2 or SBM**

If the SBP or the SBM2 or the SBM is the only encoder board inserted, this board will always be recognized as evaluation for the motor encoder.

5. **If only one encoder board is inserted, this should be installed in Slot C (otherwise it will not be possible to evaluate the motor temperature).**

6. **SBM or SBM2**

Different encoder cables must be used for encoders and multiturn encoders! Fault F051 (during operation) or alarm A18 or A19 is output if an incorrect encoder cable is used.

7. **SBR2**

As soon as the supply voltage for the electronics is connected, the pulse encoder simulation outputs as many pulses as are needed for the position actual value counter count for a two-pole resolver to correspond to that of the current rotor position (with reference to 1 revolution).

2.3.2 TB boards

1. In the combination **CUPM + CBP + (T100 or T300)** no access to the parameters of the T100 or T300 is possible via the PROFIBUS. This only applies when the CBP is slotted into slot A or slot C.
Remedy: slot CBP into the adapter board (ADB) in slot G. See also Catalog DA65.11, Chapter 6.
2. Basic device parameters cannot be read or altered via the USS interface of the **T100**.
3. With the OP1S, TB parameters (**T100, T300**) can only be read and altered if the parameter number is entered via the numerical keypad. Access to parameter numbers is no longer possible with the "up" and "down" keys if the next parameter does not exist.
4. If the **T300** is used together with the **SLB** in a single unit, The T300 with item no. 477 407 9000 02 must be product version B or higher. All product versions can be used for item no. 477 407 9000 12.
5. If the **T100** is used together with the **SLB** in a single unit, T100 must be product version L8 or higher.

2.3.3 EB boards

**EB1 item no.
477 491 9000 00 with
product vers. A on
slot A of the CUPM**

EB1 connector 5103/monitoring parameters 662.3 (analog input channel3) does **not** display the input of channel 3, but the input of channel 2.

Remedy

Slot EB1 into another slot (adapter board or slot C), or in the case of item no. 477 491 9000 00, use product version B or higher. All product versions can be used with item no. 477 491 9000 10.

3 Instructions for Design of Drives in Conformance with EMC Regulations

3.1 Foreword

The modular design of SIMOVERT MASTERDRIVES permits a large number of possible drive converter/equipment combinations so that it is not practical to provide a separate description for every individual combination here. It is more purposeful for this document to provide basic information and generally applicable rules so that you can configure your particular drive converter/equipment combination in an "electromagnetically compatible" manner.

The drives are operated in widely varying environments and any additionally used components (control systems, switch-mode power sections, etc.) can differ considerably as far as their noise immunity and noise emission levels are concerned. For this reason, it is permissible to deviate from the EMC regulations on a case-to-case basis after individual investigation.

In the context of the EMC Law, SIMOVERT MASTERDRIVES are considered as "components" rather than "units". For a better understanding of these instructions, however, the generally used term "units" is used.

With effect from June 1996, the "EMC product standard including special test methods for electric drive units" EN 61800-3 (VDE 0160 T100, IEC 1800-3) is applicable for frequency converters. Before this product standard came into force, the standards EN 50081 with EN 55011 and EN 50082 with IEC 801 were applicable. These are no longer relevant for frequency converters now that the product standard has come into force.

Please contact your local Siemens office regarding any other queries you may have relating to EMC.

3.2 Principles of EMC

3.2.1 What is EMC?

EMC stands for "ElectroMagnetic Compatibility" and, in accordance with the EMC Law §2(7), it defines "the capability of a unit to operate satisfactorily in an electromagnetic environment, without itself causing electromagnetic disturbances which would be unacceptable for other electrical units in this environment".

In principle, this means that units should not interfere with each other. And this is a feature that you have always looked for in your electrical products!

3.2.2 Noise emission and noise immunity

EMC is dependent on two characteristics of the units concerned - the emitted noise and the noise immunity. Electrical equipment can either be treated as a noise source (transmitter) and/or a noise receiver. Electromagnetic compatibility exists when the existing interference sources do not affect the function of the noise receivers. It is also possible for a unit to be both an interference source and an interference receiver at the same time. For example, the power section of a frequency converter can be regarded as a noise source, whereas the control section can be regarded as a noise receiver.

The **noise emission** of frequency converters is governed by the European Standard EN 61800-3. The cable-related noise at the mains connection is measured under standard conditions as radio interference voltage. Electromagnetically emitted noise is measured as radio interference (radiated noise). The standard defines limit values "First environment" (public supply networks) and "Second environment" (industrial networks).

When the equipment is connected up to the public supply, the maximum harmonics specified by the local power supply company must be observed.

The **noise immunity** of a unit describes how it behaves when subjected to electromagnetic noise/interference. The requirements and evaluation criteria for the behaviour of the electrical units are also laid down in standard EN 61800-3.

3.2.3 Industrial and domestic applications

Limit values are laid down for emitted noise and noise immunity depending on the application for which the units are envisaged. A differentiation is made between industrial and domestic environments. In industrial environments, the noise immunity of the units must be very high, but lower requirements are made concerning the emitted noise. In domestic environments, i.e. when connected to public supply systems, there are strict regulations concerning emitted noise but, on the other hand, the units can be designed with a lower noise immunity.

If the drive is an integral part of a system, it does not initially have to satisfy any demands regarding emitted noise and noise immunity. However, the EMC Law specifies that a system must as a whole be electromagnetically compatible within its environment. Within the system, the owner will, in his own interest, make sure that his equipment is electromagnetically compatible.

Without a radio interference suppression filter, the emitted noise of the SIMOVERT MASTERDRIVES frequency converters exceeds the limit value "First environment". Limit values are currently still under discussion for the "Second environment" sector (see EN 61800-3 section 6.3.2). However, their high noise immunity makes them insensitive to the noise emitted by units in their vicinity. If all control components of the system (e.g. automation devices) have a noise immunity suitable for industrial environments, then it is not necessary for every drive to maintain this limit value.

3.2.4 Non-grounded systems

In some industrial sectors, non-grounded supplies (IT supplies) are used to increase the availability of the plant/installation. In the event of a ground fault, no fault current flows, and the plant can still produce. However, when a radio interference suppression filter is used, a fault current will flow when a ground fault occurs, which may cause shutdown of the drives or even the destruction of the radio interference suppression filter. In order to minimize this fault current, the radio interference suppression filter has to be designed differently which will quickly reach the physical limits. Radio interference suppression filters additionally affect the concept of non-grounded supply networks and can thus result in a safety risk when used with these networks (see Product Standard EN 61800-3: 1996). If required, radio interference suppression should thus be realized at the grounded primary side of the supply transformer or with a single special filter at the secondary side. The special filter also generates leakage currents to ground. A ground-leakage monitor which is usually used in non-grounded systems has to be adjusted to the special filter.

3.3 The frequency converter and its electromagnetic compatibility

3.3.1 The frequency converter as a noise source

Mode of operation of SIMOVERT MASTERDRIVES

SIMOVERT MASTERDRIVES frequency converters operate with a voltage-source DC link.

In order to keep the power losses as low as possible, the inverter switches the DC link voltage to the motor winding in the form of voltage blocks.

An almost sinusoidal current flows in the motor.

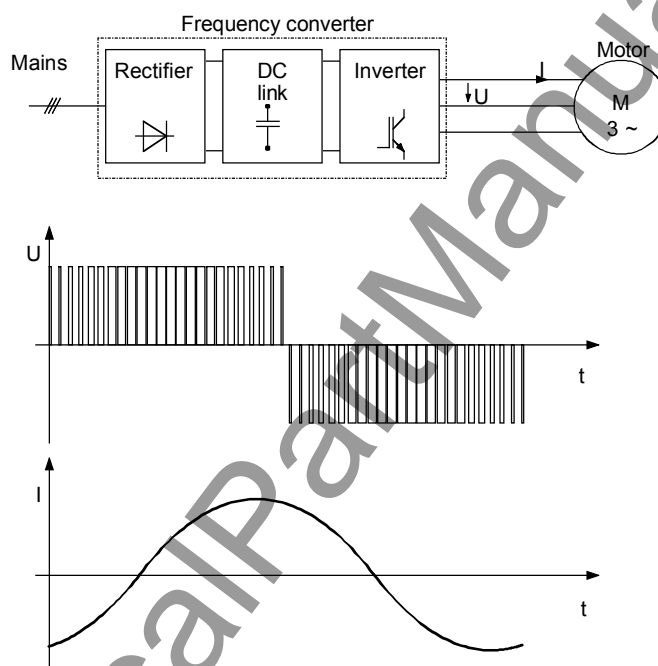


Fig. 3-1 Block diagram showing output voltage V and motor current I of a frequency converter

The described mode of operation in conjunction with high-performance semiconductor switching elements have made it possible to develop compact frequency converters which now play a vital role in drive technology.

As well as having many advantages, the fast semiconductor switches also have one disadvantage:

A pulse-type noise current flows to ground through parasitic capacitances C_P at each switching edge. Parasitic capacitances exist between the motor cable and ground, and also within the motor.

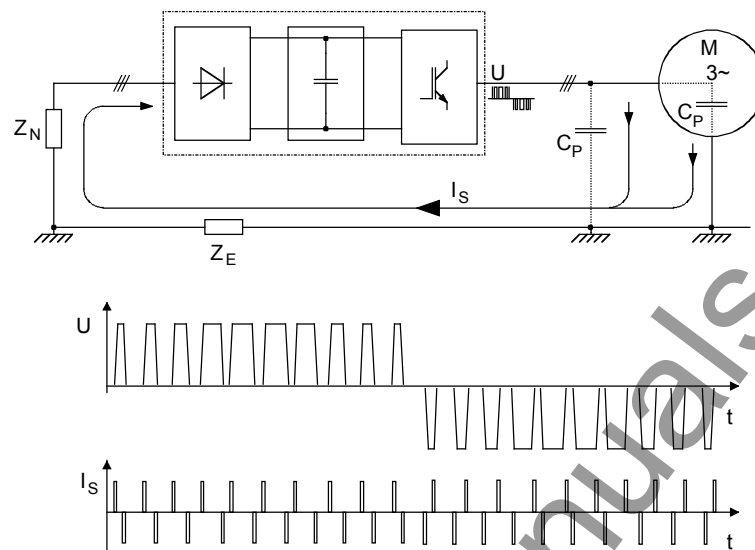


Fig. 3-2 Block diagram showing output voltage U and fault current I_s

The source of the fault current I_s is the inverter, which is the reason why the fault current must also flow back to the inverter. Impedance Z_N and ground impedance Z_E act in the return flow path. Impedance Z_N forms parasitic capacitances between the supply cable and ground, which is connected in parallel with the impedance (between phase and ground) of the supply transformer. The noise current itself and the voltage drops across Z_N and Z_E caused by the noise current can also affect other electrical units.

Frequency converters generate the high-frequency noise currents which have already been described. In addition, low-frequency harmonics should be taken into account. As a result of rectification of the line supply, a non-sinusoidal line current is drawn which causes a distortion of the line supply voltage.

Low-frequency harmonics are reduced using **line reactors**.

The high-frequency noise emission can only be reduced if the generated noise current is correctly routed. Using non-shielded motor cables, the noise current flows in an undefined fashion back to the frequency converter, e.g. via foundation/base frame grounders, cable ducts, cabinet frames. These current paths have a very low resistance for currents with a frequency of 50 or 60 Hz. However, the noise current induces a high-frequency component, which can result in problematical voltage drops.

A **shielded motor cable** is absolutely necessary to enable the fault current to flow back to the frequency converter in a defined fashion. The shield must be connected to the housing of the frequency converter and to the motor housing through a large surface area. The shield now forms the easiest path for the noise current to take when returning to the frequency converter.

Measures to reduce noise emission

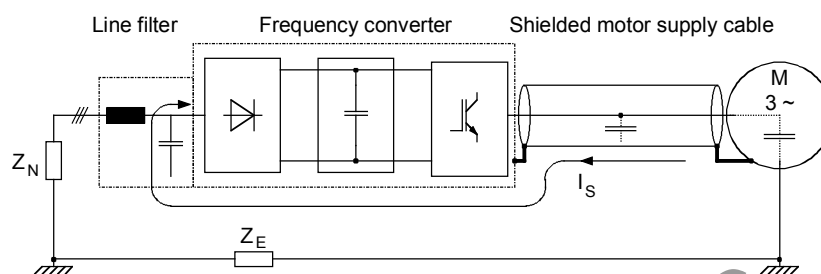


Fig. 3-3 Flow of the noise current with shielded motor cable

A shielded motor cable with a **shield connected at both sides** causes the noise current to flow back to the frequency converter through the shield.

Although (almost) no voltage drop arises across impedance Z_E for shielded motor cables, the voltage drop across impedance Z_N can affect other electrical units.

For this reason, a **radio interference suppression filter** should be installed in the supply feeder cable to the frequency converter. Arrangement of the components as per the following figure.

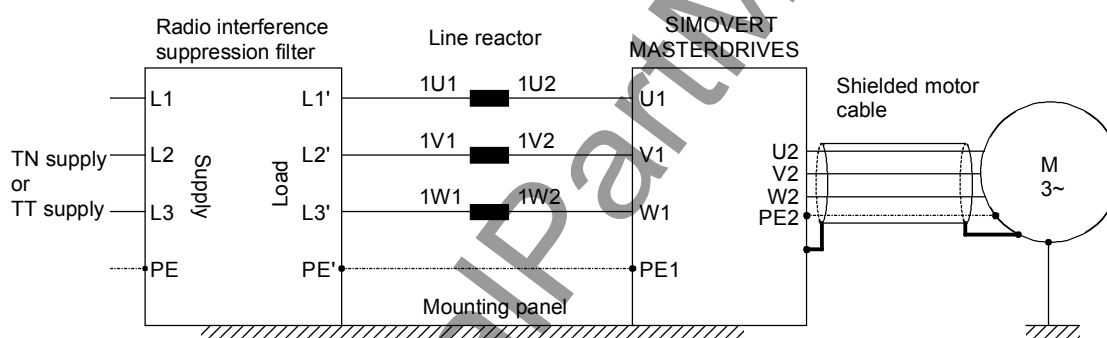


Fig. 3-4 Arrangement of the components

Radio interference suppression filters and frequency converters must be connected through a low-ohmic resistance for the high-frequency noise currents. In practice, this requirement is best satisfied by mounting the frequency converters and radio interference suppression filters on a common panel. Frequency converters and radio interference suppression filters must be connected to the mounting panel through the largest possible surface area.

The SIMOVERT MASTERDRIVES must be installed in an enclosed **cabinet** in order to limit the radio interference radiation. In particular, the radio interference radiation is determined by the control section with its microprocessor and it is therefore comparable with the noise emitted from a computer. If there are no radio transmission services in the immediate vicinity of the SIMOVERT MASTERDRIVES, there is no need for a high-frequency-sealed cabinet.

Radio interference radiation is not limited if the units are installed in racks. In this case, adequate shielding should be provided by suitably designing the equipment room/area.

3.3.2 The frequency converter as a noise receiver

Ways in which noise is received

Noise can enter a unit either galvanically, inductively or capacitively. The equivalent circuit diagram shows a noise source which causes noise current I_S in the unit due to capacitive coupling effects. The magnitude of the coupling capacitance C_K is determined by the cabling and the mechanical design.

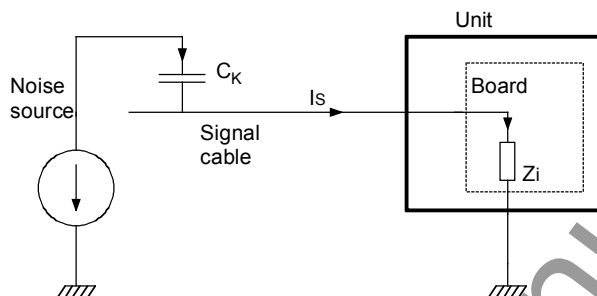


Fig. 3-5 Capacitive coupling for non-shielded signal cables

Noise current I_S produces a voltage drop across impedance Z_i . If the noise current flows through a board with fast electronic components (e.g. microprocessor), even a small spike in the μs area and an amplitude of just a few volts can lead to disturbing noise.

Measures to increase noise immunity

The most effective way of preventing noise being coupled-in is to rigorously **separate power and signal cables**.

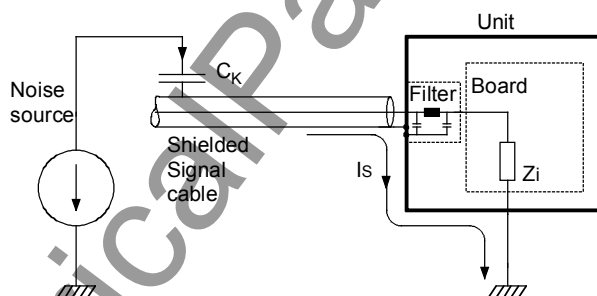


Fig. 3-6 Increasing the noise immunity by using shielded signal cables

The inputs and outputs of the SIMOVERT MASTERDRIVES control section are fitted with filters that keep noise currents I_S separate from the electronics. The filters also smooth the useful signal. In the case of signal cables with extremely high-frequency signals, e.g. from the digital tachometer, this smoothing has a disturbing effect. As no smoothing is possible on account of its functionality, **shielded signal cables** have to be used here. The noise current now flows back to the noise source via the shield and the housing.

The shields of **digital signal cables** always have to be connected at both ends, i.e. at the transmitter and at the receiver!

In the case of **analog signal cables**, low-frequency noise can arise if the shield is connected at both ends (hum is coupled-in). In this case, the shield must only be connected at one end at the SIMOVERT MASTERDRIVES. The other end of the shield should be grounded through a capacitor (e.g. 10 nF/100 V type MKT). This capacitor enables the shield to be connected at both ends after all as far as high-frequency noise is concerned.

3.4 EMC planning

If two units are not electromagnetically compatible, the noise radiated by the noise transmitter can be reduced, or the noise immunity of the noise receiver can be increased. Noise sources are often power electronic units with a large current drain. Complex filters are necessary to reduce their noise emission. Noise receivers especially include control devices and sensors/transmitters, as well as their evaluation circuit. Not so much effort and cost is required to increase the noise immunity of low-power units. In industrial environments, it is therefore more cost-effective to increase the noise immunity than to reduce the noise emission.

To maintain the "Second environment" limit value class specified in EN 55011, the radio interference voltage at the mains connection point can be a maximum of 79 dB (μV) between 150 kHz and 500 kHz, and a maximum of 73 dB (μV) between 500 kHz und 30 MHz. When expressed in volts, these values are 9 mV and 4.5 mV respectively!

Before radio interference measures can be applied, it must first be clarified at which locations you or your customer require EMC. See the following example:

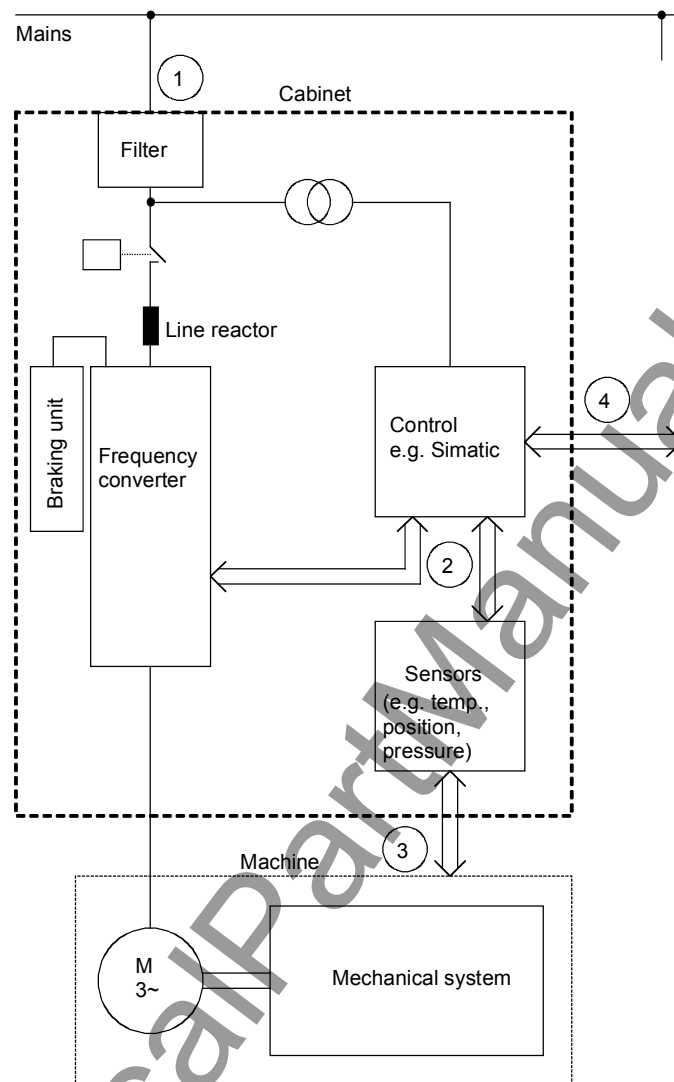


Fig. 3-7 Block diagram of a drive system

The purpose of a frequency converter is to drive a motor. The frequency converter, the relevant open-loop control and sensor system are accommodated in a cabinet. The emitted noise has to be limited at the mains connection point and therefore radio interference suppression filters and line reactors are installed in the cabinet.

Assuming that all requirements are met at Point ① - can it be supposed that electromagnetic compatibility exists?

This question cannot just be answered with "yes" because EMC has to be reliably ensured inside the cabinet as well. It is possible that the control system produces electromagnetic influences at interfaces ② and ④, and the sensor system at interfaces ② and ③.

Therefore, a radio interference suppression filter by itself cannot ensure EMC!

See the following sections.

3.4.1 The zone concept

The most cost-effective measure of reducing interference is to spatially separate the noise sources and the noise receivers. This must, however, already be taken into account during the planning stage of a machine/system. The first question that has to be answered is whether the unit used is a noise source or a noise receiver. Noise sources in this connection are, for example, frequency converters, braking units, contactors.

Noise receivers are, for example, automation devices, encoders and sensors.

The machine/system is then divided up into EMC zones and the units are assigned to these zones. Each zone has its own requirements regarding noise emission and noise immunity. The zones have to be spatially separated, which is best done using a metal housing or, within a cabinet, using grounded partitions. If necessary, filters have to be used at the zone interfaces. The zone concept is explained using the following diagram as an example which shows a simplified drive system:

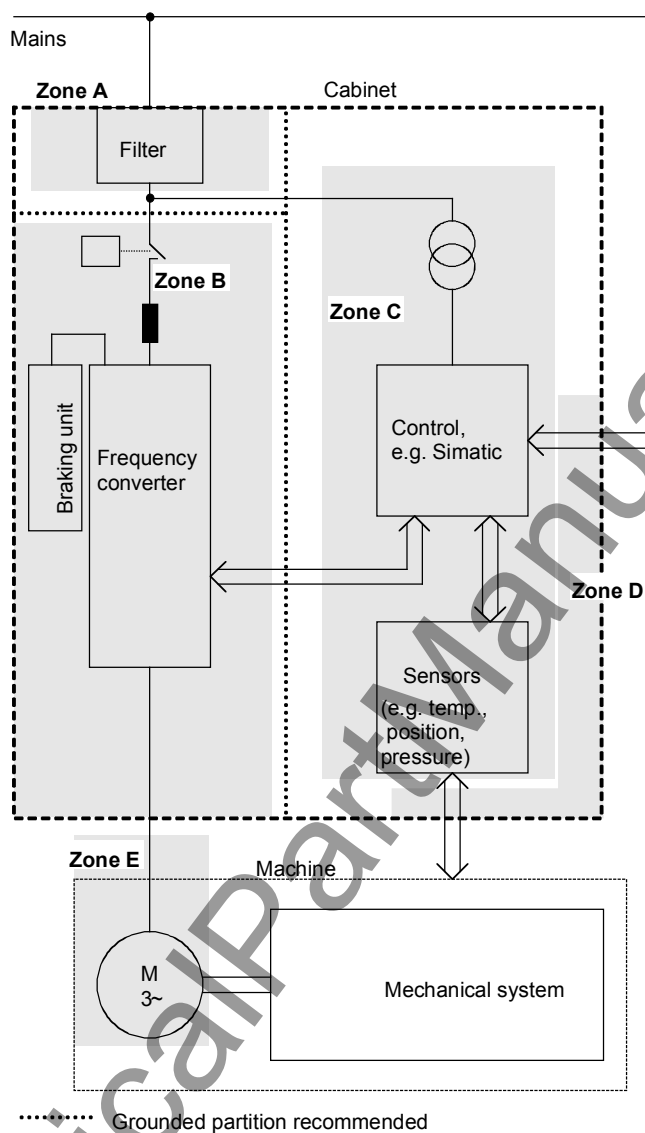


Fig. 3-8 Sub-dividing a drive system into zones

- ◆ Zone A is the cabinet connection to the line supply including filter. The emitted noise should be kept at specific limit values here.
- ◆ Zone B contains the line reactor and the noise sources: frequency converter, braking unit, contactor.
- ◆ Zone C accommodates the control transformer and the noise receivers: control and sensor system.
- ◆ Zone D forms the interface between the signal and control cables to the periphery. A defined noise immunity level is required here.
- ◆ Zone E comprises the three-phase motor and the motor supply cable.
- ◆ The zones should be spatially separated in order to achieve electromagnetic de-coupling.

- ◆ Minimum clearance 20 cm.
- ◆ De-coupling by means of grounded partitions is even better. It is not permissible to route cables which have been assigned to various zones together in the same cable ducts!
- ◆ If necessary, filters should be installed at the interface locations between the zones.
- ◆ Non-shielded signal cables can be used within one zone.
- ◆ All bus cables (e.g. RS 485, RS 232) and signal cables leaving the cabinet must be shielded.

3.4.2 Use of filters and coupling elements

EMC cannot be brought about just by installing filters! Measures such as shielded motor feeder cables and spatial separation are also necessary.

Radio interference suppression filters

Radio interference suppression filters reduce the cable-related noise interference voltage at the mains connection point. In order to maintain the limit values ("First environment" or "Second environment"), a radio interference suppression filter is necessary, irrespective of whether a dv/dt or sinusoidal filter is used at the output of the frequency converter.

dv/dt filters

dv/dt filters are used in the first place to protect the motor winding, by reducing the maximum voltage stressing, and in the second place, the reduced voltage gradient will result in a lower noise current.

Sinusoidal filters

Sinusoidal filters are low-pass filters which generate an almost sinusoidal voltage from the voltage blocks which the converter switches at the output terminals. The voltage gradient and the maximum voltage peaks are limited even more effectively than in the case of dv/dt filters.

Coupling elements

In addition, data line filters and/or coupling elements may be required at the interfaces between the zones. Coupling elements with electrical isolation (e.g. isolating amplifiers) prevent the noise from being propagated from one zone to the next. Isolating amplifiers particularly have to be provided in the case of analog signals.

3.5 Design of drives in conformance with EMC regulations

3.5.1 Basic EMC rules

Rules 1 to 13 are generally applicable. Rules 14 to 20 are particularly important for limiting noise emission.

Rule 1 All of the metal cabinet parts must be connected through the largest possible surface areas (not paint on paint). If required, use serrated washers. The cabinet door must be connected to the cabinet through grounding straps which must be kept as short as possible.

NOTE Grounding installations/machines is essentially a protective measure. However, in the case of drive systems, this also has an influence on the noise emission and noise immunity. A system can either be grounded in a star configuration or each component grounded separately. Preference should be given to the latter grounding system in the case of drive systems, i.e. all parts of the installation to be grounded are connected through their surface or in a mesh pattern.

Rule 2 Signal cables and power cables must be routed separately (to eliminate coupled-in noise). Minimum clearance: 20 cm. Provide partitions between power cables and signal cables. The partitions must be grounded at several points along their length.

Rule 3 Contactors, relays, solenoid valves, electromechanical operating hours counters, etc. in the cabinet must be provided with quenching elements, for example, RC elements, diodes, varistors. These quenching devices must be connected directly at the coil.

Rule 4 Non-shielded cables associated with the same circuit (outgoing and incoming conductor) must be twisted, or the surface between the outgoing and incoming conductors kept as small as possible in order to prevent unnecessary coupling effects.

Rule 5 Eliminate any unnecessary cable lengths to keep coupling capacitances and inductances low.

Rule 6 Connect the reserve cables/conductors to ground at both ends to achieve an additional shielding effect.

Rule 7 In general, it is possible to reduce the noise being coupled-in by routing cables close to grounded cabinet panels. Therefore, wiring should be routed as close as possible to the cabinet housing and the mounting panels and not freely through the cabinet. The same applies for reserve cables/conductors.

Rule 8 Tachometers, encoders or resolvers must be connected through a shielded cable. The shield must be connected to the tachometer, encoder or resolver and at the SIMOVERT MASTERDRIVES through a large surface area. The shield must not be interrupted, e.g. using intermediate terminals. Pre-assembled cables with multiple shields should be used for encoders and resolvers (see Catalog DA65.11).

- Rule 9** The cable shields of digital signal cables must be connected to ground at both ends (transmitter and receiver) through the largest possible surface area. If the equipotential bonding is poor between the shield connections, an additional equipotential bonding conductor with at least 10 mm² must be connected in parallel to the shield, to reduce the shield current. Generally, the shields can be connected to ground (= cabinet housing) in several places. The shields can also be connected to ground at several locations, even outside the cabinet.
- Foil-type shields are not to be favoured. They do not shield as well as braided shields; they are poorer by a factor of at least 5.
- Rule 10** The cable shields of **analog** signal cables can be connected to ground at both ends if the equipotential bonding is good. Good equipotential bonding is achieved if Rule 1 is observed.
- If low-frequency noise occurs on analog cables, for example: speed/measured value fluctuations as a result of equalizing currents (hum), the shields are only connected for analog signals at one end at the SIMOVERT MASTERDRIVES. The other end of the shield should be grounded through a capacitor (e.g. 10 nF/100 V type MKT). However, the shield is still connected at both ends to ground for high frequency as a result of the capacitor.
- Rule 11** If possible, the signal cables should only enter the cabinet at one side.
- Rule 12** If SIMOVERT MASTERDRIVES are operated from an external 24 V power supply, this power supply must not feed several consumers separately installed in various cabinets (hum can be coupled-in!). The optimum solution is for each SIMOVERT MASTERDRIVES to have its own power supply.
- Rule 13** Prevent noise from being coupled-in through the supply.
- SIMOVERT MASTERDRIVES and automation units/control electronics should be connected up to different supply networks. If there is only one common network, the automation units/control electronics have to be de-coupled from the supply using an isolating transformer.
- Rule 14** The use of a radio interference suppression filter is obligatory to maintain limit value class "First environment" or "Second environment", even if sinusoidal filters or dv/dt filters are installed between the motor and SIMOVERT MASTERDRIVES.
- Whether an additional filter has to be installed for further consumers, depends on the control used and the wiring of the remaining cabinet.

Rule 15

A noise suppression filter should always be placed close to the fault source. The filter must be connected to the cabinet housing, mounting panel, etc. through a large surface area. A bare metal mounting panel (e.g. manufactured from stainless steel, galvanized steel) is best, as electrical contact is established through the entire mounting surface. If the mounting panel is painted, the paint has to be removed at the screw mounting points for the frequency converter and the noise suppression filter to ensure good electrical contact.

The incoming and outgoing cables of the radio interference suppression filter have to be spatially separated/isolated.

Rule 16

In order to limit the noise emitted, all variable-speed motors have to be connected-up using shielded cables, with the shields being connected to the respective housings at both ends in a low-inductive manner (through the largest possible surface area). The motor feeder cables also have to be shielded inside the cabinet or at least shielded using grounded partitions. Suitable motor feeder cable e.g. Siemens PROTOFLEX-EMV-CY (4 x 1.5 mm² ... 4 x 120 mm²) with Cu shield. Cables with steel shields are unsuitable.

A suitable PG gland with shield connection can be used at the motor to connect the shield. It should also be ensured that there is a low-impedance connection between the motor terminal box and the motor housing. If required, connect-up using an additional grounding conductor. **Do not use plastic motor terminal boxes!**

Rule 17

A line reactor has to be installed between the radio interference suppression filter and the SIMOVERT MASTERDRIVES.

Rule 18

The line supply cable has to be spatially separated from the motor feeder cables, e.g. by grounded partitions.

Rule 19

The shield between the motor and SIMOVERT MASTERDRIVES must not be interrupted by the installation of components such as output reactors, sinusoidal filters, dv/dt filters, fuses, contactors. The components must be mounted on a mounting panel which simultaneously serves as the shield connection for the incoming and outgoing motor cables. Grounded partitions may be necessary to shield the components.

Rule 20

In order to limit the radio interference (especially for limit value class "First environment"), in addition to the line supply cable, all cables externally connected to the cabinet must be shielded.

Examples of these basic rules:

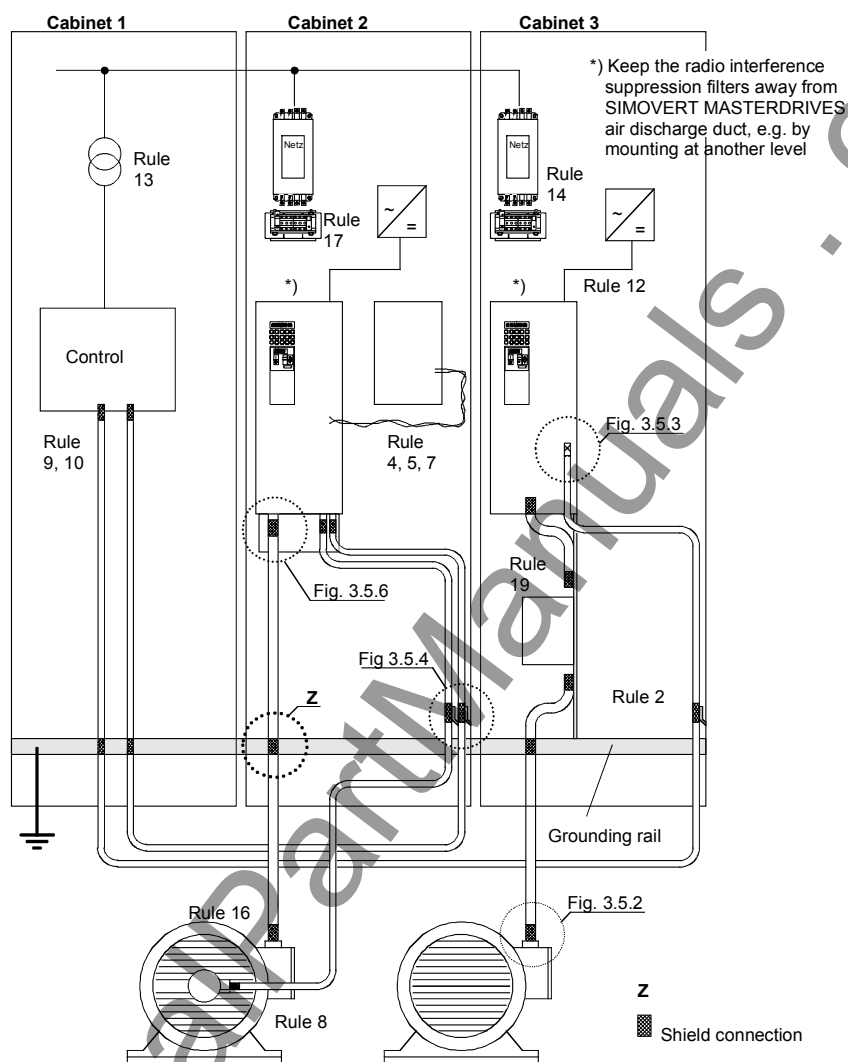


Fig. 3-9 Examples for applying the basic EMC rules

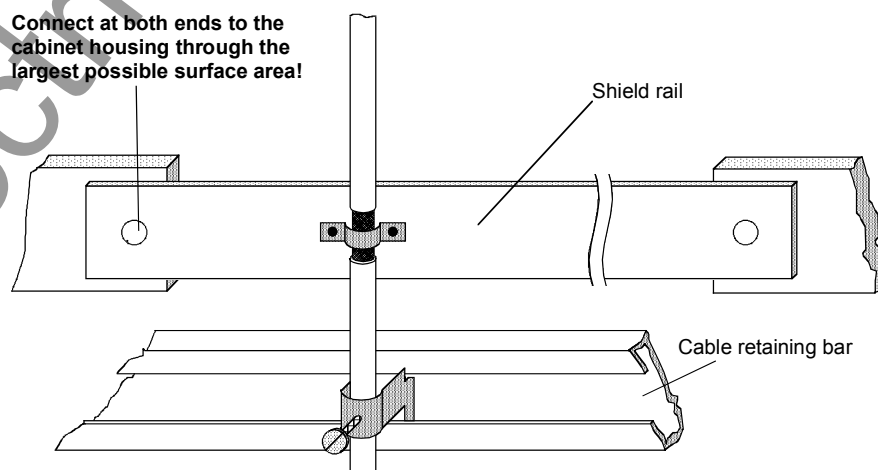


Fig. 3-10 Connecting the motor cable shield where the cable enters the cabinet

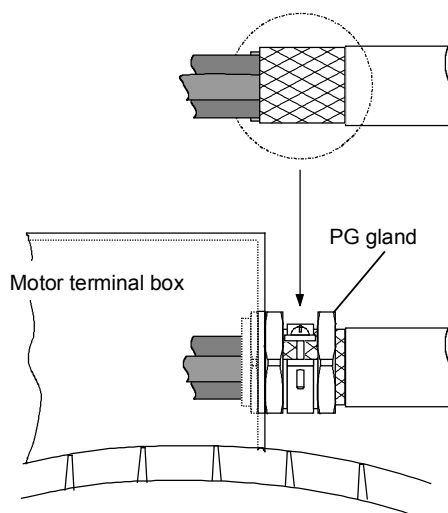


Fig. 3-11 Shield connection at the motor

The shield can be connected through a PG gland (nickel-plated brass) with a strain relief bar. Thus, the degree of protection IP 20 can be achieved.

For higher degrees of protection (up to IP 68), there are special PG glands with shield connection, e.g.:

- ◆ SKINDICHT SHVE, Messrs. Lapp, Stuttgart
- ◆ UNI IRIS Dicht or UNI EMV Dicht, Messrs. Pflitsch, Hückeswagen

It is not permissible to use plastic motor terminal boxes!

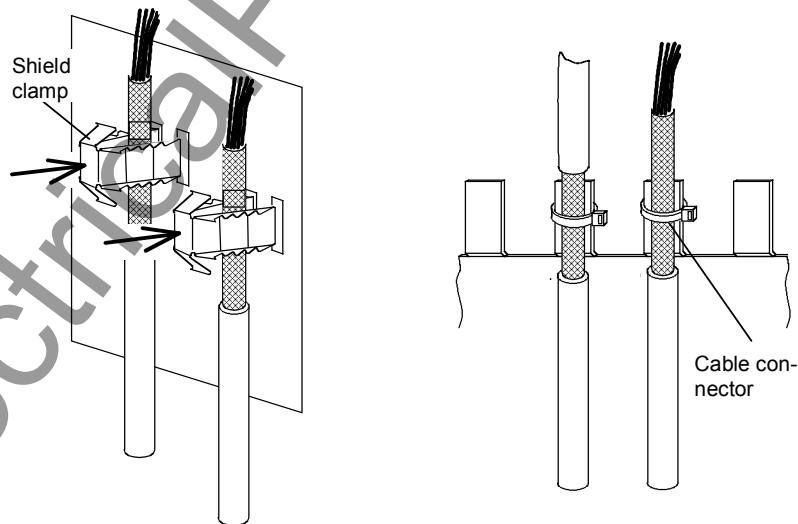


Fig. 3-12 Connecting the signal cable shields for SIMOVERT MASTERDRIVES

- ◆ Every SIMOVERT MASTERDRIVES has shield clamps to connect the signal cable shields.
- ◆ For chassis units (sizes $\geq E$), the shields can be additionally connected using cable connectors at the shield connecting locations.

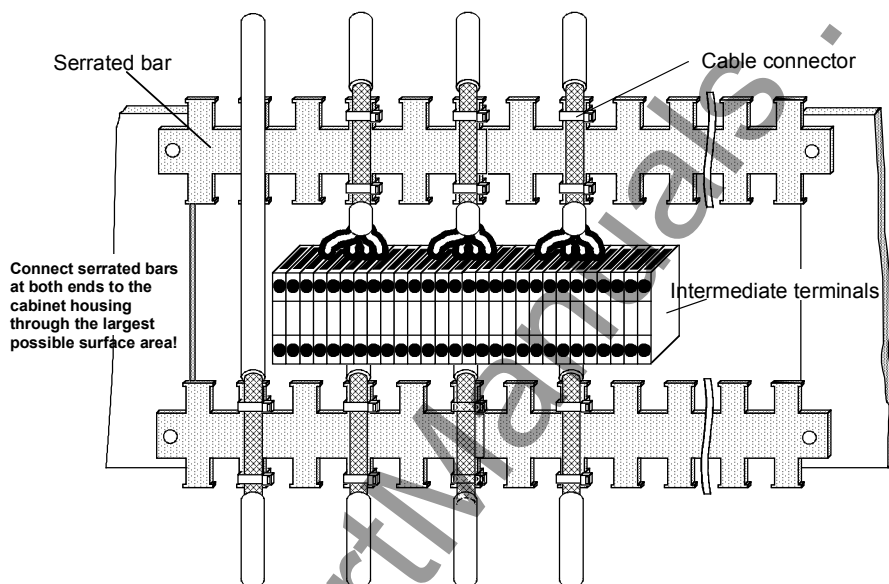


Fig. 3-13 Connecting signal cable shields in the cabinet

Wherever possible, intermediate terminals should not be used as they reduce the shielding effect!

3.5.2 Examples

Compact PLUS type drive

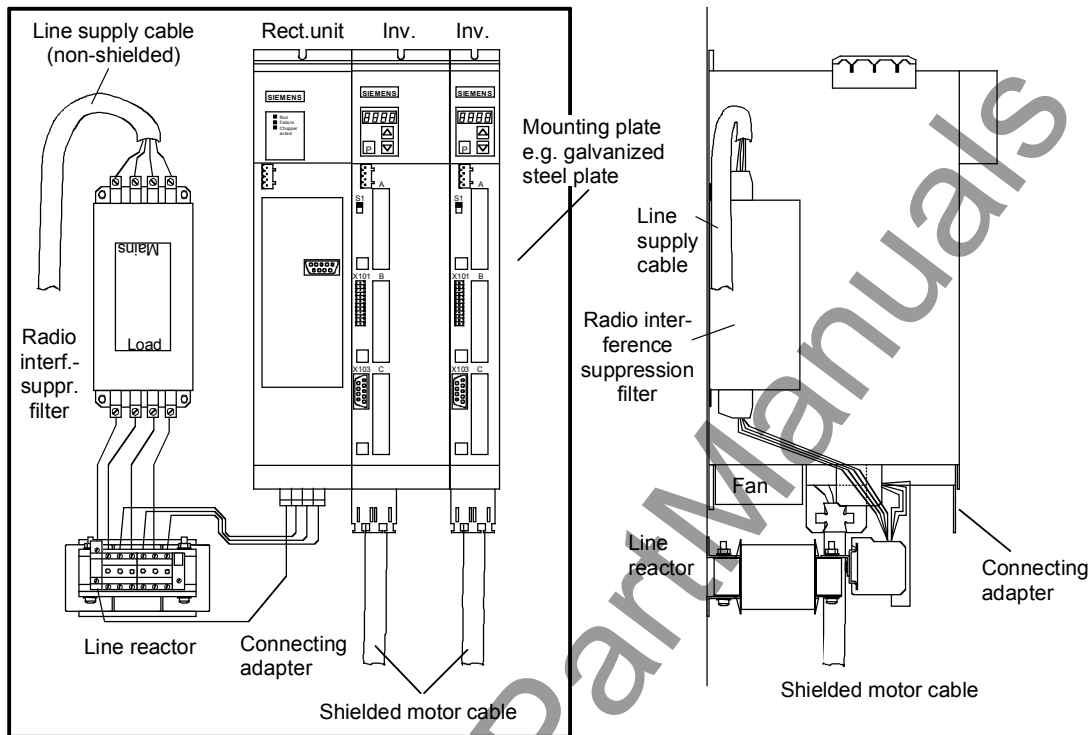


Fig. 3-14 Example of a Compact PLUS type unit with radio interference suppression filter and line reactor

The cabling should be kept as short as possible. The line supply cable to the radio interference suppression filter must be routed separately away from other cables (zone concept!).

The motor must be connected using a shielded cable! The shield has to be connected through the largest possible surface area at the motor and the inverter.

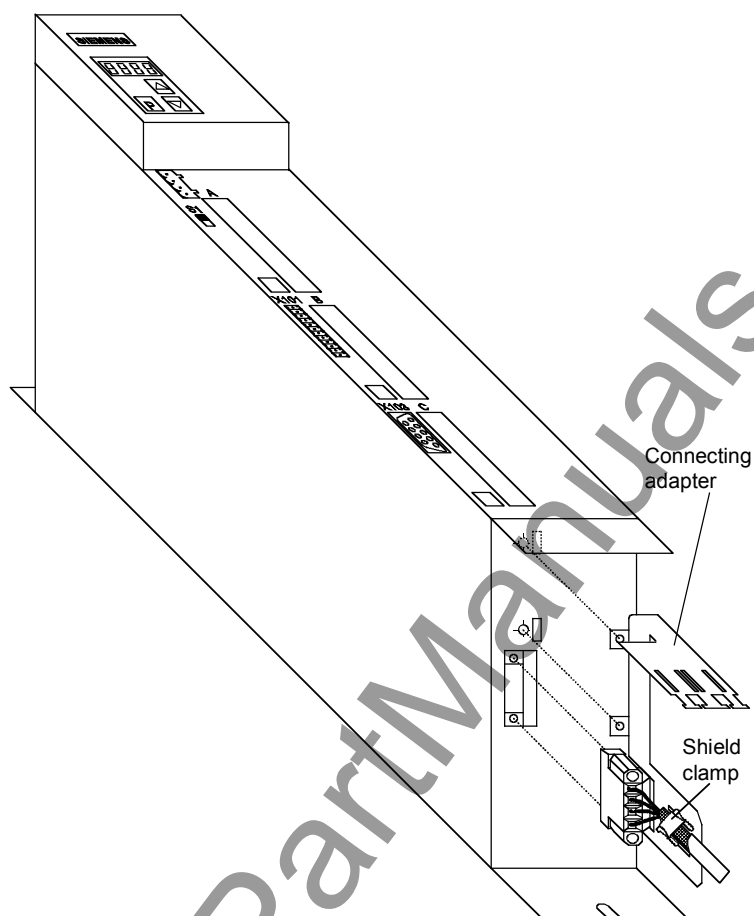


Fig. 3-15 Installing the motor connection and the connecting adapter

The following procedure should be carried out to connect the motor cable and to mount the shield:

- ◆ First connect the motor cable to the removed X2 motor connector
- ◆ Connect the shield of the motor cable through the largest possible surface area at the connecting adapter, e.g. with shield clamps.
- ◆ Insert the fastening straps of the connecting adapter through the slots in the lower housing section and screw them into place.
- ◆ Locate the motor connector X2 and screw tightly to the unit.

The control cables can be attached at the front of the connecting adapter using shield clamps.

Drive unit of Compact type

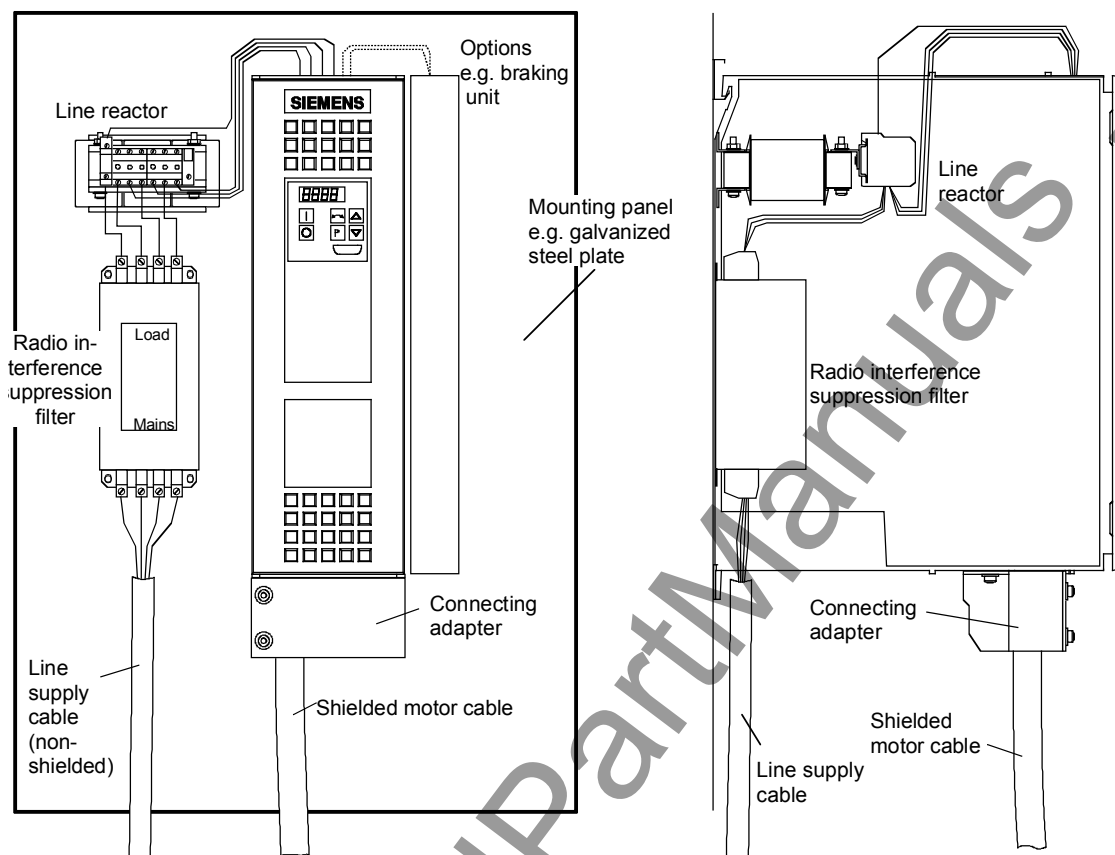


Fig. 3-16 Example of a Compact type unit with radio interference suppression filter and line reactor

The cabling should be kept as short as possible. The line supply cable to the radio interference suppression filter must be routed separately away from other cables (zone concept!).

The motor must be connected using a shielded cable! The shield must be connected through the largest possible surface area at the motor and drive converter. The optional connecting adapter can be used to connect the shield to SIMOVERT MASTERDRIVES.

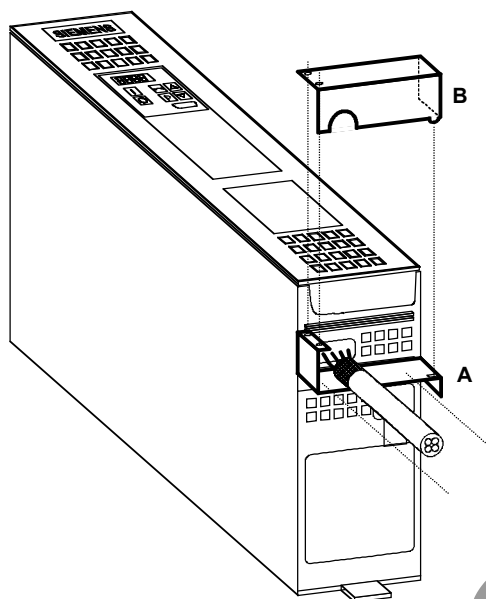


Fig. 3-17 Mounting the connecting adapter

- ◆ Screw lower section A to SIMOVERT MASTERDRIVES.
- ◆ Mount SIMOVERT MASTERDRIVES on the mounting panel.
- ◆ Connect the shielded motor cable and shield to section A through the largest possible surface area, e.g. attach using cable connectors.
- ◆ Locate upper part B and screw into place. The shields of signal cables can be connected to the upper section.

Chassis type drive unit

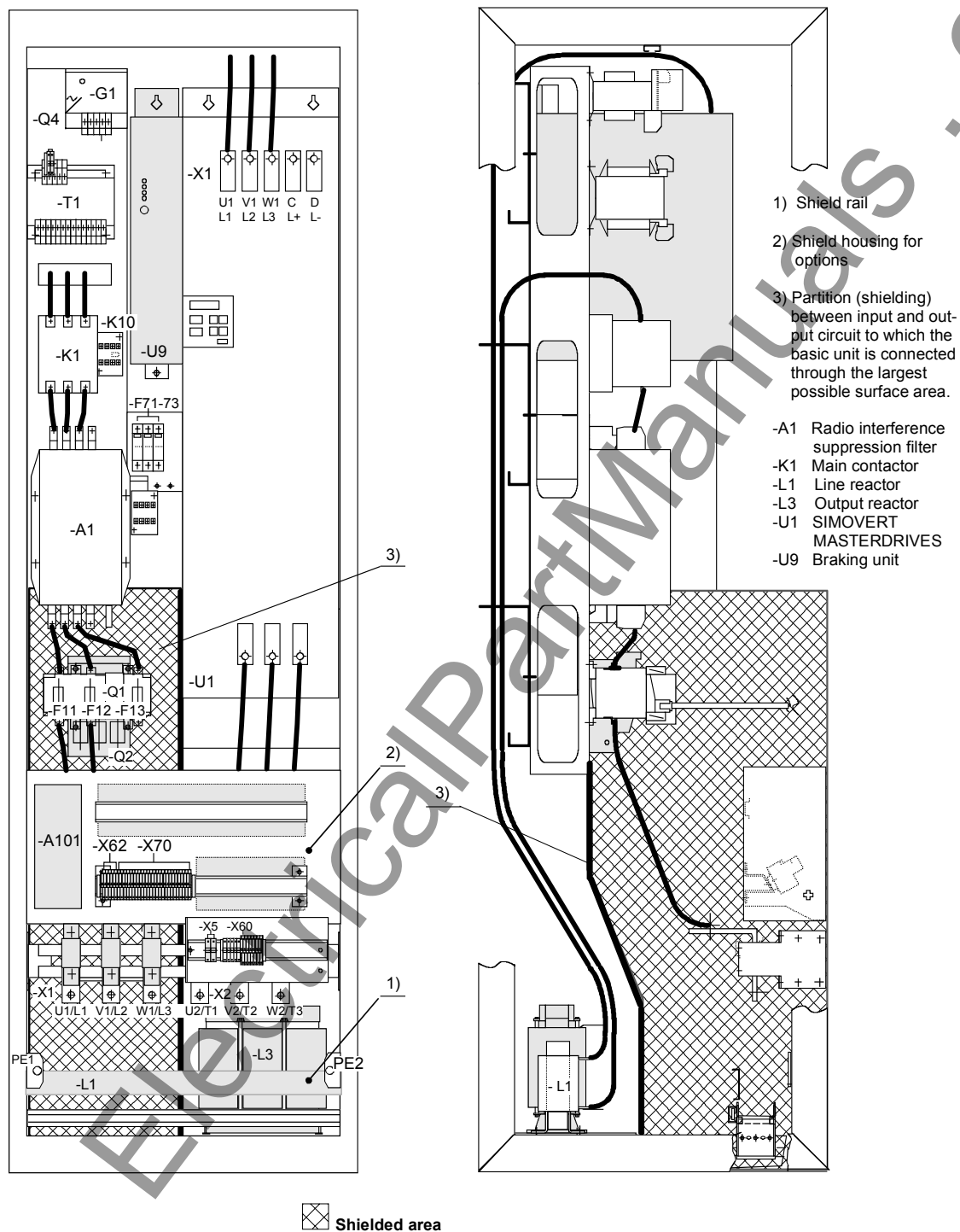


Fig. 3-18

Example of a chassis unit mounted in the cabinet with radio interference suppression filter and line reactor

Example of correct cable routing

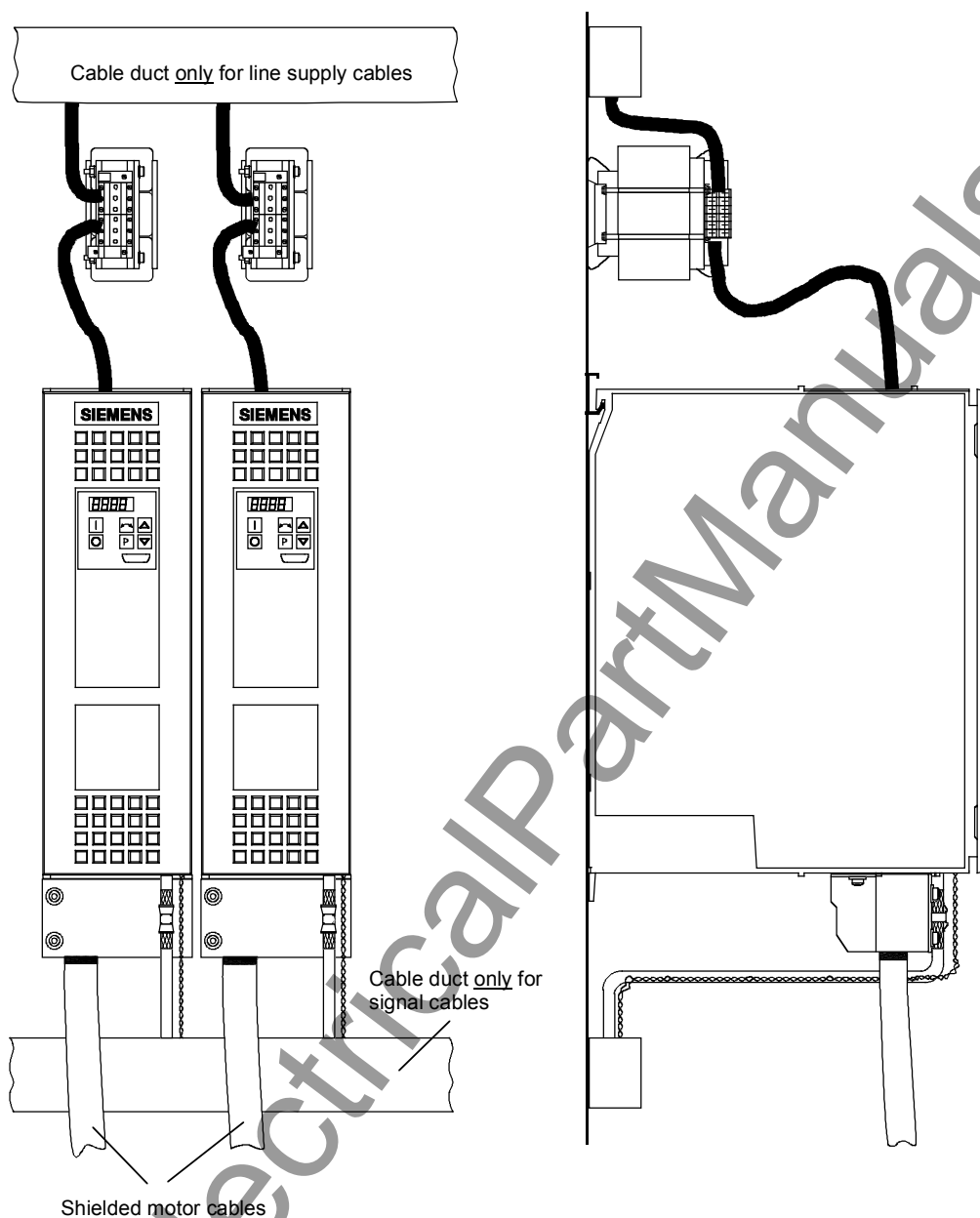


Fig. 3-19 Installation with separate cable ducts

Installation with cable ducts only for the line supply cables. Line supply cables are non-shielded.

The motor and signal cables are routed separately from each other.

The shields of the motor and signal cables have to be mounted on the shield connections through the largest possible surface area.

Example of incorrect cable routing

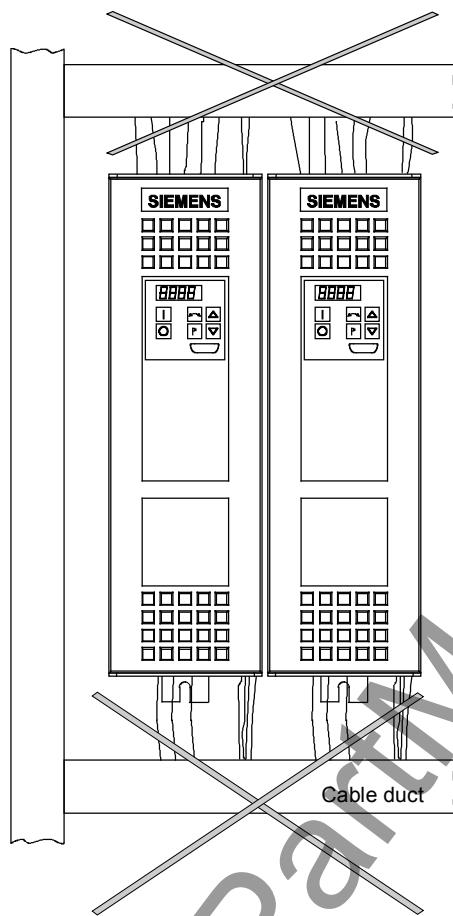


Fig. 3-20 Installation with cable ducts

Installation with cable ducts, mounted on a painted mounting panel. All of the cables are non-shielded.

Optically this layout looks good.

But from an EMC perspective, this installation is useless!

The motor and signal cables are routed in parallel in the lower cable duct. The same is true for line supply cables and external power supplies in the upper cable duct. All of the cables are then routed together in the vertical cable duct.

Cabling such as this allows noise to be easily propagated and coupled-in!

3.6 Assignment of SIMOVERT MASTERDRIVES, radio interference suppression filters and line reactors

The assignment of SIMOVERT MASTERDRIVES, radio interference suppression filters and line reactors is specified in Catalog DA65.11 and the Operating Instructions for the 6SE70 radio interference suppression filters.

The 6SE70 radio interference suppression filters were checked to make sure they maintain the limit values, using layouts consisting of SIMOVERT MASTERDRIVES and the associated line reactors. The components were mounted in cabinets (Type 8MC) in observance of the specified rules. The motor feeder cable was 30 m long.

3.7 Specified standards

EN 55011:	1991	Limits and methods of measurement of radio disturbance characteristics of industrial, scientific and medical (ISM) radio-frequency equipment
EN 50081-1:	1992	Generic emission standard Part 1: Residential, commercial and light industry
EN 50081-2:	1993	Generic emission standard Part 2: Industrial environment
EN 50082-1:	1992	Generic immunity standard Part 1: Residential, commercial and light industry
EN 50082-2:	1995	Generic immunity standard Part 2: Industrial environment
EN 61800-3:	1996	EMC product standard including special test methods for variable-speed electric drive units

4 Function Blocks and Parameters

Control functions

A large number of open-loop and closed-loop control functions, communication functions, as well as diagnostics and operator control functions are implemented in the software of the converters and inverters by means of function blocks. These function blocks can be parameterized and freely interconnected.

The interconnection method can be compared with electrical circuit engineering where various function units, e.g. integrated circuits or other components are interconnected by cables.

The difference is, however, that function blocks are interconnected not by cables, but via software.

4.1 Function blocks

Functions are implemented in function blocks. The function scope of the individual function blocks depends on its special task.

The function blocks are provided with inputs, outputs and parameters and are processed in time slots.

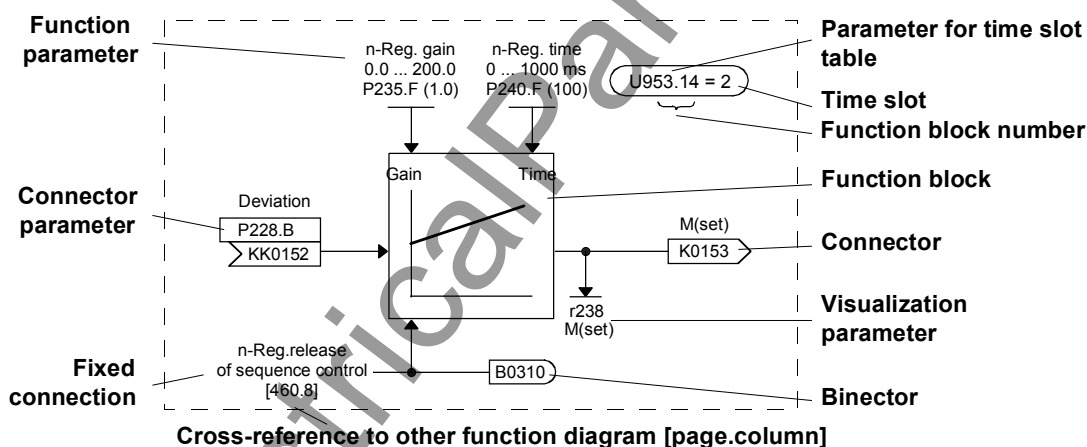


Fig. 4-1

A function block

Function block number

Each function block has a function block number (FB number) by which it can be clearly identified. With the FB number, you can define which time slot can be used for processing a large number of function blocks. For this purpose, each function block is allocated an indexed parameter which contains the relevant FB number in its parameter number and its parameter index.

Example:

U950.01 is the code of FB number 001

U953.50 is the code of FB number 250

U953.99 is the code of FB number 299

U954.74 is the code of FB number 374

The parameter for selecting the time slot as well as the corresponding factory setting are indicated in the function diagrams for each function block. This data takes the form of an ellipse in order to distinguish it optically from the other elements of a function block.

In addition to the time slot, the processing sequence can also be determined for most of the function blocks.

4.2 Connectors and binectors

Connectors and binectors are elements which are used to exchange signals between individual function blocks. They are each cyclically filled by function blocks with one signal value. Other function blocks can then call up these values, depending on parameterization.

Connectors

Connectors can be likened to storage locations which are used to archive "analog" signals. They are clearly designated. Each connector designation comprises the connector name, the connector number and an identification letter.

The identification letter depends on the numerical representation:

- ◆ K Connector with word length (16 bit)
- ◆ KK Connector with double-word length (32 bit, increased accuracy)

The connector number always has four digits.

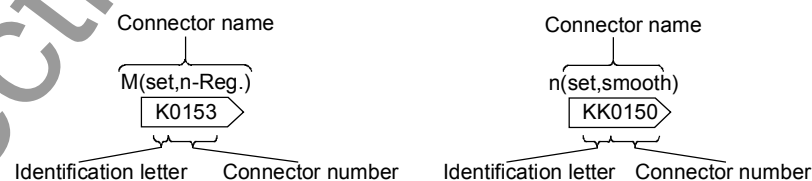


Fig. 4-2 Connectors with word lengths of 16 bit and 32 bit

Value range of the connectors

The values stored in the connectors are normalized values, with a few exceptions (e.g. connectors for control words).

The value range of these connectors covers a percentage value range of:

- ◆ -200 % (8000H / 8000 0000H for double-word connectors) to
- ◆ +199.99 % (7FFFH / 7FFF FFFFH for double-word connectors).

100 % corresponds to the value 4000H (4000 0000H for double-word connectors).

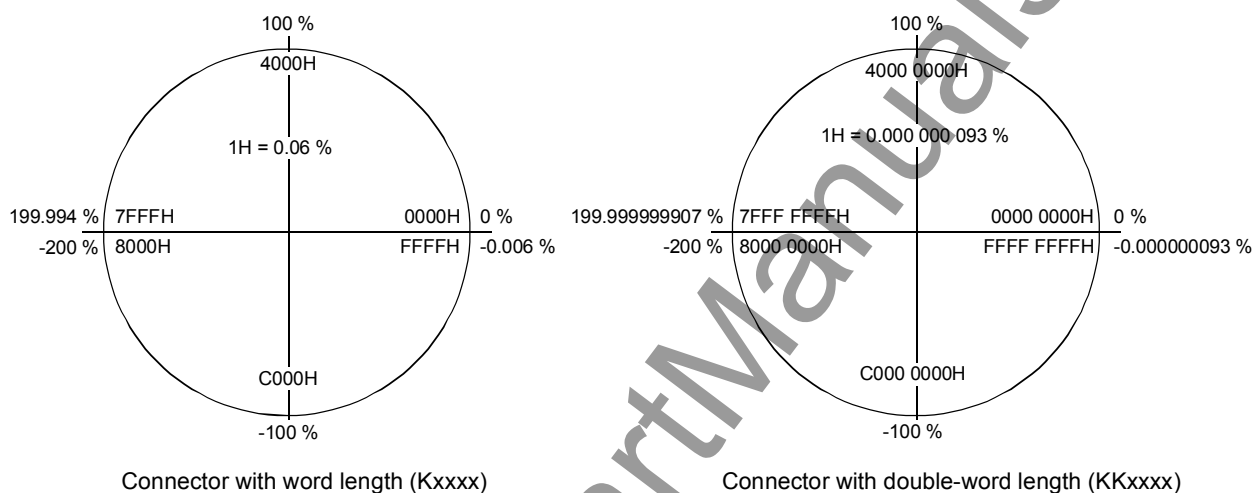


Fig. 4-3 Value range and assignment of the figure ranges for connectors

Binectors

Function blocks archive the **binary** (digital) output information in binary **connectors**, the binectors. Binectors can therefore be likened to storage locations used for storing binary signals. They are clearly identified. Each binector designation comprises the binector name, the binector number and an identification letter. The identification letter is B. The binector number always has four digits.

On account of their definition, binectors can only assume the two states "0" (logically no) and "1" (logically yes).

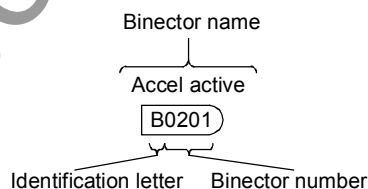


Fig. 4-4 Binectors

4.3 Parameters

Parameters are the intervention points for adapting function blocks to an application, for interconnecting function blocks via connectors and binectors and for visualizing internal signals.

The various parameters are differentiated according to their function as follows:

- ◆ Function parameters (can be read and written)
- ◆ BICO parameters (can be read and written)
- ◆ Visualization parameters (can only be read).

Each parameter is clearly designated. The parameter designation comprises the parameter name and the parameter number, and enables every parameter to be clearly identified. In addition to the parameter name and the parameter number, many parameters also have a parameter index. With the aid of this index, it is possible to store several values for one parameter under one parameter number.

The function diagrams indicate the factory setting for every BICO parameter and every function parameter. They further indicate the value ranges for the changeable function parameters.

Parameter numbers on the PMU

The parameter numbers shown on the parameterizing unit (PMU) which is directly mounted on the unit consist of a letter and a three-digit number.

The following applies for the letters:

- ◆ Upper-case letters (P, U, H and L) represent the BICO parameters and function parameters which can be changed
- ◆ Lower-case letters (r, n, d and c) represent the visualization parameters which cannot be changed.

The three-digit number covers the value range from 000 to 999; but not all values are used.

Parameter numbers on the OP1S

The OP1S operator control panel enables parameters to be selected directly by their parameter numbers. As the OP1S only has a numerical keypad, the parameter number must be replaced by a figure when input. The following replace mode is applicable:

- ◆ "P"xxx and "r"xxx are replaced by "0"xxx
- ◆ "H"xxx and "d"xxx are replaced by "1"xxx
- ◆ "U"xxx and "n"xxx are replaced by "2"xxx
- ◆ "L"xxx and "c"xxx are replaced by "3"xxx

Examples:

Select r004 on OP1S: Input 0004
 Select P050 on OP1S: Input 0050
 Select U123 on OP1S: Input 2123
 Select L411 on OP1S: Input 3411

Function parameters The response of a function block is determined by function parameters. Typical examples of function parameters are:

- ◆ Normalization of an input signal
- ◆ Acceleration or deceleration times in the ramp-function generator
- ◆ Proportional gain (Kp) and integral time (Tn) in the speed controller.

Function parameters can be indexed. The significance of the parameter values stored in the various indices depends on the definition of the respective parameter. A special group is formed by the function parameters which are part of the so-called function data sets.

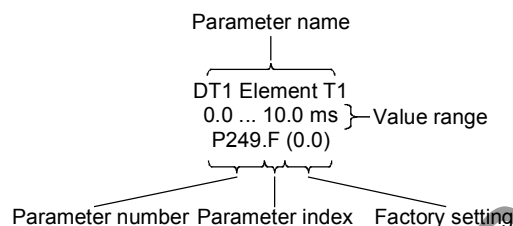


Fig. 4-5 Function parameters

**Function data sets
(Setpoint data sets)**

Special function parameters are put together in function data sets. These parameters are marked in the function diagrams with the parameter index **.F**.

The parameters concerned are indexed four-fold, which means that one parameter value can be stored under each parameter index, i.e. a total of four parameter values can be stored.

The active function data set determines which value is currently being used. If function data set 1 is active, the parameter value stored in parameter index 1 is used. If function data set 2 is active, the parameter value stored in parameter index 2 is used, etc.

Example:

P462.1 = 0.50
P462.2 = 1.00
P462.3 = 3.00
P462.4 = 8.00

A total of 4 values are stored under parameter P462 (Accel Time). If function data set 1 is active, the acceleration time is 0.50 secs. If function data set 2 is active, the acceleration time is 1.00 secs. If function data set 3 is active, the acceleration time is 3.00 secs and if function data set 4 is active, the acceleration time is 8.00 secs.

The individual function data sets are selected by means of control word bits 16 and 17 in control word 2 (P576.B and P577.B). Changeover is possible at any time.

The active function data sets are displayed via the visualization parameter r013 (Active FuncDataSet).

CAUTION

Changeover of all the indexed parameters of the function data sets between parameter indices 1, 2, 3 and 4 is always effected jointly.

BICO parameters

With BICO parameters, you can determine the sources of the input signals of a function block. This means that you can use BICO parameters to define the connectors and binectors from which a function block reads in its input signals. In this manner, you can "soft-wire" the function blocks stored in the units to meet your requirements. This is referred to as the BICO system.

For every BICO parameter, the type of input signals (connector or binector) which you can connect to the inputs is specified. BICO parameters have the following identification:

- ◆ B Binector parameter
for connecting binectors
- ◆ K Connector parameter
for connecting connectors with word length (16 bit)
- ◆ KK Connector parameter
for connecting connectors with double-word length (32 bit)

Reciprocal "softwiring" of binectors and connectors is not permitted. However, you can always connect connector with word length and double-word length to the connector parameters.

BICO parameters are available in two forms; they can either be

- ◆ non-indexed, or
- ◆ double-indexed.

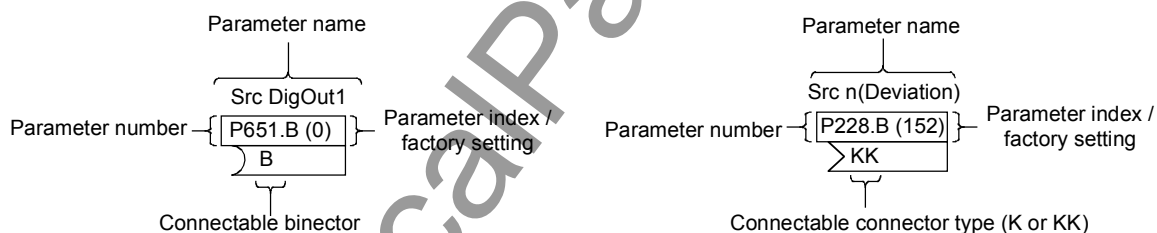


Fig. 4-6 Binector and connector parameters

**BICO data sets
(Basic/reserve data sets)**

Selected BICO parameters are put together in BICO data sets. These parameters are marked in the function diagrams with the parameter index **.B**.

The parameters concerned are double-indexed, which means that one parameter value can be stored under each parameter index of these parameters, i.e. a total of two parameter values can be stored.

The active BICO data set determines which value is currently being used. If BICO data set 1 is active, the parameter value stored in parameter index 1 is used. If BICO data set 2 is active, the parameter value stored in parameter index 2 is used.

Example:

P554.1 = 10

P554.2 = 2100

A total of 2 values are stored under parameter P554 (Src ON/OFF1). If BICO data set 1 is active, the ON command comes from digital input 1 of the basic unit. If BICO data set 2 is active, the ON command comes from bit 0 of the first data word received by serial interface 1.

Individual BICO data sets are selected by means of control word bit 30 in control word 2 (P590).

The active BICO data set is displayed via visualization parameter r012 (Active BICO DS).

CAUTION

All indexed BICO parameters are always switched jointly between parameter index 1 and 2.

Visualization parameters

Visualization parameters are used for visualizing internal quantities (e.g. applicable output current). These parameters are only displayed and cannot be changed by you.

To distinguish them from the other parameters, they are designated with a lower-case letter (r, n, d and c) in the parameter number.

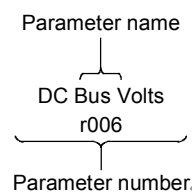


Fig. 4-7 Visualization parameters

4.4 Connecting up function blocks (BICO system)

BICO system is the term used to describe the method of creating connections between function blocks. This is performed with the aid of **binectors** and **connectors**. The name **BICO** system is derived from these two terms.

A connection between two function blocks consists of a connector or binector on the one side, and a BICO parameter on the other side. The connection is always made from the point of view of the input of a function block. You must always assign an output to an input.

Assignment is made by entering in a BICO parameter the number of the connector or the binector from which the required input signals are read in. You are allowed to enter the same connector and binector numbers several times in different BICO parameters and thus use output signals of one function block as input signals for several other function blocks.

Example:

In the following figure, connector K0152 is connected to connector parameter P228. For this purpose, you must assign the number of connector K0152 as the value to the connector parameter P228, i.e. in this case 152.

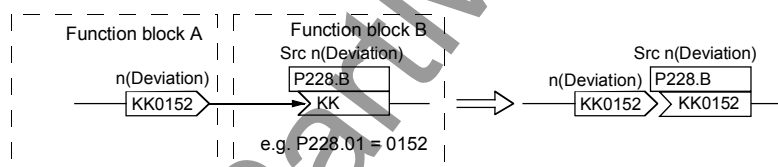


Fig. 4-8 Connecting two function blocks

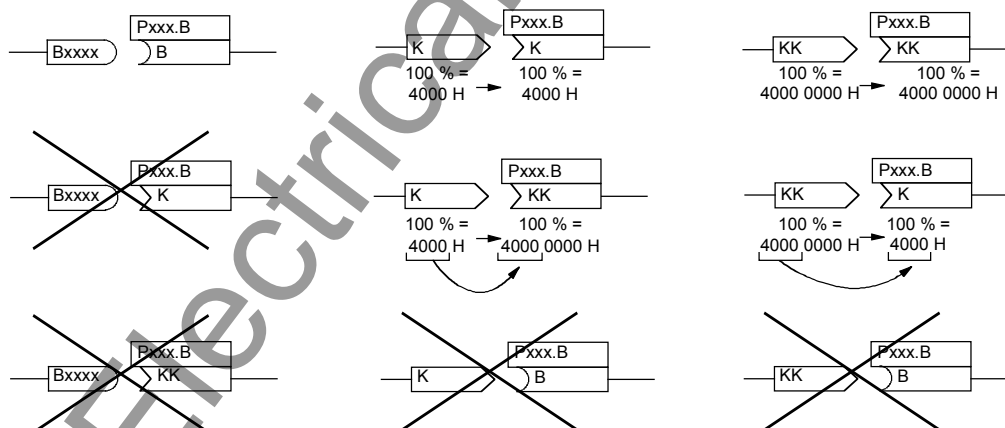


Fig. 4-9 Possible and impossible BICO connections

Interconnecting different connector types

Depending on their characteristics, connectors either have a length of a word (16 bit) or a double-word (32 bit). Accordingly, function blocks have BICO parameters which are suitable for connecting the respective connector type. It is, however, possible in principle to mix the types among the connectors. The word length is then automatically adjusted according to the following mode:

Interconnection of a word connector to	a word connector parameter	Value stays the same
	a double-word connector parameter	Value is taken over in high-word, low-word is filled up with 0000H
Interconnection of a double-word connector to	a word connector parameter	Value is taken over from high-word, low-word deleted
	a double-word connector parameter	Value stays the same

Table 4-1 Interconnecting different connector types

NOTE

When a double-word connector is interconnected to a word connector parameter, the signal resolution will drop from 32 bit to 16 bit. As the low-word is cut off, the information of the lower-order 16 bit of the double-word connectors is then lost.

5 Parameterization

It is possible to parameterize the units of the SIMOVERT MASTERDRIVES series by various methods of parameter input. Every unit can be set via the dedicated parameterizing unit (PMU) without the need to use additional components.

Each unit is supplied with the user software DriveMonitor and comprehensive electronic documentation on a CD. In the case of installation on a standard PC the units can be parameterized via the serial interface of the PC. The software provides extensive parameter aids and a prompted start-up function.

The unit can be further parameterized by entering parameters with the OP1S manual operator panel and via a controller at the field bus level (e.g. Profibus).

NOTE

In firmware V.20 (for performance 2 units) BICO parameters can also be changed in the "Run" drive status (see also parameter list "Changeable in"). In contrast to firmware v1.x in which BICO parameters could only be changed in the "Ready" drive status, structural changes can also be made on performance 2 units with firmware V2.0 during running operation.

WARNING



Unintentional axis movements may occur as a result of undesired changes to BICO parameters in the "Run" drive status.

5.1 Parameter menus

Parameters with related functions are compiled in menus for structuring the parameter set stored in the units. A menu thus represents a selection out of the entire supply of parameters of the unit.

It is possible for one parameter to belong to several menus. The parameter list indicates which individual menus a parameter belongs to. Assignment is effected via the menu number allocated to each menu.

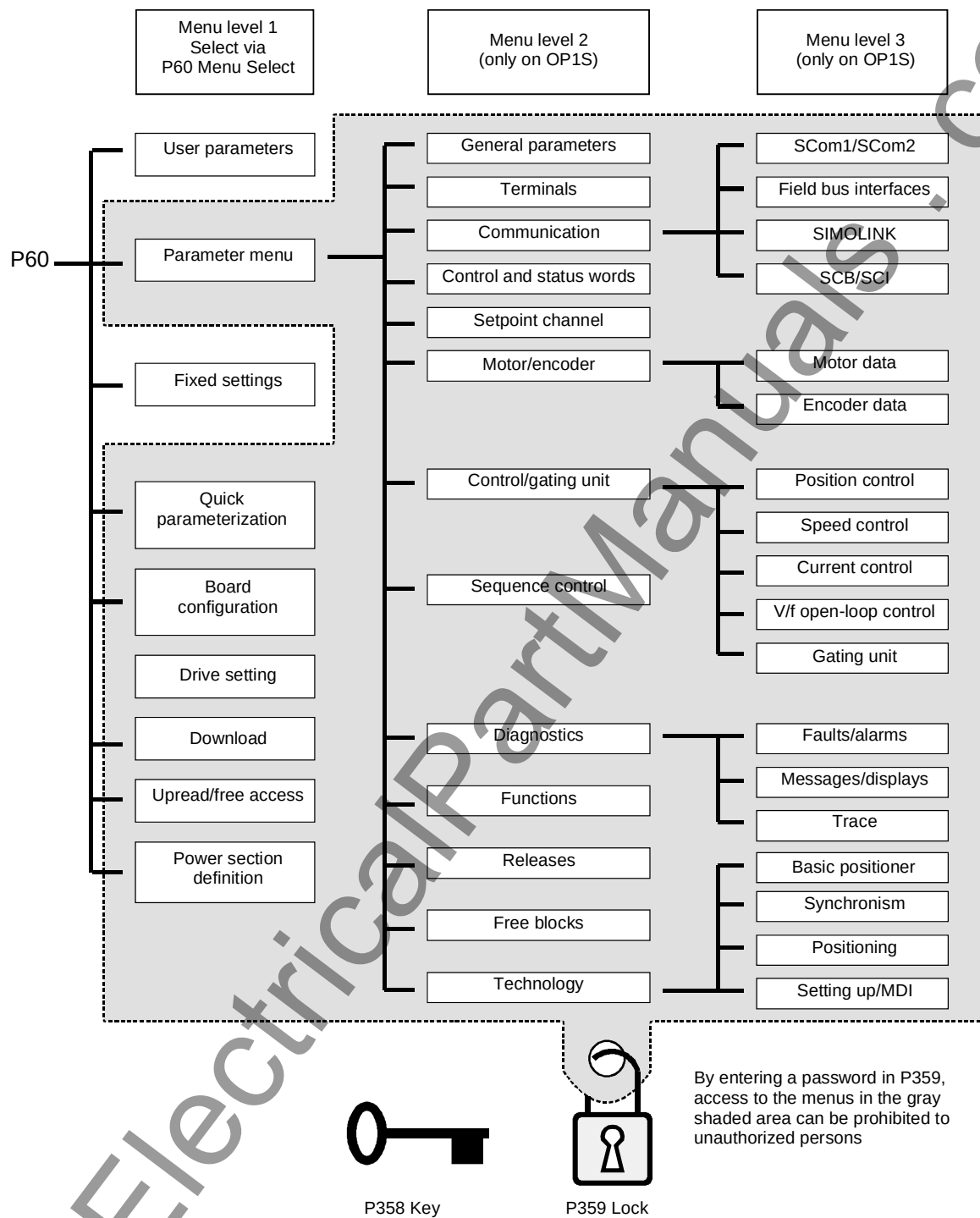


Fig. 5-1

Parameter menus

Menu levels

The parameter menus have several menu levels. The first level contains the main menus. These are effective for all sources of parameter inputs (PMU, OP1S, DriveMonitor, field bus interfaces).

The main menus are selected in parameter P60 Menu Selection.

Examples:

P060 = 0 "User parameters" menu selected

P060 = 1 "Parameter menu" selected

...

P060 = 8 "Power section definition" menu selected

Menu levels 2 and 3 enable the parameter set to be more extensively structured. They are used for parameterizing the units with the OP1S operator control panel.

Main menus

P060	Menu	Description
0	User parameters	Freely configurable menu
1	Parameter menu	- Contains complete parameter set - More extensive structure of the functions achieved by using an OP1S operator control panel
2	Fixed settings	Used to perform a parameter reset to a factory or user setting
3	Quick parameterization	- Used for quick parameterization with parameter modules - When selected, the unit switches to status 5 "Drive setting"
4	Board configuration	- Used for configuring the optional boards - When selected, the unit switches to status 4 "Board configuration"
5	Drive setting	- Used for detailed parameterization of important motor, encoder and control data - When selected, the unit switches to status 5 "Drive setting"
6	Download	- Used to download parameters from an OP1S, a PC or an automation unit - When selected, the unit switches to status 21 "Download"
7	Upread/free access	- Contains the complete parameter set and is used for free access to all parameters without being restricted by further menus - Enables all parameters to be upread/upload by an OP1S, PC or automation unit
8	Power section definition	- Used to define the power section (only necessary for units of the Compact and chassis type) - When selected, the unit switches to status 0 "Power section definition"

Table 5-1 Main menus

User parameters

In principle, parameters are firmly assigned to the menus. However, the "User parameters" menu has a special status. Parameters assigned to this menu are not fixed, but can be changed. You are thus able to put together the parameters required for your application in this menu and structure them according to your needs.

The parameters to be included in the "User parameters" menu are selected in parameter P360 (Select UserParam). This parameter is indexed and permits the input of 100 parameter numbers. The sequence in which the parameter numbers are entered also determines the sequence in which they appear in the "User parameters" menu. If parameters with parameter numbers greater than 999 are to be included in the menu, they have to be input in the usual notation for the OP1S (replacing letters by figures).

Example:

Parameterization of P360	Contained in "User parameters" menu:
P360.1 = 053	P053 Parameter access (always contained)
P360.2 = 060	P060 Menu select (always contained)
P360.3 = 462	P462 Accel Time
P360.4 = 464	P464 Decel Time
P360.5 = 235	P235 n-Reg Gain1
P360.6 = 240	P240 n-Reg Time
P360.7 = 2306	U306 Timer5 Time_s

Table 5-2 Example: Parameterizing a user menu

Lock and key

In order to prevent undesired parameterization of the units and to protect your know-how stored in the parameterization, it is possible to restrict access to the parameters by defining your own passwords with the parameters:

- ◆ P358 key and
- ◆ P359 lock.

If P358 and P359 do not have the same parameterization, only the "User parameters" and the "Fixed settings" menus can be selected in parameter P60 (Menu selection). This means that only the enabled parameters in the "User parameters" menu and the parameters of the "Fixed settings" menu are accessible to the operator. These restrictions are canceled again only if P358 and P359 are given the same parameter setting.

You should proceed in the following manner when using the lock and key mechanism:

1. Adopt key parameter P358 in the "User parameters" menu (P360.x = 358).
2. Program the lock parameter P359 in both parameter indices with your specific password.
3. Change over to the "User parameters" menu.

Depending on the parameterization of the key parameter P358 (the same or not the same as P359), you can now leave the "User parameters" menu and carry out or not carry out further parameterization (Exception: "Fixed settings" menu).

Examples:

Lock	Key	Event
P359.1 = 0 P359.2 = 0 (Factory setting)	P358.1 = 0 P358.2 = 0 (Factory setting)	Lock and key have the same parameter setting, all menus are accessible.
P359.1 = 12345 P359.2 = 54321	P358.1 = 0 P358.2 = 0	Lock and key do not have the same parameter setting, only the "User parameters" and "Fixed settings" menus are accessible.
P359.1 = 12345 P359.2 = 54321	P358.1 = 12345 P358.2 = 54321	Lock and key have the same parameter setting, all menus are accessible.

Table 5-3 Examples of using the lock and key mechanism

5.2 Changeability of parameters

The parameters stored in the units can only be changed under certain conditions. The following preconditions must be satisfied before parameters can be changed:

Preconditions	Remarks
Either a function parameter or a BICO parameter must be involved (identified by upper-case letters in the parameter number).	Visualization parameters (identified by lower-case letters in the parameter number) cannot be changed.
Parameter access must be granted for the source from which the parameters are to be changed.	Release is given in P053 Parameter access.
A menu must be selected in which the parameter to be changed is contained.	The menu assignment is indicated in the parameter list for every parameter.
The unit must be in a status which permits parameters to be changed.	The statuses in which it is possible to change parameters are specified in the parameter list.

Table 5-4 Preconditions for being able to change parameters

NOTE

The current status of the units can be interrogated in parameter r001.

Examples:

Status (r001)	P053	Result
"Ready for ON" (09)	2	P222 Src n(act) can only be changed via the PMU
"Ready for ON" (09)	6	P222 Src n(act) can be changed via the PMU and SCom1 (e.g. OP1S)
"Operation" (14)	6	P222 Src n(act) cannot be changed on account of the drive status

Table 5-5 Influence of drive status (r001) and parameter access (P053) on the changeability of a parameter

5.3 Parameter input via the PMU

The PMU parameterizing unit enables parameterization, operator control and visualization of the converters and inverters directly on the unit itself. It is an integral part of the basic units. It has a four-digit seven-segment display and several keys.

The PMU is used with preference for parameterizing simple applications requiring a small number of set parameters, and for quick parameterization.

PMU in units of the Compact PLUS type

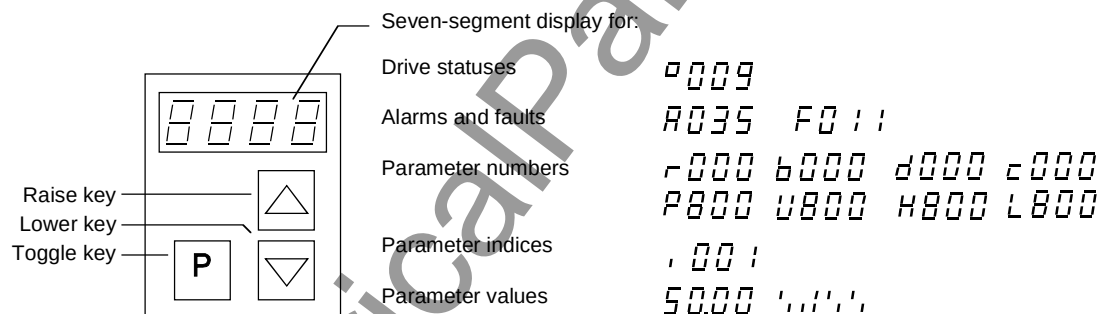


Fig. 5-2 PMU in units of the Compact PLUS type

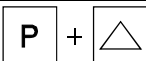
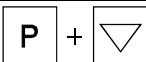
Key	Significance	Function
	Toggle key	- For switching between parameter number, parameter index and parameter value in the indicated sequence (command becomes effective when the key is released) - If fault display is active: For acknowledging the fault
	Raise key	For increasing the displayed value: - Short press = single-step increase - Long press = rapid increase
	Lower key	For lowering the displayed value: - Short press = single-step decrease - Long press = rapid decrease
	Hold toggle key and depress raise key	- If parameter number level is active: For jumping back and forth between the last selected parameter number and the operating display (r000) - If fault display is active: For switching over to parameter number level - If parameter value level is active: For shifting the displayed value one digit to the right if parameter value cannot be displayed with 4 figures (left-hand figure flashes if there are any further invisible figures to the left)
	Hold toggle key and depress lower key	- If parameter number level is active: For jumping directly to operating display (r000) - If parameter value level is active: For shifting the displayed value one digit to the left if the parameter value cannot be displayed with 4 figures (right-hand figure flashes if there are any further invisible figures to the right)

Table 5-6 Operator control elements of the PMU (Compact PLUS type)

PMU in units of the Compact and chassis type

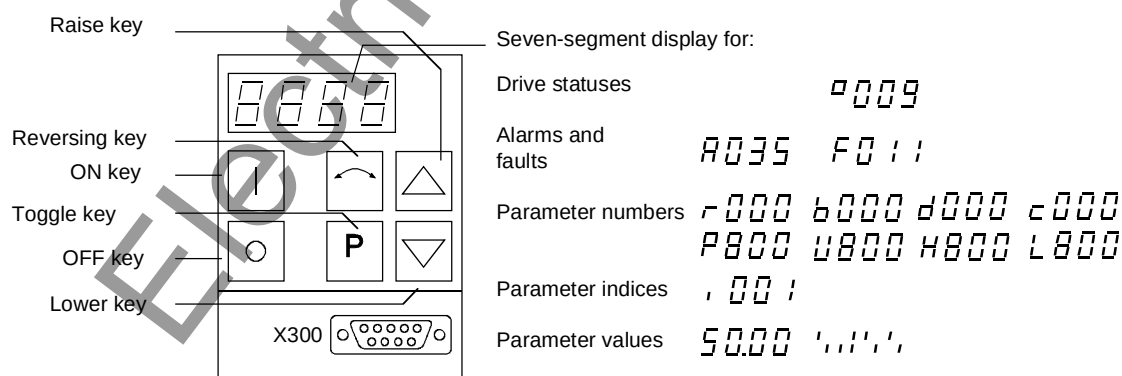


Fig. 5-3 PMU parameterizing unit

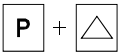
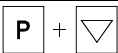
Key	Significance	Function
	ON key	- For energizing the drive (enabling motor activation). - If there is a fault: For returning to fault display
	OFF key	For de-energizing the drive by means of OFF1, OFF2 or OFF3 (P554 to 560) depending on parameterization.
	Reversing key	For reversing the direction of rotation of the drive. The function must be enabled by P571 and P572
	Toggle key	- For switching between parameter number, parameter index and parameter value in the sequence indicated (command becomes effective when the key is released). - If fault display is active: For acknowledging the fault
	Raise key	For increasing the displayed value: - Short press = single-step increase - Long press = rapid increase
	Lower key	For lowering the displayed value: - Short press = single-step decrease - Long press = rapid decrease
	Hold toggle key and depress raise key	- If parameter number level is active: For jumping back and forth between the last selected parameter number and the operating display (r000) - If fault display is active: For switching over to parameter number level - If parameter value level is active: For shifting the displayed value one digit to the right if parameter value cannot be displayed with 4 figures (left-hand figure flashes if there are any further invisible figures to the left)
	Hold toggle key and depress lower key	- If parameter number level is active: For jumping directly to the operating display (r000) - If parameter value level is active: For shifting the displayed value one digit to the left if parameter value cannot be displayed with 4 figures (right-hand figure flashes if there are any further invisible figures to the right)

Table 5-7 Operator control elements on the PMU

**Toggle key
(P key)**

As the PMU only has a four-digit seven-segment display, the 3 descriptive elements of a parameter

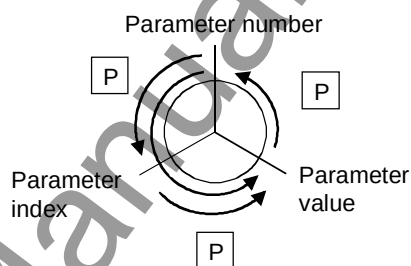
- ◆ Parameter number,
- ◆ Parameter index (if the parameter is indexed) and
- ◆ Parameter value

cannot be displayed at the same time. For this reason, you have to switch between the individual descriptive elements by depressing the toggle key. After the desired level has been selected, adjustment can be made using the raise key or the lower key.

With the toggle key, you can change over:

- from the parameter number to the parameter index
- from the parameter index to the parameter value
- from the parameter value to the parameter number

If the parameter is not indexed, you can jump directly from the parameter number to the parameter value.

**NOTE**

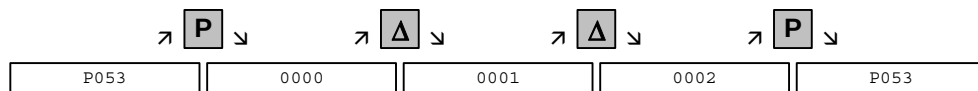
If you change the value of a parameter, this change generally becomes effective immediately. It is only in the case of acknowledgement parameters (marked in the parameter list by an asterisk ' * ') that the change does not become effective until you change over from the parameter value to the parameter number.

Parameter changes made using the PMU are always safely stored in the EEPROM (protected in case of power failure) once the toggle key has been depressed.

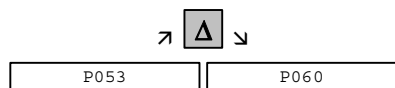
Example

The following example shows the individual operator control steps to be carried out on the PMU for a parameter reset to factory setting.

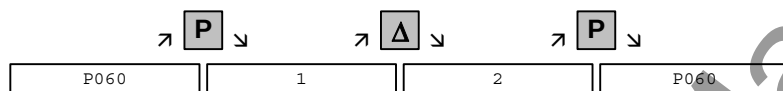
Set P053 to 0002 and grant parameter access via PMU



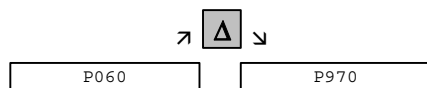
Select P060



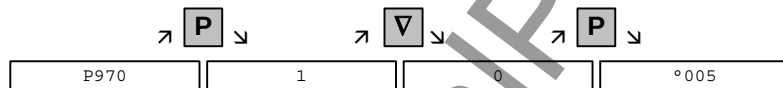
Set P060 to 0002 and select "Fixed settings" menu



Select P970



Set P970 to 0000 and start parameter reset



5.4 Parameter input via the OP1S

5.4.1 General

The operator control panel (OP1S) is an optional input/output device which can be used for parameterizing and starting up the units. Plain-text displays greatly facilitate parameterization.

The OP1S has a non-volatile memory and can permanently store complete sets of parameters. It can therefore be used for archiving sets of parameters. The parameter sets must be read out (upread) from the units first. Stored parameter sets can also be transferred (downloaded) to other units.

The OP1S and the unit to be operated communicate with each other via a serial interface (RS485) using the USS protocol. During communication, the OP1S assumes the function of the master whereas the connected units function as slaves.

The OP1S can be operated at baud rates of 9.6 kBd and 19.2 kBd, and is capable of communicating with up to 32 slaves (addresses 0 to 31). It can therefore be used both in a point-to-point link (e.g. during initial parameterization) and within a bus configuration.

The plain-text displays can be shown in one of five different languages (German, English, Spanish, French, Italian). The language is chosen by selecting the relevant parameter for the slave in question.

Order numbers

Components	Order Number
OP1S	6SE7090-0XX84-2FK0
Connecting cable 3 m	6SX7010-0AB03
Connecting cable 5 m	6SX7010-0AB05
Adapter for installation in cabinet door incl. 5 m cable	6SX7010-0AA00

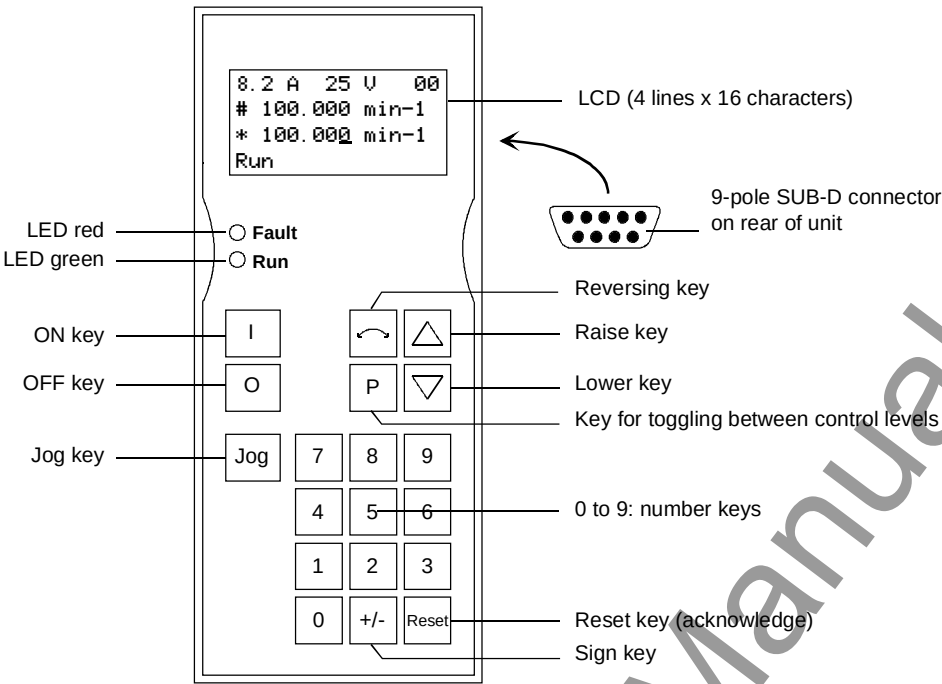
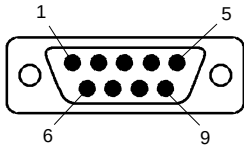


Fig. 5-4 View of the OP1S

OP1S connection



Pin	Designation	Significance	Range
1			
2			
3	RS485 P	Data via RS485 interface	
4			
5	N5V	Ground	
6	P5V	5 V aux. voltage supply	±5%, 200 mA
7			
8	RS485 N	Data via RS485 interface	
9		Reference potential	

Table 5-8 OP1S connections

5.4.2 Connecting, run-up

5.4.2.1 Connecting

The OP1S can be connected to the units in the following ways:

- ◆ Connection via 3 m or 5 m cable (e.g. as a hand-held input device for start-up)
- ◆ Connection via cable and adapter for installation in a cabinet door
- ◆ Plugging into MASTERDRIVES Compact units (for point-to-point linking or bus configuration)
- ◆ Plugging into MASTERDRIVES Compact PLUS units (for bus configuration)

Connection via cable

The cable is plugged into the Sub D socket X103 on units of the Compact PLUS type and into Sub D socket X300 on units of the Compact and chassis type.

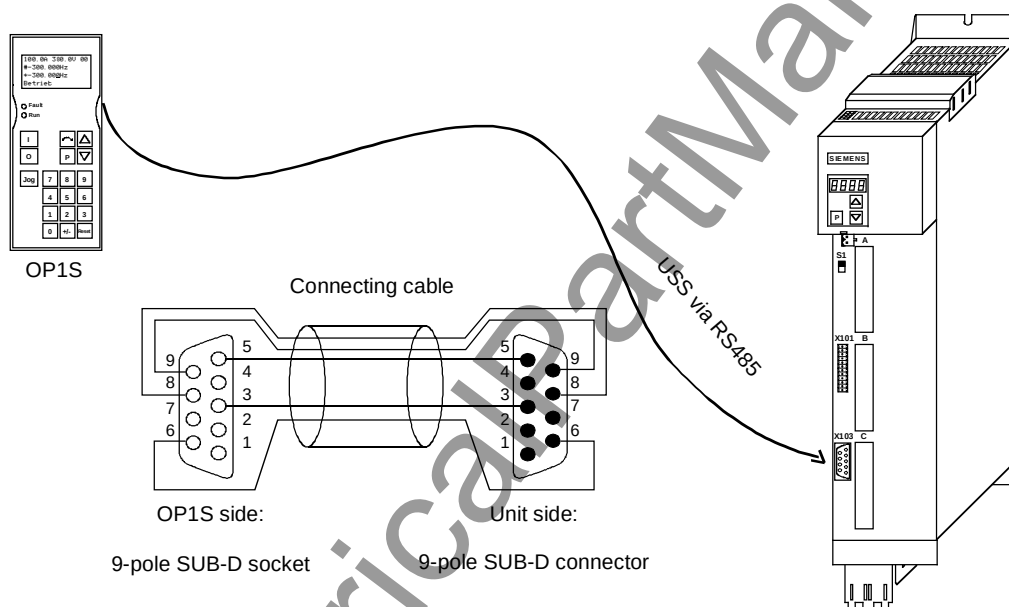


Fig. 5-5 Example: The OP1S in a point-to-point link with the Compact PLUS unit

Plugging into units of the Compact and chassis type

Carefully penetrate the pre-punched holes for the fixing screws in the front panel of the Compact units. Plug the OP1S onto the Sub D socket X300 and screw it tight using the two screws (M5 x 10, accessory pack) from the inside of the front panel.

Plugging onto Compact PLUS rectifier unit

On the Compact PLUS rectifier unit, you can plug the OP1S onto the Sub D socket X320 and lock it in place on the front cover.

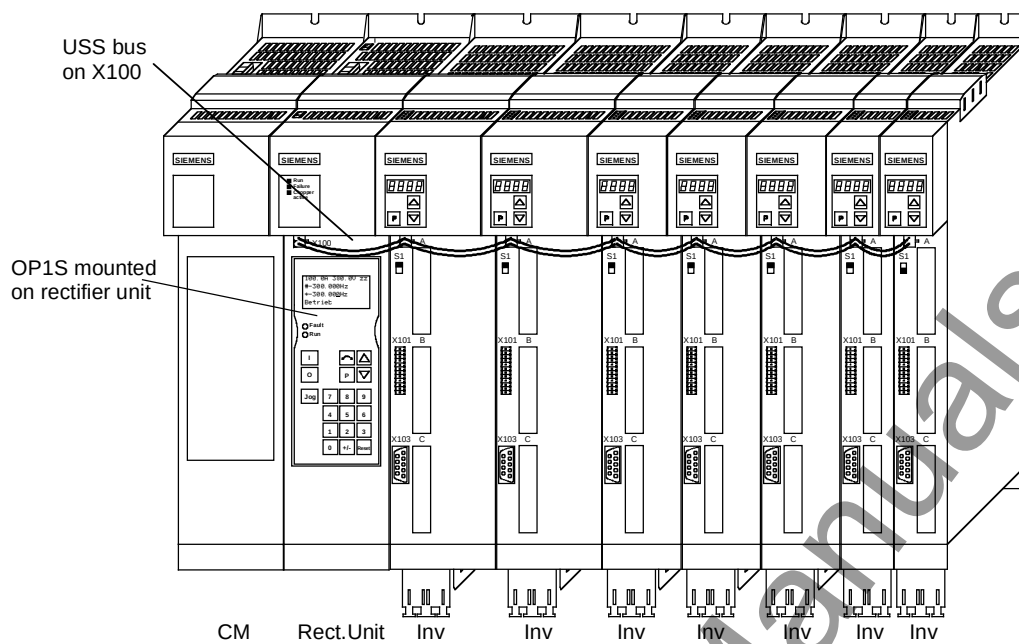


Fig. 5-6 Example: The OP1S during bus configuration with Compact PLUS units

NOTE

During bus operation, the Compact PLUS rectifier unit is only for mechanically restraining the OP1S and for connecting the bus to the inverters. It does not function as a slave.

5.4.2.2 Run-up

After the power supply for the unit connected to the OP1S has been turned on or after the OP1S has been plugged into a unit which is operating, there is a run-up phase.

NOTICE

The OP1S must not be plugged into the Sub D socket if the SCom1 interface parallel to the socket is already being used elsewhere, e.g. bus operation with SIMATIC as the master.

NOTE

In the as-delivered state or after a reset of the parameters to the factory setting with the unit's own control panel, a point-to-point link can be adopted with the OP1S without any further preparatory measures.

When a bus system is started up with the OP1S, the slaves must first be configured individually. The plugs of the bus cable must be removed for this purpose (see section "Bus operation").

During the run-up phase, the text "Search slave" is shown in the first line of the display, followed by "Slave found" and the found slave number as well as the set baud rate.

```
Slave found
Address:  [00]
Baudrate: [6]
```

Example of a display after the run-up phase (6 corresponds to 9.6 kBd)

After approximately 4 s, the display changes to

```
SIEMENS
MASTERDRIVES VC
6SE7016-1EA61
SW:V3.0 OP:V2T20
```

Example of what is displayed after a slave address has been found

After a further 2 s, there is a changeover to the operating display. If it is not possible to start communicating with the slave, an error message "Error: Configuration not ok" appears. About 2 s later, a request is made for new configuration.

```
New config?
#yes
no
```

Error message displayed when communication is not possible

If the "P" key is pressed, the connected unit is reconfigured, i.e. the interface parameters are set to the standard values.

Number of PKWs (P702): 127
 Number of PZDs (P703): 2 or 4
 Telegram failure time (P704): 0 ms

If communication with the slave is still impossible, the reasons may be as follows:

- ◆ Defective cabling
- ◆ Bus operation with two or more slaves with the same bus address (see section "Bus operation")
- ◆ The baud rate set in the slave is neither 9.6 nor 19.2 kBd

In the latter case, an error message "Error: No slave found" appears. The unit's own PMU control panel must then be used to set parameter P701 (baud rate) to 6 (9.6 kBd) or 7 (19.2 kBd) or to reset the parameters to the factory setting.

5.4.3 Operator control

5.4.3.1 Operator control elements

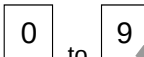
Key	Significance	Function
	ON key	For energizing the drive (enabling motor activation). The function must be enabled by P554.
	OFF key	For de-energizing the drive by means of OFF1, OFF2 or OFF3. The function must be enabled by P554 to P560.
	Jog key	For jogging with jog setpoint 1 (only effective when the unit is in the "Ready to start" state). This function must be enabled by P568.
	Reversing key	For reversing the direction of rotation of the drive. This function must be enabled by P571 and P572.
	Toggle key	- For selecting menu levels and switching between parameter number, parameter index and parameter value in the sequence indicated. The current level is displayed by the position of the cursor on the LCD display (the command comes into effect when the key is released). - For conducting a numerical input.
	Reset key	- For leaving menu levels - If fault display is active: For acknowledging the fault. This function must be enabled by P565
	Raise key	For increasing the displayed value: - Short press = single-step increase - Long press = rapid increase - If motorized potentiometer is active, this is for raising the setpoint. This function must be enabled by P573
	Lower key	For lowering the displayed value: - Short press = single-step decrease - Long press = rapid decrease - If motorized potentiometer is active, this is for lowering the setpoint. This function must be enabled by P574
	Sign key	For changing the sign so that negative values can be entered
	Number keys	Numerical input

Table 5-9 Operator control elements of the OP1S

5.4.3.2 Operating display

After run-up of the OP1S, the following operating display appears:

```

0.0A 0V 00
# 0.00 min-1
* 0.00 min-1
Ready.

```

Example of an operating display in the "Ready" status

The values shown in the operating display (except for slave number, 1st line on the far right) can be specified by means of parameterization:

1st line, left (P0049.001) in the example "Output current"

1st line, right (P0049.002) in the example "DC link voltage"

2nd line actual value (P0049.003) in the example "Actual speed"
(only a visualization parameter)

3rd line setpoint (P0049.004) in the example "Speed setpoint"

4th line (P0049.005) in the example "Operating state"

In the operating display, the actual value is indicated with "#" and the setpoint with "*".

In addition to the operating display on the display unit, the operating state is indicated by the red and green LEDs as follows:

	Flashing	Continuous
red LED	Alarm	Fault
green LED	Ready for ON	Operation

Table 5-10 Operating displays

5.4.3.3 Basic menu

When the "P" key is pressed, a changeover is made from the operating display to the basic menu.

↗ **P** ↘

0.0 A 0 V 00	MotionControl
# 0.00 min-1	*Menu Selection
* 0.00 min-1	OP: Upread
Ready.	OP: Download

Display of the basic menu

The basic menu is the same for all units. The following selections can be made:

- ◆ Menu selection
- ◆ OP: Upread
- ◆ OP: Download
- ◆ Delete data
- ◆ Change slave
- ◆ Config. slave
- ◆ Slave ID

As not all the lines can be shown at the same time, it is possible to scroll the display as required with the "Lower" and "Raise keys."

↗ ▾ ↘ ↗ ▾ ↘ ↗ ▾ ↘ ↗ ▾ ↘ and so on

MotionControl	MotionControl	MotionControl	MotionControl	MotionControl
*Menu Selection	*Menu Selection	*Menu Selection	OP: Upread	OP: Download
OP: Upread	#OP: Upread	OP: Upread	OP: Download	Delete data
OP: Download	OP: Download	#OP: Download	#Delete data	#Change slave

Example of switching from one line to the next

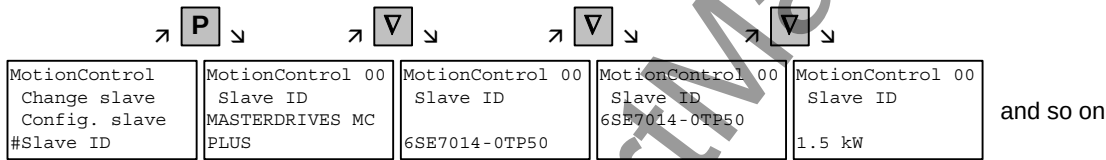
The currently active function is indicated by the "*" symbol and the selected function by the "#" symbol. After the "P" key has been pressed, the relevant symbol jumps to the selected function. The "Reset" key is for returning to the operating display.

5.4.3.4 Slave ID

With the "Slave ID" function, the user can request information about the connected slave. The slave ID consists, for example, of the following lines:

MASTERDRIVES MC
PLUS
6SE7014-0TP50
1.5 kW
V1.0
15.09.1997

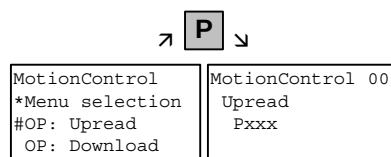
Starting from the basic menu, the "Slave ID" function is selected with "Raise" or "Lower" and activated with "P". As all the lines cannot be shown at the same time, it is possible to scroll the display as required with the "Lower" and "Raise" keys. In addition, the slave number is shown at the top on the right-hand side.



Example of a slave ID

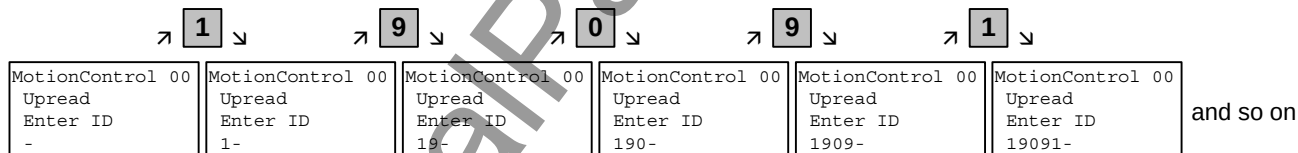
5.4.3.5 OP: Upread

With the "OP: Upread" function, the parameters of the connected slave can be upread and stored in the flash memory inside the OP1S. Starting from the basic menu, the "OP: Upread" function is selected with "Lower" or "Raise" and started with "P". If the available memory is insufficient, the procedure is interrupted with an appropriate error message. During upread, the OP1S indicates the parameter currently being read. In addition, the slave number is shown at the top on the right-hand side.



Example: Selecting and starting the "Upread" procedure

With "Reset", the procedure can be interrupted at any time. If the upread procedure has been completed in full, the user is requested to enter an ID with a maximum of 12 characters for the stored parameter set. This identification can, for example, consist of the date and two differentiating numbers. It is entered with the numerical keypad. With "Lower" a number which has been entered can be deleted.

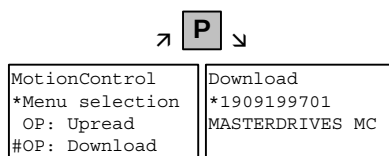


Example of entering an ID

When "P" is pressed, the message "Upread ok" appears and the display changes to the basic menu.

5.4.3.6 OP: Download

With the "OP: Download" function, a parameter set stored in the OP1S can be written into the connected slave. Starting from the basic menu, the "OP: Download" function is selected with "Lower" or "Raise" and activated with "P".



Example: Selecting and activating the "Download" function

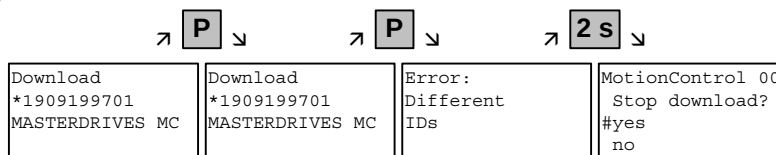
One of the parameter sets stored in the OP1S must now be selected with "Lower" or "Raise" (displayed in the second line). The selected ID is confirmed with "P". The slave ID can now be displayed with "Lower" or "Raise" (see section "Slave ID"). The "Download" procedure is then started with "P". During download, the OP1S displays the currently written parameter.



Example: Confirming the ID and starting the "Download" procedure

With "Reset", the procedure can be stopped at any time. If downloading has been fully completed, the message "Download ok" appears and the display returns to the basic menu.

After the data set to be downloaded has been selected, if the identification of the stored software version does not agree with the software version of the unit, an error message appears for approximately 2 seconds. The operator is then asked whether downloading is to be discontinued.

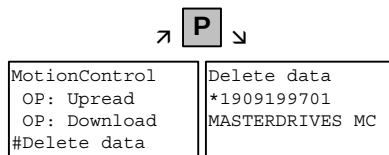


Yes: The "Download" procedure is discontinued.

No: The "Download" procedure is carried out.

5.4.3.7 Delete data

With the "Delete data" function, the user can delete parameter sets stored in the OP1S, thus, for example, creating space for new parameter sets. Starting from the basic menu, the "Delete data" function is selected with "Lower" or "Raise" and activated with "P".



Example: Selection and activation of the "Delete data" function

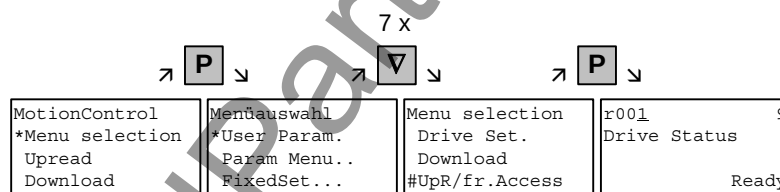
One of the parameter sets stored in the OP1S must now be selected with "Lower" or "Raise" (displayed in the second line). With "P", the selected ID is confirmed. The slave ID can now be displayed with "Lower" or "Raise" (see section "Slave ID"). The "Delete data" procedure can now be started with "P". After completion, the message "Data deleted" appears and the display returns to the basic menu.

5.4.3.8 Menu selection

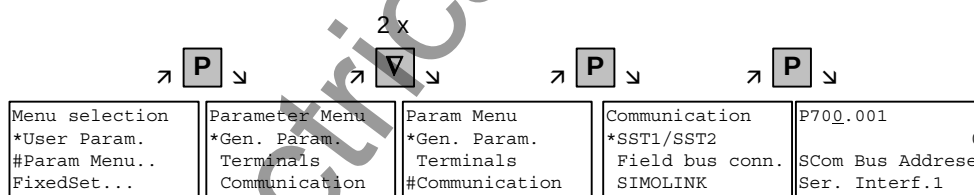
The actual parameterization and start-up of the connected slave is performed by means of the "Menu selection" function. Starting from the basic menu, the "Menu selection" function is selected with "Lower" or "Raise". By pressing "P", the unit-specific sub-menu is displayed with the following choices:

- ◆ User Param.
- ◆ Param Menu..
- ◆ FixedSet...
- ◆ Quick Param...
- ◆ Board Conf.
- ◆ Drive Set
- ◆ Download
- ◆ UpR/fr.Access
- ◆ Power Def.

Two or more dots after these items mean that there is a further sub-menu level. If "Parameter menu.." is selected, access is possible to all parameters via correspondingly structured sub-menus. If "UpR/fr. Access" is selected, direct access is gained to the parameter level.



Example: Selecting the parameter level by means of UpR/fr.access



Example: Selecting a parameter via sub-menus

Parameter display and parameter correction

A parameter number can be selected from the parameter level directly with the numerical keys or with "Raise"/"Lower". The parameter number is shown as a three-figure quantity. In the event of four-figure parameter numbers, the first figure (1, 2 or 3) is not displayed. A distinction is made with the letters (P, H, U etc.).

↗ 0 ↘	↗ 4 ↘	↗ 9 ↘
r001 Drive Status Ready	r000	r004
9		
		r049.001 OP OperDisp 1 st line, on left 4

Example: Direct input of the parameter number with the numerical keypad

↗ Δ ↘	↗ Δ ↘	↗ Δ ↘
r001 Drive Status Ready.	r002 Actual speed 0 min-1	r004 Output Amps 0.0 A
9		
		r006 DC Bus Volts 0 V

Example: Correcting the parameter number by means of "Raise"

If the parameter is found not to exist when the number is entered, a message "No PNU" appears. A non-existent parameter number can be skipped by selecting "Raise" or "Lower".

How the parameters are shown on the display depends on the type of parameter. There are, for example, parameters with and without an index, with and without an index text and with and without a selection text.

Example: Parameter with index and index text

P704.001	0 ms
SCom Tlg OFF	
Ser Interf.1	

- 1st line: Parameter number, parameter index
- 2nd line: Parameter value with unit
- 3rd line: Parameter name
- 4th line: Index text

Example: Parameter with index, index text and selection text

P701.001	6
SCom Baud rate	
Ser Interf.1	
9600 Baud	

- 1st line: Parameter number, parameter index, parameter value
- 2nd line: Parameter name
- 3rd line: Index text
- 4th line: Selection text

Example: Parameter without index, with selection text, binary value

P053	0006Hex
Parameter Access	
0000000000000110	
ComBoard: No	

1st line: Parameter number, parameter value, hexadecimal parameter value

2nd line: Parameter name

3rd line: Parameter value, binary

4th line: Selection text

Transition between the parameter number, parameter index and parameter value levels is made with "P".

Parameter number → "P" → Parameter index → "P" → Parameter value

If there is no parameter index, this level is skipped. The parameter index and the parameter value can be corrected directly with the "Raise"/"Lower" keys. An exception to this are parameter values shown in binary form. In this case, the individual bits are selected with "Raise"/"Lower" and corrected with the numerical keys (0 or 1).

If the index number is entered by means of the numerical keys, the value is not accepted until "P" is pressed. If the "Raise" or "Lower" keys are used to correct the number, the value comes into effect immediately. The acceptance of an entered parameter value and return to the parameter number does not take place until "P" is pressed. The level selected in each case (parameter number, parameter index, parameter value) is marked with the cursor. If an incorrect parameter value is entered, the old value can be obtained by pressing "Reset". The "Reset" key can also be used to go one level lower.

Parameter value → "Reset" → Parameter index → "Reset" → Para.No.

Parameters which can be changed are shown in upper-case letters and visualization parameters which cannot be changed are shown in lower-case letters. If a parameter can only be changed under special conditions or if an incorrect value has been entered with the numerical keys, an appropriate message follows, e.g.:

- ◆ "Value not perm." Incorrect value entered
- ◆ "Value <> min/max" Value too large or too small
- ◆ "P53/P92?" No parameter access
- ◆ "Operating status?" Value can only be changed in the "Drive setting" status, for example

With "Reset", the message is deleted and the old value is re-instated.

NOTE

Parameter changes are always stored with power-failure protection in the EEPROM of the unit connected to the OP1S.

Example of parameter correction:

Selection of parameter value		Correction of parameter value		Accept and return	
↗ P ↘		↗ Δ ↘		↗ P ↘	
P605	0	P605	0	P605	1
Brake control		Brake control		Brake control	
without brake		without brake		Brake w/o chkbk	
				Brake w/o chkbk	

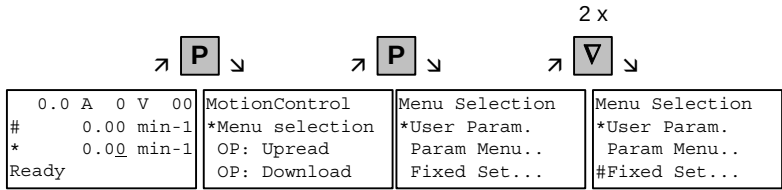
Selection of parameter value		Correction of parameter value		Accept and return	
↗ P ↘		↗ 5 ↘		↗ P ↘	
P600	0 ms	P600	0 ms	P600	5 ms
MCont mssg time		MCont mssg time		MCont mssg time	

Selection of parameter index		Correction of parameter index		Selection of parameter value		Correction of parameter value		Accept and return	
↗ P ↘		↗ Δ ↘		↗ P ↘		↗ 4 ↘		↗ P ↘	
P049.001	4	P049.001	4	P049.002	6	P049.002	6	P049.002	4
OP OperDisp		OP OperDisp		OP OperDisp		OP OperDisp		OP OperDisp	
1st line, left		1st line, left		1st line, right		1st line, right		1st line, ???	
								1st line, ???	

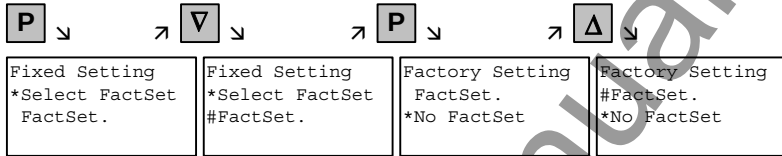
Selection of parameter index		Selection of bit		Correction of bit		Accept and return	
↗ P ↘		↗ Δ ↘		↗ 0 ↘		↗ P ↘	
P053	0006Hex	P053	0006Hex	P053	0006Hex	P053	0004Hex
Parameter Access		Parameter Access		Parameter Access		Parameter Access	
0000000000000110		0000000000000110		0000000000000110		0000000000000110	
ComBoard: No		ComBoard: No		BaseKeypad: Yes		BaseKeyp: No	
						BaseKeyp: No	

Some parameters may also be displayed without a parameter number, e.g. during quick parameterization or if "Fixed setting" is selected. In this case, parameterization is carried out via various sub-menus.

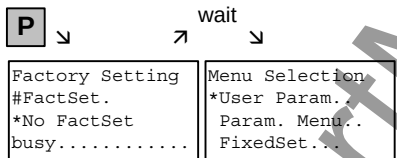
Example of how to proceed for a parameter reset.



Selection of fixed setting



Selection of factory setting



Start of factory setting

NOTE

It is not possible to start the parameter reset in the "Run" status.

Fault and alarm messages

A fault or alarm message is indicated by the red LED. In the event of a fault, the red LED lights up and stays on. A fault message appears in the 3rd and 4th line of the operating display.



0.0 A 0 V 00	0.0 A 0 V 00
# 0.00 min-1	# 0.00 min-1
F065: SCom Tlg	1T 3h 2"
Fault 1/1	Fault 1/1

Example of a fault display

The fault number and the respective text are shown in the 3rd line. Up to 8 fault messages can be stored but only the first fault to occur is shown on the display. Several subsequent faults are shown in the 4th line, e.g. with 1/3 (first of three). Information on all faults can be obtained from the fault memory. With "Raise"/"Lower", the associated operating hours are shown when a fault is waiting to be remedied.

After the cause of a fault has been removed, the fault is acknowledged with "Reset" inside the operating display (the "Reset" key must be appropriately parameterized. See section "Issuing commands via the OP1S"). By pressing "P" and "Lower" at the same time, it is possible to skip back directly to the operating display from the parameter level.

When there is an alarm, the red LED flashes. A warning appears in the 4th line of the operating display.

8.2 A 520 V 00	
# 100.00 min-1	
* 100.00 min-1	
-33:Overspeed	

Example of an alarm display

The alarm number and the respective text is shown in the 4th line. There can be several alarms at the same time but only the first alarm to occur is shown on the display. Several alarms are shown in the 4th line before the alarm number with an "+" instead of "-". Information on all alarms can be obtained with the alarm parameters r953 to r969.

An alarm cannot be acknowledged. As soon as the cause no longer exists, the alarm/display disappears automatically.

5.4.3.9 Issuing commands via the OP1S

Control functions and setpoint specifications for the connected unit can be selected with the corresponding keys of the OP1S, for example during start-up. To do so, the sources of the control commands have to be added to the corresponding bits of word 1 of the SCom1 interface. For setpoint specification, the sources of the setpoints must be appropriately "interconnected". In addition, the setpoint to be changed is to be parameterized as a displayed value in the 3rd line of the operating display.

Key	Function	Parameter number	Parameter value
 	ON/OFF1	P554 Source ON/OFF1	2100
 	Motorized potentiometer: setpoint higher, lower (only effective within the operating display)	P573 Source Raise MOP P574 Source Lower MOP P443 Source Main Setpoint P049.004 Setpoint Operating Disp	2113 2114 KK0058 (MOP Output) 424 (MOP Out)
 to   or 	Setpoint specification by means of fixed setpoint (only effective within the operating display. If entered with numerical key, confirm with "P")	P443 Source Main Setpoint P573 Source Raise MOP P574 Source Lower MOP P049.004 Setpoint Operating Disp	KK0040 (Fixed setpoints) 0 0 e.g. 401 (selected fixed setpoint)
	Reversing	P571 Source clockwise direc. of rotation P572 Source anti-clockwise direc. of rotation	2111 2112
	Acknowledging (only effective within the operating display)	P565 Source Acknowledge	2107
	Jogging with jog setpoint 1 (only effective in the "Ready" status)	P568 Source Jog Bit 0 P448 Jog Setpoint 1	2108 Setpoint in %

NOTE

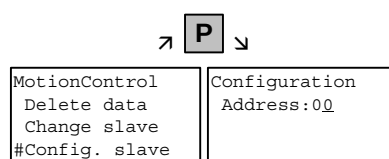
The OFF function can also be performed with OFF2 or OFF3 instead of OFF1. For this, the source of OFF2 (P555) or OFF3 (P556) must be "interconnected" to 2101 or 2102 respectively in addition to setting P554.

5.4.4 Bus operation

In order to start operating a bus system with the OP1S, the slaves must first be configured individually. To do this, the bus connecting cable between the slaves must be interrupted (pull out the bus-cable plug). For configuration, the OP1S is connected with each slave one after the other. A precondition for carrying out the configuration is a baud rate of 9.6 or 19.2 kBd set in the slave (see section "Run-up").

5.4.4.1 Configuring slaves

Starting from the basic menu, the "Config. slave" function is selected with "Lower"/"Raise" and activated with "P". The user is now requested to enter a slave address.



Example of activating the "Config. slave" function

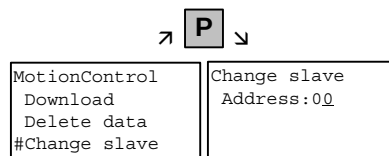
After a different slave address for each slave has been entered by means of the "Raise" key or with the numerical keypad and confirmed with "P", configuration is carried out, i.e. the interface parameters are set to the standard value (see section "Run-up"). In addition, the slave address is entered and a baud rate of 9.6 kBd is set in the slave. After configuration has been completed, the message "Configuration ok" appears, followed by a return to the basic menu. If the configuration of all slaves has been successfully completed, bus operation can be started after the bus connection between the slaves has been restored.

NOTE

During bus operation, each slave must have a different address (P700). Bus operation is also possible at 19.6 kBd (set P701 to 7). The baud rate, however, must be set the same in all slaves.

5.4.4.2 Changing slaves

During bus operation, a specific slave can be selected via the OP1S with the "Change slave" function without any re-plugging. Starting from the basic menu, the "Change slave" function is selected with the "Lower"/"Raise" key and activated with "P". The user is then requested to enter a slave address.



Example of activating the "Change slave" function

After the slave address has been entered with "Raise"/"Lower" and confirmed with "P", a change is made to the required slave and the display returns to the basic menu. If the slave cannot be found, an error message is output.

5.4.5 Technical data

Order number	6SE7090-0XX84-2FK0
Supply voltage	5 V DC $\pm$ 5 %, 200 mA
Operating temperature	0 °C to +55 °C
Storage temperature	-25 °C to +70 °C
Transport temperature	-25 °C to +70 °C
Environment class	Acc. to DIN IEC 721 Part 3-3/04.90
• Humidity	03K3
• Pollution resistance	13C3
Protection class	II acc. DIN VDE 0160 Part 1/05.82 IEC 536/1976
Degree of protection	Acc. to DIN VDE 0470 Part 1/11.92
• Front	IP54 EN60529
• Rear	IP21
Dimensions W x H x D	74 x 174 x 26 mm
Standards	VDE 0160/E04.91 VDE 0558 Part 1/07.87 UL, CSA

Table 5-11 Technical data

5.5 Parameter input with DriveMonitor

NOTE

Please refer to the online help for detailed information on DriveMonitor ( button or F1 key).

5.5.1 Installation and connection

5.5.1.1 Installation

A CD is included with the devices of the MASTERDRIVES Series when they are delivered. The operating tool supplied on the CD (DriveMonitor) is automatically installed from this CD. If "automatic notification on change" is activated for the CD drive on the PC, user guidance starts when you insert the CD and takes you through installation of DriveMonitor. If this is not the case, start file "Autoplay.exe" in the root directory of the CD.

5.5.1.2 Connection

There are two ways of connecting a PC to a device of the SIMOVERT MASTERDRIVES Series via the USS interface. The devices of the SIMOVERT MASTERDRIVES Series have both an RS232 and an RS485 interface.

RS232 interface

The serial interface that PCs are equipped with by default functions as an RS232 interface. This interface is not suitable for bus operation and is therefore only intended for operation of a SIMOVERT MASTERDRIVES device.

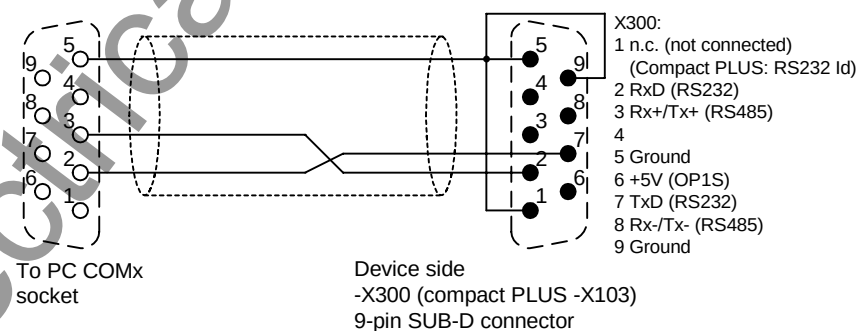


Fig. 5-7 Connecting cable for connecting PC COM(1-4) to SIMOVERT MASTERDRIVES X300

NOTICE

DriveMonitor must not be operated via the Sub-D socket X300 if the SST1 interface parallel to it is already being used for another purpose, e.g. bus operation with SIMATIC as the master.

RS485 interface

The RS485 interface is multi-point capable and therefore suitable for bus operation. You can use it to connect 31 SIMOVERT MASTERDRIVES with a PC. On the PC, either an integrated RS485 interface or an RS232 ↔ RS485 interface converter is necessary. On the device, an RS485 interface is integrated into the -X300 (compact PLUS -X103) connection. For the cable: see pin assignment -X300 and device documentation of the interface converter.

5.5.2 Establishing the connection between DriveMonitor and the device**5.5.2.1 Setting the USS interface**

You can configure the interface with menu *Tools* → *ONLINE Settings*.

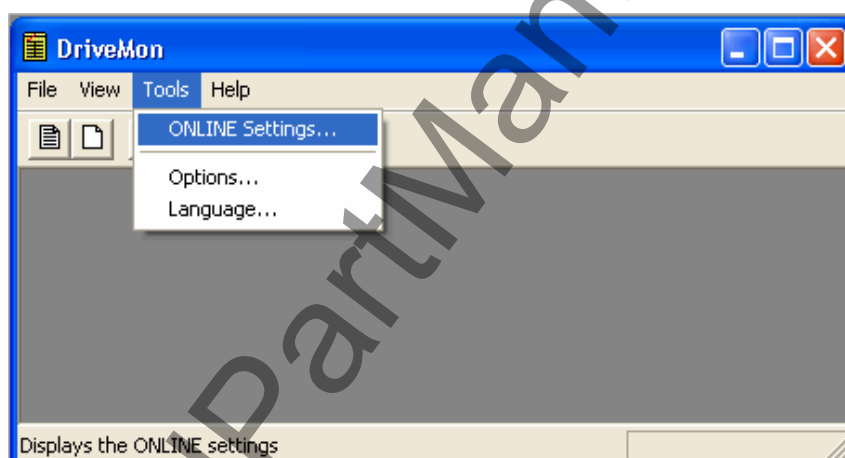


Fig. 5-8 Online settings

The following settings (Fig. 5-9) are possible:

- ◆ **Tab card "Bus Type"**, options
 USS (operation via serial interface)
 Profibus DP (only if DriveMonitor is operated under Drive ES).
- ◆ **Tab card "Interface"**
 You can enter the required COM interface of the PC (COM1 to COM4) and the required baudrate here.

NOTE

Set the baudrate to the baudrate parameterized in SIMOVERT MASTERDRIVES (P701) (factory setting 9600 baud).

Further settings: operating mode of the bus in RS485 operation; setting according to the description of the interface converter RS232/RS485

- ◆ **Tab card "Extended"**
 Request retries and Response timeout; here you can increase the values already set if communication errors occur frequently.

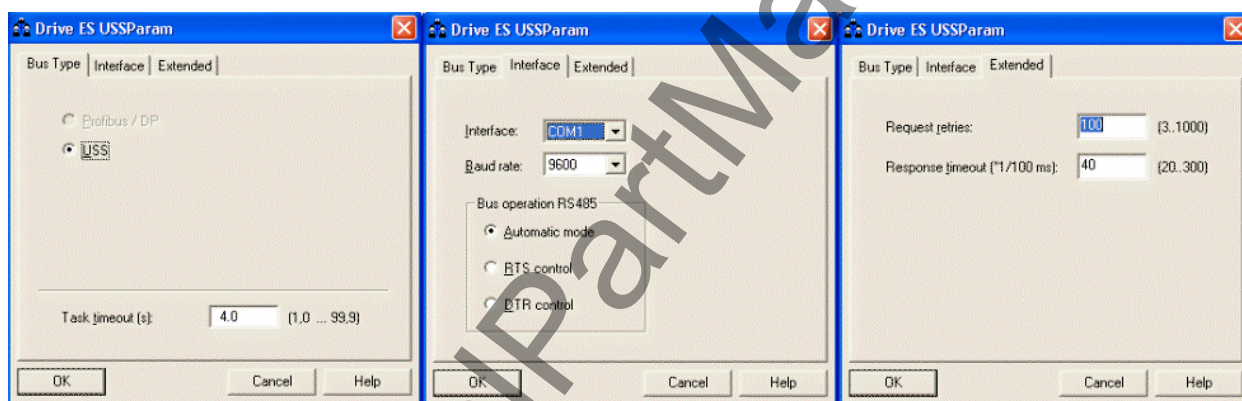


Fig. 5-9 Interface configuration

5.5.2.2 Starting the USS bus scan

DriveMonitor starts with an empty drive window. Via the menu "Set up an ONLINE connection..." the USS bus can be scanned for connected devices:

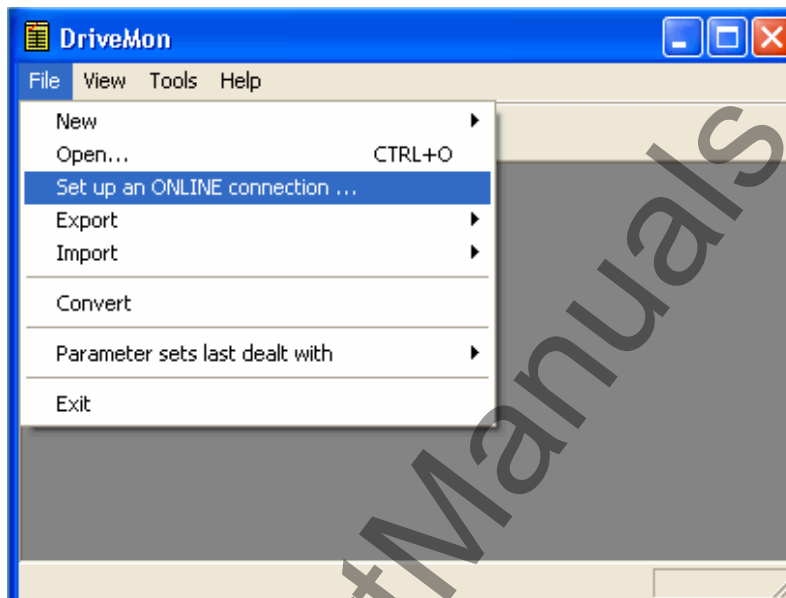


Fig. 5-10 Starting the USS bus scan

NOTE

The "Set up an online connection" menu is only valid from Version 5.2 onwards.

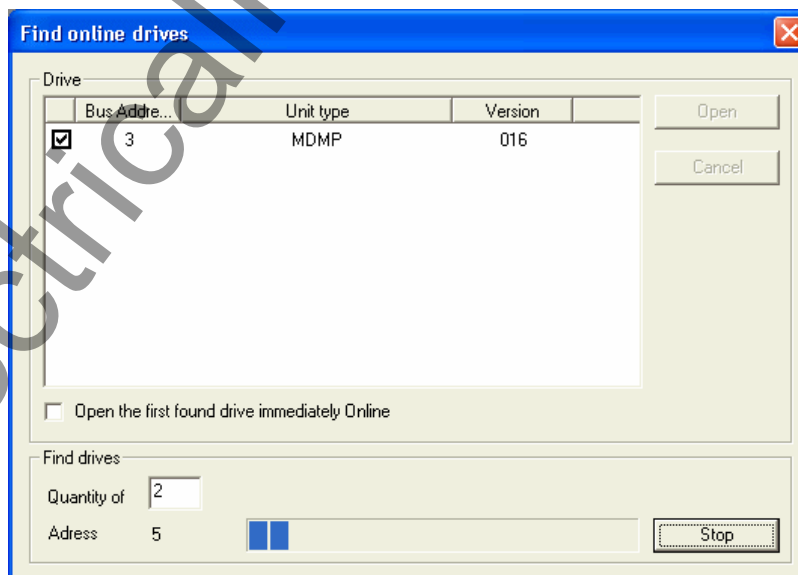


Fig. 5-11 Search for online drives

During the search the USS bus is scanned **with the set baudrate only**. The baud rate can be changed via "Tools → ONLINE Settings", see section 5.5.2.1.

5.5.2.3 Creating a parameter set

With menu *File* → *New* → ... you can create a new drive for parameterization (see Fig. 5-12). The system creates a download file (*.dnl), in which the drive characteristic data (type, device version) are stored. You can create the download file on the basis of an empty parameter set or the factory setting.

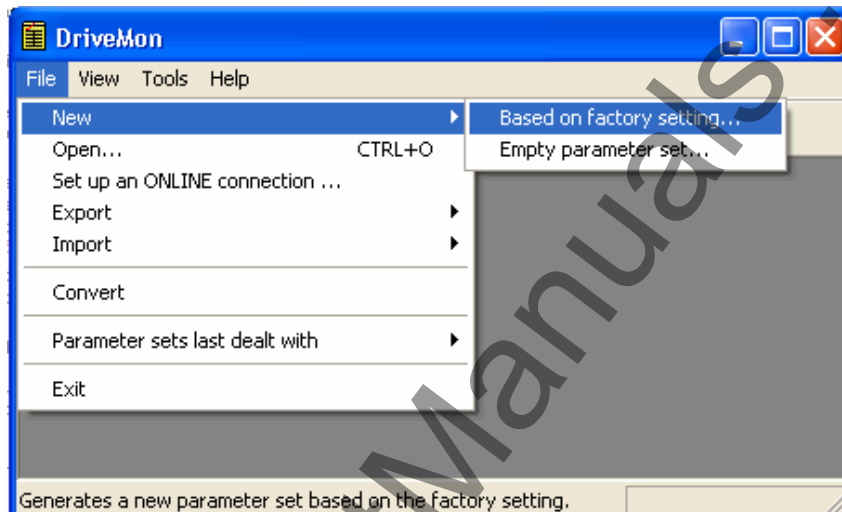


Fig. 5-12 Creating a new drive

Based on factory setting:

- ◆ The parameter list is preassigned with the factory setting values

Empty parameter set:

- ◆ For compilation of individually used parameters

If the parameters of a parameter set that has already been created have to be changed, this can be done by calling the corresponding download file via the "*File* → *Open*" menu function. The last four drives can be opened via "*Parameter sets last dealt with*".

When you create a new drive, the window "Drive Properties" (Fig. 5-13) opens. Here you must enter the following data:

- ◆ In dropdown list box "Device type", select the type of device (e.g. MASTERDRIVES MC). You can only select the devices stored.
- ◆ In dropdown list box "Device version", you can select the software version of the device. You can generate databases for (new) software versions that are not listed when you start online parameterization.
- ◆ You must only specify the bus address of the drive during online operation (switchover with button Online/Offline)

NOTE

The specified bus address must be the same as that of the parameterized SST bus address in SIMOVERT MASTERDRIVES (P700).

No bus address is assigned to the drive with the button "Disconnect network connection".

NOTE

Field "Number of PCD" has no special significance for the parameterization of MASTERDRIVES and should be left at "2".

If the value is changed, it must be/remain ensured that the setting value in the program matches the value in parameter P703 of the drive at all times.

Fig. 5-13 Create file; Drive properties

After confirming the drive properties with **ok** you have to enter the name and storage location of the download file to be created.

5.5.3 Parameterization

5.5.3.1 Structure of the parameter lists, parameterization with DriveMonitor

Parameterization using the parameter list is basically the same as parameterization using PMU (See Chapter 6 "Parameterizing Steps"). The parameter list provides the following advantages:

- ◆ Simultaneous visibility of a larger number of parameters
- ◆ Text display for parameter names, index number, index text, parameter value, binectors, and connectors
- ◆ On a change of parameters: Display of parameter limits or possible parameter values

The parameter list has the following structure:

Field No.	Field Name	Function
1	P. Nr	Here the parameter number is displayed. You can only change the field in menu <i>Free parameterization</i> .
2	Name	Display of the parameter name, in accordance with the parameter list
3	Ind	Display of the parameter index for indexed parameters. To see more than index 1, click on the [+] sign. The display is then expanded and all indices of the parameter are displayed
4	Index text	Meaning of the index of the parameter
5	Parameter value	Display of the current parameter value. You can change this by double-clicking on it or selecting and pressing <i>Enter</i> .
6	Dim	Physical dimension of the parameter, if there is one

With buttons *Offline*, *Online (RAM)*, *Online (EEPROM)* (Fig. 5-14 [1]) you can switch modes. When you switch to online mode, device identification is performed. If the configured device and the real device do not match (device type, software version), an alarm appears. If an unknown software version is recognized, the option of creating the database is offered. (This process takes several minutes.)

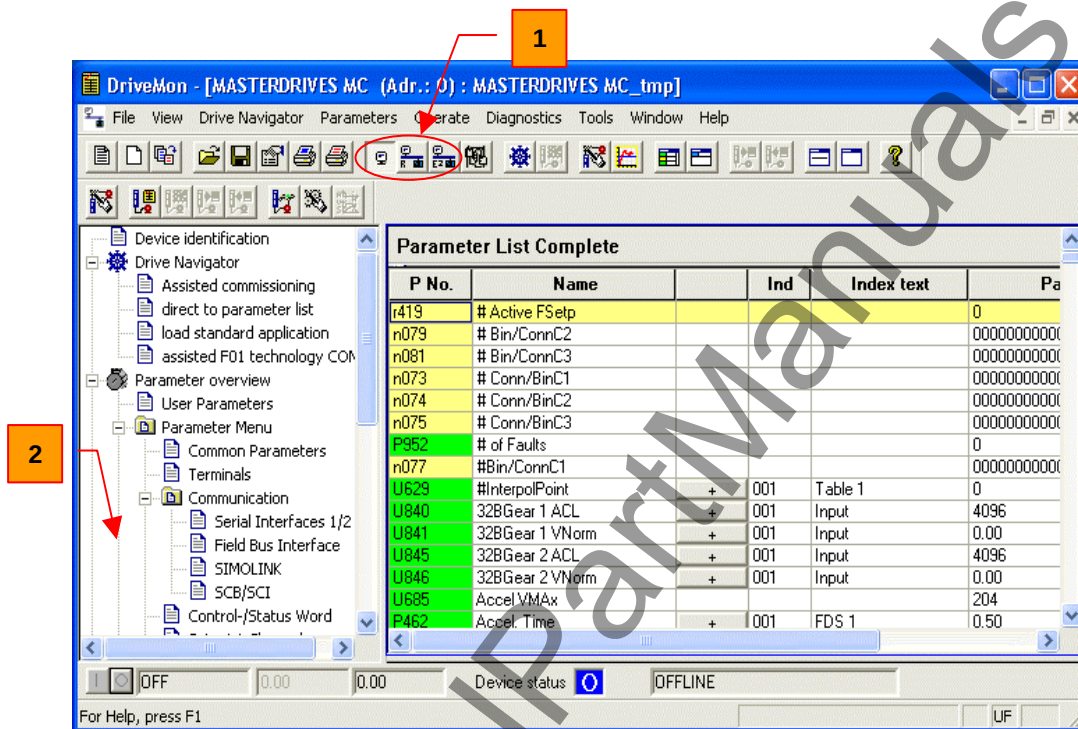


Fig. 5-14 Drive window/parameter list

The DriveMonitor drive window has a directory tree for navigation purposes (Fig. 5-14 [2]). You can deselect this additional operating tool in menu *View - Parameter selection*.

The drive window contains all elements required for the parameterization and operation of the connected device. In the lower bar, the status of the connection with the device is displayed:



Connection and device ok



Connection ok, device in fault state



Connection ok, device in alarm state



Device is parameterized offline



No connection with the device can be established (only offline parameterization possible).

NOTE

If no connection with the device can be established because the device does not physically exist or is not connected, you can perform offline parameterization. To do so, you have to change to offline mode. In that way, you can create an individually adapted download file, which you can load into the device later.

Drive Navigator

This is used to quickly access important functions of the DriveMonitor.
Settings for Drive Navigator under *Tools -> Options* (Fig. 5-16):

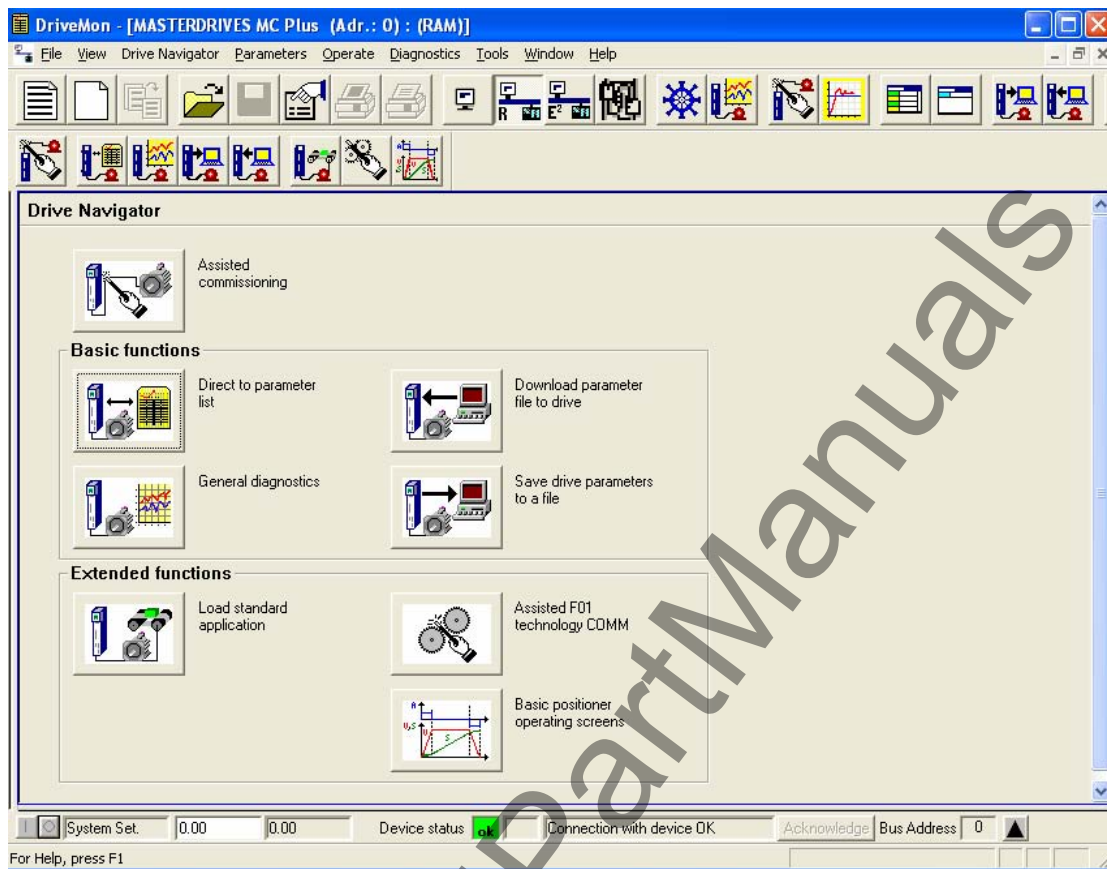


Fig. 5-15 Drive Navigator

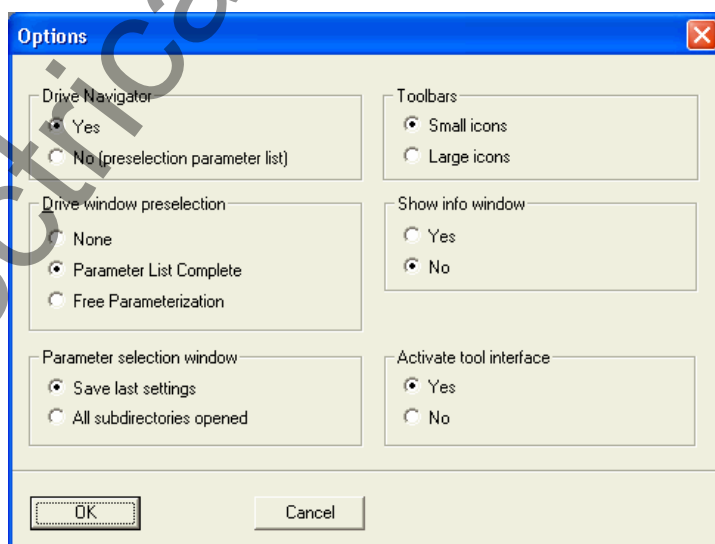


Fig. 5-16 Options menu display

Toolbar of the Drive Navigator



=



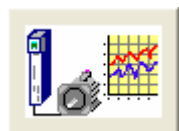
Assisted commissioning



=



Direct to parameter list



=



General diagnostics



=



Save drive parameters to a file



=



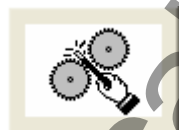
Download parameter file to drive



=



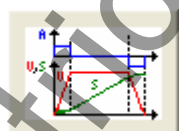
Load standard application



=



Assisted F01 technology COMM



=



Basic positioner operating screens

5.5.3.2 General diagnostics

Via the *Diagnostics* → *General diagnostics* menu the following window opens. This window gives a general overview of the active warnings and faults and their history. Both the warning and the fault number as well as plain text are displayed.

General Diagnostics

Active Warnings		
No.	Warning Text	About
2	SIMOLINK start alarm	...
18	Encoder adjustment	...
19	Encoder data serial protocol	...
23	Motor temperature	...

Aktive Fault				
No.	Fault Text	Fault...	Fault Time	About
153	Request master control enable	0	0000:0000:0017	...

Fault History				
No.	Fault Text	Fault...	Fault Time	About
2	153 Request master control enable	0	0000:0000:0017	...
3	2 Pre-charging fault	1	0000:0000:0017	...

Operat. Hours: 17 d 1 h 17 s

Firmwareversion: V2.20.0

CalcTimeHdroom: 27 %

Drive Temp: 23 °C

Drive Utilizat.: 66 %

DC Bus Volts: 541 V

Output Amps: 13.9 A

Motor Torque: 79.78 %

Motor Temperat.: 35 °C

n(act): 3000 min⁻¹

[Extended Diagnostics](#)

Fig. 5-17 General diagnostics

Via the *Extended Diagnostics* button you can reach the next diagnostics window.

Extended Diagnostics

Graphic Diagnostics

Bus Diagnostics

Trace Function

Cross Reference Binectors

Cross Reference Connectors

Abbrechen

Fig. 5-18 Extended diagnostics

6 Parameterizing Steps

In general, parameterization can be subdivided into the following main steps:

Detailed parameterization

1. Power section definition (P060 = 8)
2. Board definition (P060 = 4)
3. Drive definition (P060 = 5)
4. Function adjustment.

Not all parameterizing steps have to be run through in detail in each case during start-up. It is possible under certain conditions to combine some of the steps and shorten parameterization by using quick procedures. The following quick procedures are possible:

Quick parameterization

1. Parameterizing with user settings
(Fixed setting or factory setting, P060 = 2)
2. Parameterizing with existing parameter files
(Download, P060 = 6)
3. Parameterizing with parameter modules
(Quick parameterization, P060 = 3)

Depending on the specific prevailing conditions, parameterization can be carried out either in detail or in accordance with one of the specified quick procedures.

By activating a fixed setting (P060 = 2), the parameters of the unit can also be reset to the original values.

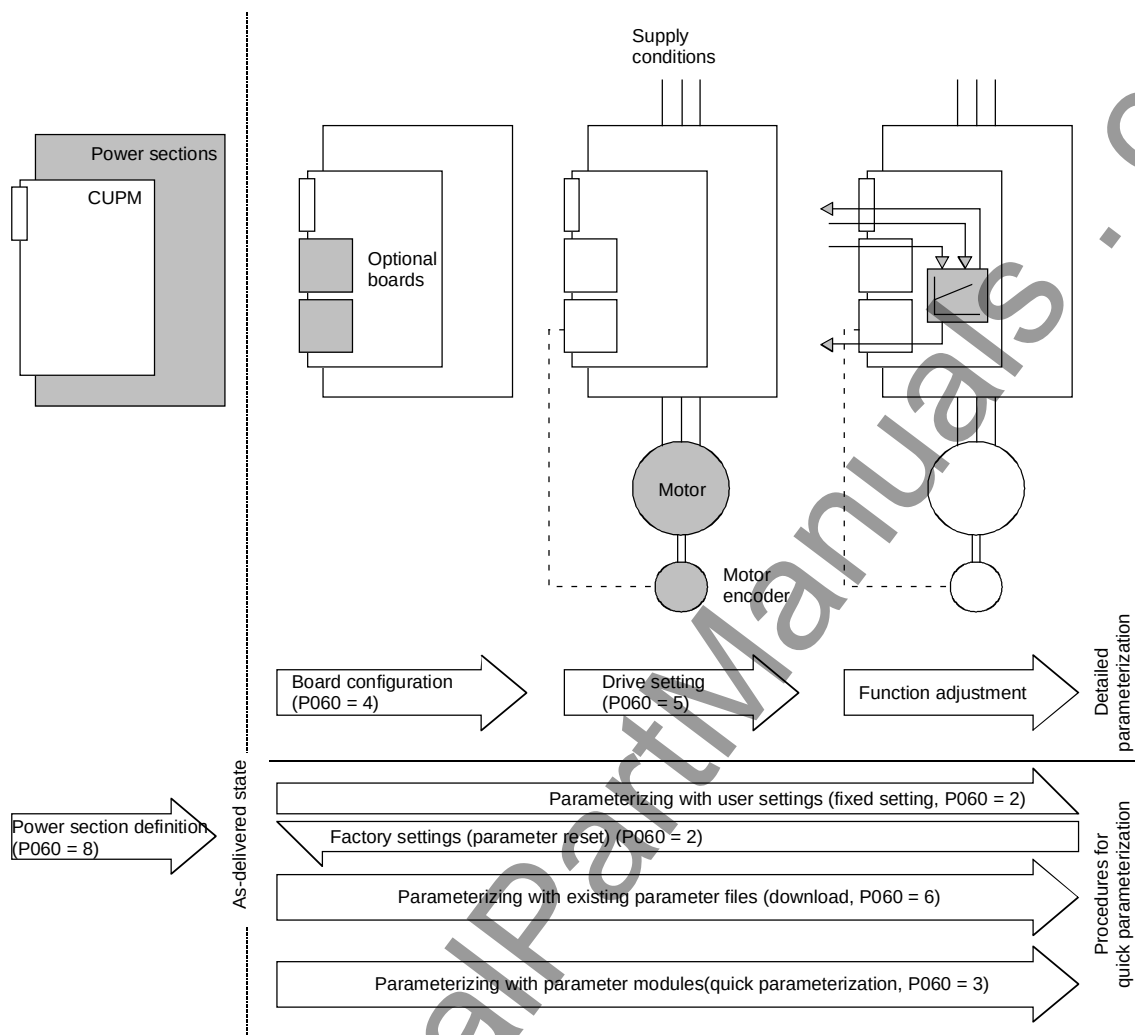


Fig. 6-1 Detailed and quick parameterization

6.1 Parameter reset to factory setting

The factory setting is the defined initial state of all parameters of a unit. The units are delivered with this setting.

You can restore this initial state at any time by resetting the parameters to the factory setting, thus canceling all parameter changes made since the unit was delivered.

The parameters for defining the power section and for releasing the technology options and the operating hours counter and fault memory are not changed by a parameter reset to factory setting.

Parameter number	Parameter name
P050	Language
P070	Order No. 6SE70..
P072	Rtd Drive Amps
P073	Rtd Drive Power
P366	Select FactSet
P947	Fault memory
P949	Fault value
U976	Serial number
U977	PIN

Table 6-1 Parameters that are not changed by the factory setting

If the parameters are reset to the factory setting via one of the parameters (SST1, SST2, SCB, 1.CB/TB, 2.CB/TB), the interface parameters of that interface are not changed either. Communication via that interface therefore continues even after a parameter reset to the factory setting.

Parameter number	Parameter name
P053	Parameterization enable
P700	SST bus address
P701	SST baud rate
P702	SST PKW number
P703	SST PZD number
P704	SST frame failure

Table 6-2 The factory setting is made either via interface SST1 or SST2: Parameters that are not changed by the factory setting either. **None** of the indices of the parameters is changed.

Parameter number	Parameter name
P053	Parameterization enable
P696	SCB protocol
P700	SST bus address
P701	SST baud rate
P702	SST PKW number
P703	SST PZD number
P704	SST frame failure

Table 6-3 The factory setting is made via interface SCB2: Parameters that are not changed by the factory setting either. **None** of the indices of the parameters is changed.

Parameter number	Parameter name
P053	Parameterization enable
P711 to P721	CB parameters 1 to 11
P722	CB/TB frame failure
P918	CB bus address

Table 6-4 The factory setting is made either via interface 1.CB/TB or 2.CB/TB: Parameters that are not changed by the factory setting either. **None** of the indices of the parameters is changed.

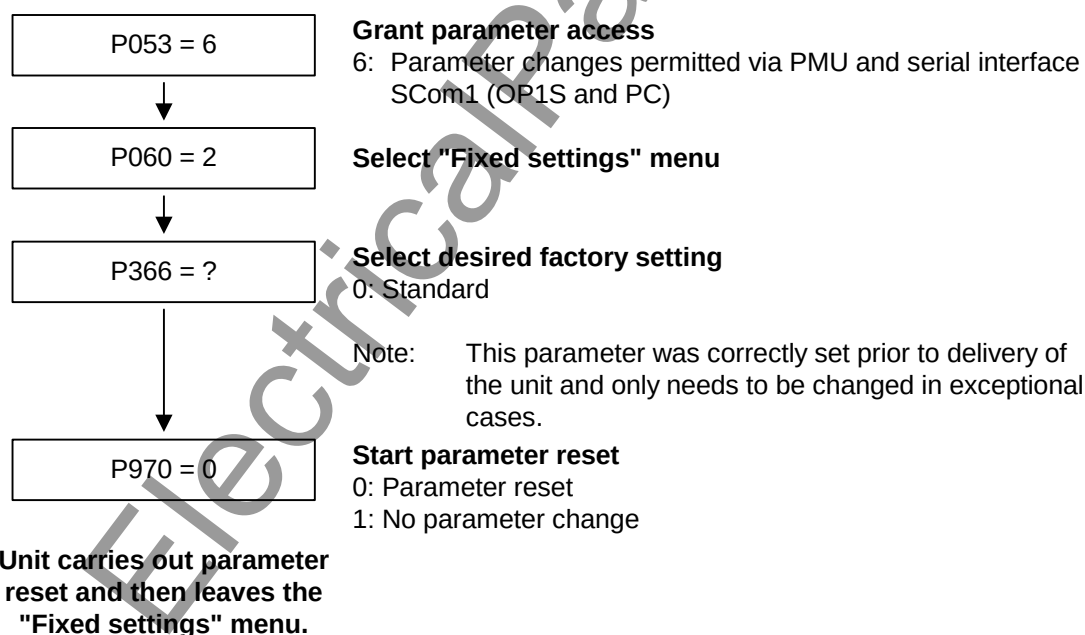


Fig. 6-2

Sequence for parameter reset to factory setting

6.2 Detailed parameterization

Detailed parameterization should always be used in cases where the application conditions of the units are not exactly known beforehand and detailed parameter adjustments need to be carried out locally. An example of a typical application is initial start-up.

6.2.1 Power section definition

During the power section definition, the control electronics is informed about which power section it is working with. This step is necessary for Compact, chassis and cabinet units. On these units, the CUPM control board is accommodated in the electronics box and is not firmly connected to the power section.

The power section definition has already been completed in the as-delivered state. It is therefore only necessary on replacement of the CUPM or after loading a firmware version with a different parameter database (version ID: Change the 1st decimal place) and not under normal conditions.

CAUTION

If CUPM boards are changed over between different units without the power section being re-defined, the unit can be destroyed when connected up to the voltage supply and energized.

If a CUPM board which has already been parameterized is inserted into a unit with a different power section, an automatic parameterization via parameter P115 = 1 has to be carried out in the drive setting status (P60 = 5) after the power section definition.

NOTE

From firmware version V 2.20 and higher, power sections above 250 kW can be selected. Selection is only possible if option F02 is enabled via a PIN (n978.2 = 1).
To activate option F02 please refer to Chapter 11.10 "Power Extension PIN F02".

The unit has to be switched to the "Power section definition" state for carrying out the power section definition. This is done by selecting the "Power section definition" menu. The power section is then defined in this menu by inputting a code number.

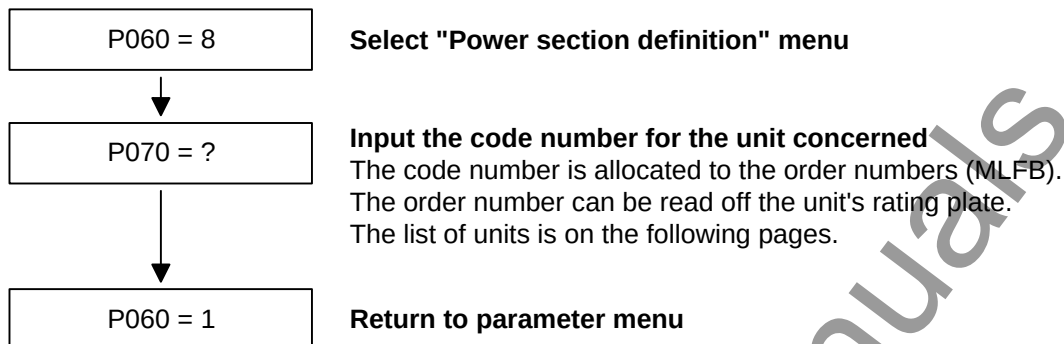


Fig. 6-3 Sequence for performing the power section definition

NOTE

To check the input data, the values for the converter supply voltage in P071 and the converter current in P072 should be checked after returning to the parameter menu. They must tally with the data given on the unit rating plate.

**Frequency
converter, Compact
PLUS AC-AC type**

Order number	In [A]	P070
6SE7011-5EP□0	1.5	1
6SE7013-0EP□0	3.0	3
6SE7015-0EP□0	5.0	5
6SE7018-0EP□0	8.0	7
6SE7021-0EP□0	10.0	9
6SE7021-4EP□0	14.0	13
6SE7022-1EP□0	20.5	15
6SE7022-7EP□0	27.0	17
6SE7023-4EP□0	34.0	19

**Inverter, Compact
PLUS DC-AC type**

Order number	In [A]	P070
6SE7012-0TP□0	2.0	2
6SE7014-0TP□0	4.0	4
6SE7016-0TP□0	6.1	6
6SE7021-0TP□0	10.2	8
6SE7021-3TP□0	13.2	12
6SE7021-8TP□0	17.5	14
6SE7022-6TP□0	25.5	16
6SE7023-4TP□0	34.0	18
6SE7023-8TP□0	37.5	20
6SE7024-7TP□0	47.0	22
6SE7026-0TP□0	59.0	24
6SE7027-2TP□0	72.0	26

□ = 5 corresponds to MASTERDRIVES Motion Control
 □ = 7 corresponds to MASTERDRIVES Motion Control Performance 2

**Frequency
converter, Compact
AC-AC type**

Order number	In [A]	PWE
6SE7016-1EA□1	6.1	3
6SE7018-0EA□1	8.0	9
6SE7021-0EA□1	10.2	11
6SE7021-3EB□1	13.2	18
6SE7021-8EB□1	17.5	25
6SE7022-6EC□1	25.5	35
6SE7023-4EC□1	34.0	42
6SE7023-8ED□1	37.5	46
6SE7024-7ED□1	47.0	52
6SE7026-0ED□1	59.0	56
6SE7027-2ED□1	72.0	66

**Inverter, Compact
DC-AC type**

Order number	In [A]	PWE
6SE7016-1TA□1	6.1	4
6SE7018-0TA□1	8.0	10
6SE7021-0TA□1	10.2	12
6SE7021-3TB□1	13.2	19
6SE7021-8TB□1	17.5	26
6SE7022-6TC□1	25.5	36
6SE7023-4TC□1	34.0	43
6SE7023-8TD□1	37.5	47
6SE7024-7TD□1	47.0	53
6SE7026-0TD□1	59.0	57
6SE7027-2TD□1	72.0	67

□ = 5 corresponds to MASTERDRIVES Motion Control
 □ = 7 corresponds to MASTERDRIVES Motion Control Performance 2

**Frequency
converter, chassis-
type AC-AC units**

Order number	In [A]	PWE
6SE7031-0EE70	92.0	74
6SE7031-2EF70	124.0	82
6SE7031-8EF70	186.0 ¹⁾ 155.0 ²⁾	98
6SE7032-1EG70	210.0 ¹⁾ 175.0 ²⁾	102
6SE7032-6EG70	260.0 ¹⁾ 218.0 ²⁾	108
6SE7033-2EG70	315.0 ¹⁾ 262.0 ²⁾	112
6SE7033-7EG70	370.0 ¹⁾ 308.0 ²⁾	116
6SE7035-1EK70	510.0 ¹⁾ 423.0 ²⁾ ³⁾	147
6SE7036-0EK70	590.0 ¹⁾ 491.0 ²⁾ ³⁾	151
6SE7037-0EK70	690.0 ⁴⁾	164

**Inverter, chassis-
type DC-AC units**

Order number	In [A]	PWE
6SE7031-0TE70	92.0	75
6SE7031-2TF70	124.0	83
6SE7031-8TF70	186.0 ¹⁾ 155.0 ²⁾	99
6SE7032-1TG70	210.0 ¹⁾ 175.0 ²⁾	103
6SE7032-6TG70	260.0 ¹⁾ 218.0 ²⁾	109
6SE7033-2TG70	315.0 ¹⁾ 262.0 ²⁾	113
6SE7033-7TG70	370.0 ¹⁾ 308.0 ²⁾	117
6SE7035-1TJ70	510.0 ¹⁾ 423.0 ²⁾ ³⁾	120
6SE7036-0TJ70	590.0 ¹⁾ 491.0 ²⁾ ³⁾	123
6SE7037-0TJ70	690.0 ⁴⁾	163
6SE7038-6TK70	860.0 ⁴⁾	127
6SE7041-1TK70	1100.0 ⁴⁾	135
6SE7041-3TL70	1300.0 ⁴⁾	154

1) theoretical rated output current at 3 kHz pulse frequency; rated output current can only be achieved if the Power Extension PIN is entered.

2) rated output current at 5 kHz pulse frequency; the permissible rated output current will be reduced further at higher pulse frequencies (see technical data, derating)

3) This device is a chassis unit (from type J). The overload over 30 s is therefore limited to 1.36 x the rated output current at 5 kHz.

4) This unit is a so-called chassis unit and can only be operated up to a maximum pulse frequency of 2.7 kHz. The overload for 30 s is limited to 1.36 times the rated output current.

**Water-cooled AC-AC
frequency converter**

Order number	In [A]	PWE
6SE7035-1EK70 -1AA0 or -1AA1	510 ¹⁾ 423 ^{2) 3)}	233
6SE7036-0EK70 -1AA0 or -1AA1	590 ¹⁾ 491 ^{2) 3)}	237
6SE7037-0EK70 -1AA0 oder -1AA1	690.0 ⁴⁾	168

**Water-cooled DC-AC
inverter**

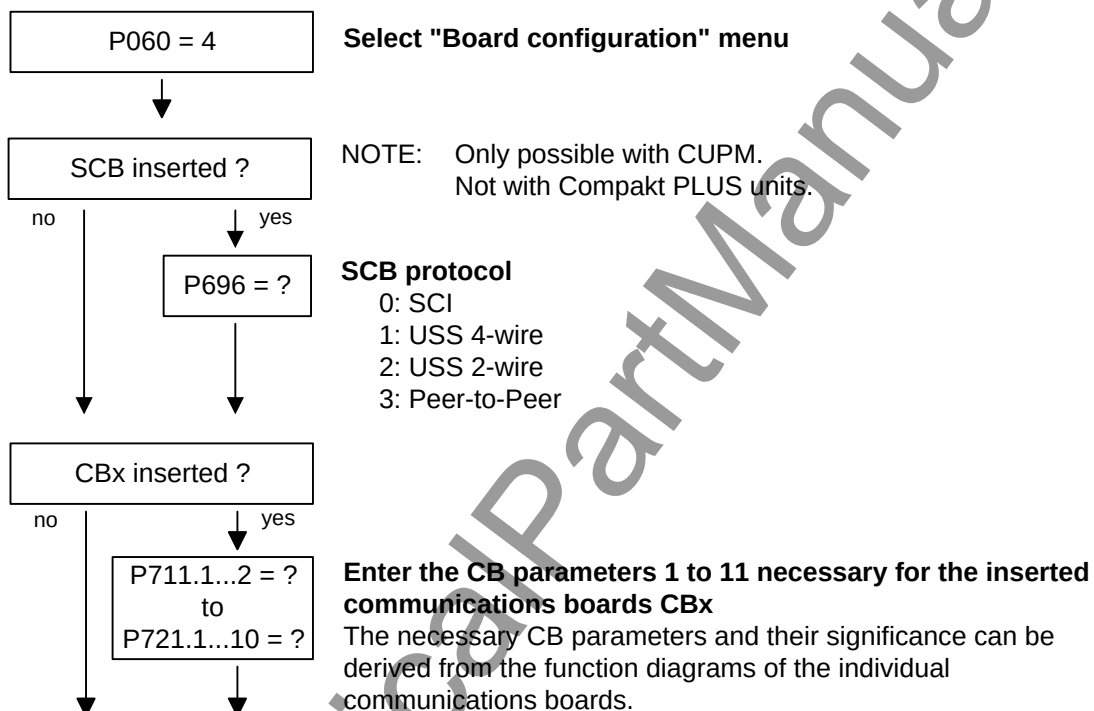
Order number	In [A]	PWE
6SE7035-1TJ70 -1AA1 or -1AA0	510.0 ¹⁾ 423 ^{2) 3)}	206
6SE7036-0TJ70 -1AA1 or -1AA0	590.0 ¹⁾ 491 ^{2) 3)}	209
6SE7037-0TJ70 -1AA0 or -1AA1	690.0 ⁴⁾	167
6SE7041-3TL70 -1AA0 or -1AA1	1300.0 ⁴⁾	199
6SE7038-6TK70 -1AA0 or -1AA1	860.0 ⁴⁾	213
6SE7041-1TK70 -1AA0 or -1AA1	1100.0 ⁴⁾	221

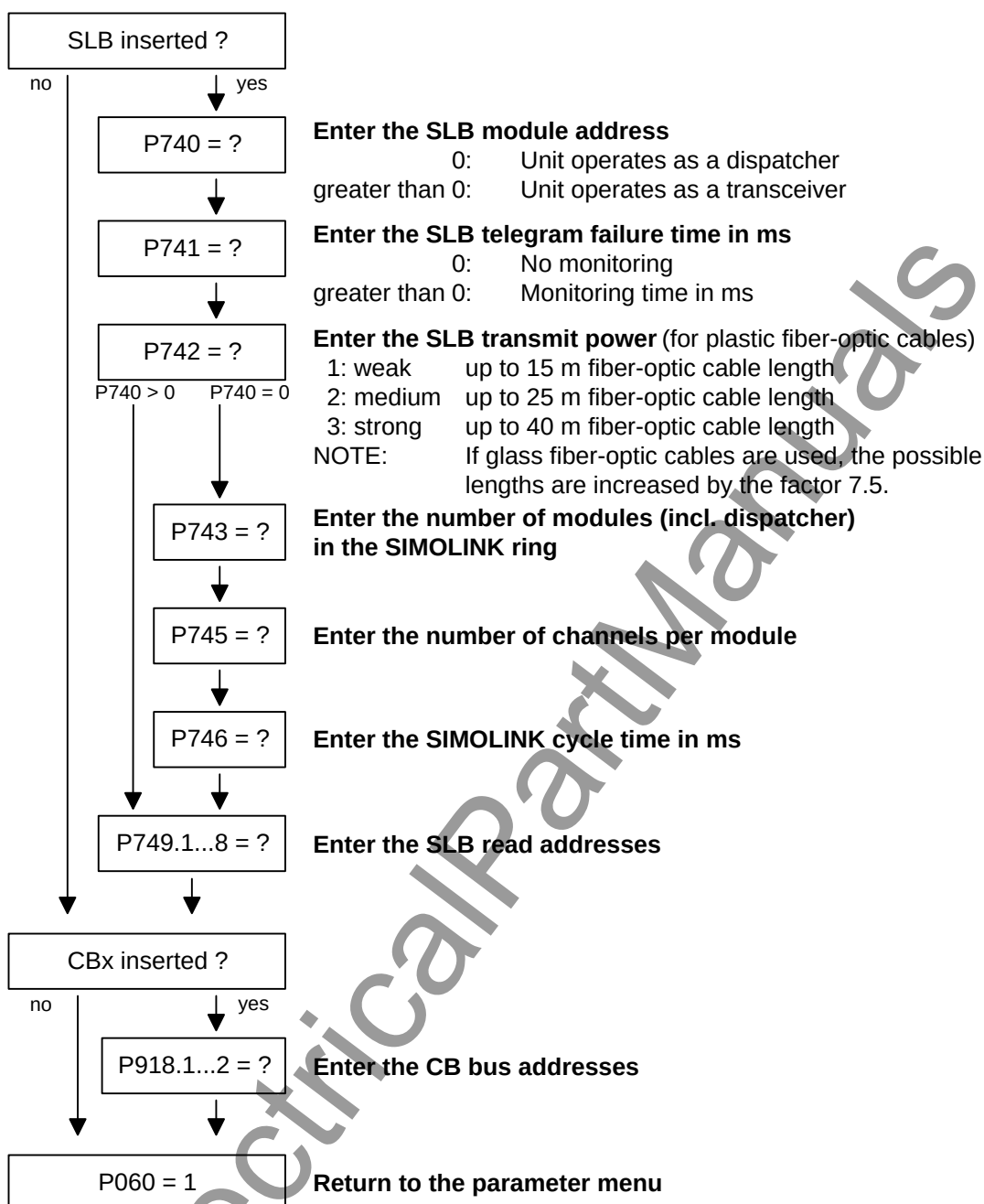
- 1) theoretical rated output current at 3 kHz pulse frequency; rated output current can only be achieved if the Power Extension PIN is entered.
- 2) rated output current at 5 kHz pulse frequency; the permissible rated output current will be reduced further at higher pulse frequencies (see technical data, derating)
- 3) This device is a chassis unit (from type J). The overload over 30 s is therefore limited to 1.36 x the rated output current at 5 kHz.
- 4) This unit is a so-called chassis unit and can only be operated up to a maximum pulse frequency of 2.7 kHz. The overload for 30 s is limited to 1.36 times the rated output current.

6.2.2 Board configuration

During board configuration, the control electronics is informed in what way the installed optional boards have to be configured. This step is always necessary when optional boards CBx or SLB are used.

The unit must be switched to the "Board configuration" status for this purpose. This is done by selecting the "Board configuration" menu. In this menu, parameters are set that are required for adapting the optional boards to the specific application (e.g. bus addresses, baud rates, etc.). After leaving the menu, the set parameters are transferred and the optional boards are initialized.





Board codes

The visualization parameter r826.x is used for displaying the board codes. These codes enable the type of installed electronic boards to be determined.

Parameter	Index	Position
r826	1	Basic board
r826	2	Slot A
r826	3	Slot B
r826	4	Slot C
r826	5	Slot D
r826	6	Slot E
r826	7	Slot F
r826	8	Slot G

If a technology board (T100, T300, T400) or an SCB1 or SCB2 is inserted in mounting positions 3 or 2, their board code can be found in the following indices:

Parameter	Index	Position
r826	5	Mounting position 2
r826	7	Mounting position 3

General board codes

Parameter value	Significance
90 to 109	Mainboards or Control Unit
110 to 119	Sensor Board (SBx)
120 to 129	Serial Communication Board (Scx)
130 to 139	Technology Board
140 to 149	Communication Board (Cbx)
150 to 169	Special boards (Ebx, SLB)

Special board codes

Board	Significance	Parameter value
CUVC	Control Unit Vector Control	92
CUMC	Control Unit Motion Control	93
CUMC+	Motion Control Compact PLUS	94
CUVC+	Control Unit Vector Control Compact PLUS	95
CUPM	Control Unit Motion Control Performance 2	96
CUMP	Control Unit Motion Control Compact PLUS Performance 2	97
CUSA	Control Unit Sinus AFE	108
SBP	Sensor Board Pulse	111
SBM	Sensor Board Encoder / Multiturn	112
SBM2	Sensor Board Encoder / Multiturn 2	113
SBR1	Sensor Board Resolver 1	114
SBR2	Sensor Board Resolver 2	115
SCB1	Serial Communication Board 1 (fiber-optic cable)	121
SCB2	Serial Communication Board 2	122
T100	Technology board	131
T300	Technology board	131
T400	Technology board	134
CBX	Communication Board	14x
CBP	Communication Board PROFIBUS	143
CBD	Communication Board DeviceNet	145
CBC	Communication Board CAN Bus	146
CBL	Communication Board CC-Link	147
CBP2	Communication Board PROFIBUS 2	148
EB1	Expansion Board 1	151
EB2	Expansion Board 2	152
SLB	SIMOLINK bus interface	161

6.2.3 Drive setting

During the drive setting, the control electronics is informed about the incoming voltage supply with which the drive converter is operating, about the connected motor and about the motor encoder. In addition, the motor control (V/f open-loop control or vector control) and the pulse frequency are selected. If required, the parameters necessary for the motor model can be calculated automatically. Furthermore, the normalization values for current, voltage, frequency, speed and torque signals are determined during the drive setting.

For start-up of the induction motor, first enter the manufacturer's parameters completely (see below):

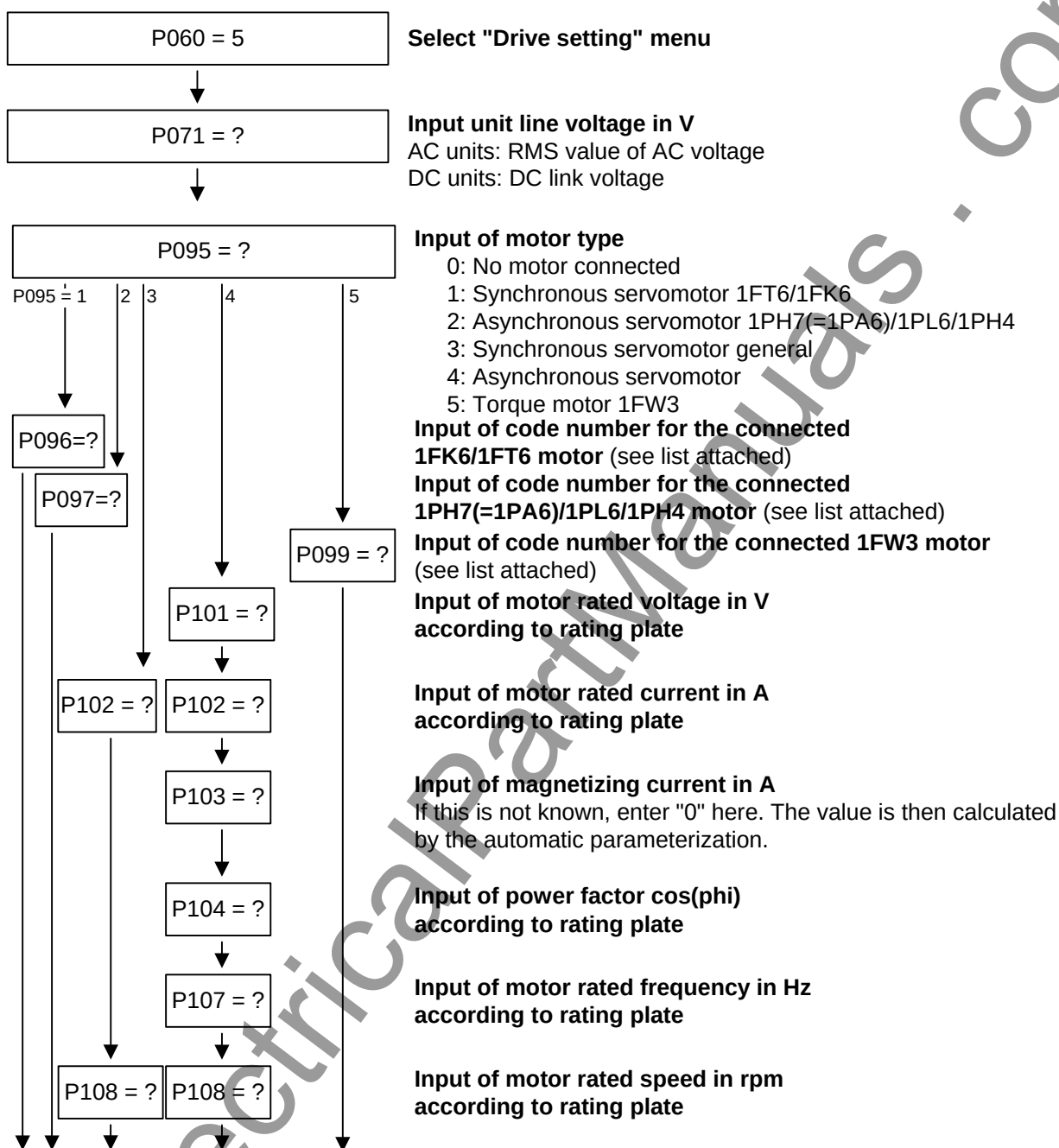
- ◆ In doing so, you must observe whether the induction motor has a star or a delta connection.
- ◆ You must always use the S1 data from the rating plate.
- ◆ You must enter the rating data for **mains duty** (not converter duty).
- ◆ You must always enter the correct rated motor current **P102** (rating plate). If there are two different rated currents on the rating plate for special fan motors, you must use the value for $M \sim n$ for constant torque (not $M \sim n^2$). A higher torque can be set with the torque and active-current limits.
- ◆ The accuracy of the rated motor current has a direct effect on the torque accuracy, as the rated torque is normalized to the rated current. If a rated current is increased by 4 %, this will also approximately result in a 4 % increase in the torque (referred to the rated motor torque).
- ◆ For group drives, you have to enter the total rated current
 $P102 = x \cdot I_{mot, rated}$
- ◆ As the rated magnetizing current **P103** (not to be confused with the no-load current during operation with rated frequency **P107** and rated voltage **P101**) is usually not known, you can first enter 0.0 %. An approximation value is calculated using power factor (cosPHI) **P104** and then entered in P103.
 Experience shows that the approximation supplies values that are rather on the large side in the case of motors with a high rating (over 800 kW), whereas it supplies values that are slightly too low in the case of motors with low rating (below 22 kW).
 The magnetizing current is defined as a field-generating current component during operation at the rated point of the machine ($U = P101$, $f = P107$, $n = P108$, $i = P102$).
- ◆ $P294 = 1$ ($\hat{=}$ controlled, flux controller not active) should be set for asynchronous motors in combination with the setting $P296 < 3$. In V1.40 and higher, this is done in automatic parameterization.
- ◆ In field weakening, operation is permitted only up to twice the motor transition frequency ($2 \times P293$). When the controller structure optimized for field weakening ($P296 = 3$) is selected for asynchronous motors, field weakening is permitted up to 5 times the motor corner frequency ($5 \times P293$).

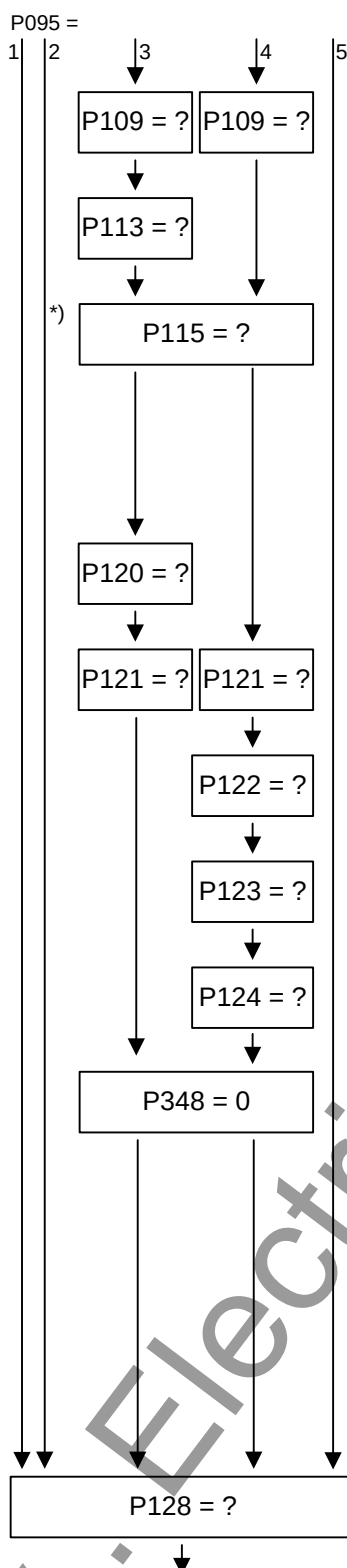
- ◆ Input of rated frequency **P107**, rated speed **P108** and number of pole pairs **P109** is necessary.
- ◆ In the case of induction motors, you should enter in **P108** not the synchronous no-load speed, but the true motor rated speed, i.e. the slip frequency at rated load must be given by parameters **P107...P109**.
- ◆ The rated motor slip ($1 - \text{P108}/60 \times \text{P109}/\text{P107}$) should usually be greater than 0.35 % $\times \text{P107}$.
These low values are, however, only achieved in the case of motors with a very high rating (above approx. 1000 kW).
Motors with average rating (45..800 kW) have slip values around 2.0...0.6 %.
Motors with low rating (below 22 kW) can also have slip values up to 10 %.
- ◆ If the rated motor frequency (engineered!) is below 8 Hz, you have to set **P107** = 8.0Hz in the drive setting. The rated motor voltage **P101** has to be calculated in the ratio $8 \text{ Hz} / f_{\text{Mot,N}}$ and the rated motor speed **P108** should result in a much greater slip:
$$\text{P108} = ((8 \text{ Hz} - \text{P107}_{\text{old}}) \times 60 / \text{P109}) + \text{P108}_{\text{old}}$$

NOTE

When leaving the "Drive setting" menu, the entered parameter values are checked for plausibility. Parameter settings that are not plausible will lead to a fault. The wrongly set parameters are entered in parameter r949 (fault value).

If current control is selected as the motor control, a suitable sensor board (SBx) must be inserted beforehand and a permissible motor encoder selected. If this is not carried out, the unit will also generate a fault when it tries to leave the "Drive setting" menu.





Input the pole pair number of the motor

Input the motor rated torques per rating plate in Nm

Start of calculation of derived motor data

For this purpose, set P115 from 0 to 1.

*) That is necessary with P095 = 2 only in the case of firmware versions 1.24 and older.

The motor parameters for setting the current control are calculated from the rating plate data. At the end of the calculation, P115 is automatically set to 0.

Input the main field inductance in mH

Pre-assigned during calculation of derived motor data (P115).

Input the stator resistance in mOhm

Pre-assigned during calculation of derived motor data (P115).

Input the total leakage reactance in mOhm

Pre-assigned during calculation of derived motor data (P115).

Input the stator reactance in mOhm

Pre-assigned during calculation of derived motor data (P115).

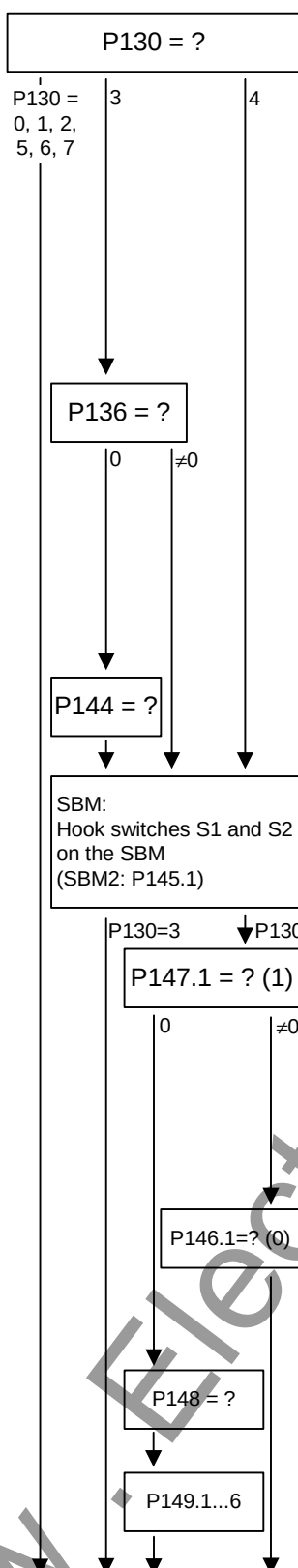
Input the rotor time constants in ms

Pre-assigned during calculation of derived motor data (P115).

Automatic parameterization deactivates dead time compensation.

NOTE: If automatic motor identification is performed after leaving the "Start drive" status (r001 = 005), no further inputs are necessary for parameters P120 to P124. For vector control (P290 = 0), automatic motor identification should always be performed (see section entitled "Motor identification"). If the motor is coasting, a no-load measurement is recommended (see section entitled "No-load measurement"). Thus optimum results regarding torque accuracy can be achieved.

Input the maximum output current in A

**Select the motor encoder**

- 0: automatic encoder detection
- 1: 2-pole resolver (SBR)
- 2: Resolver with pole pair number of motor (SBR)
- 3: Encoder (SBM)
- 4: Multiturn encoder (SBM)
- 5: Pulse encoder in Slot C (SBP)
- 6: Pulse encoder not in Slot C (SBP)
- 7: Encoder without C/D track

NOTE: Asynchronous motors 1PA6, 1PL6, 1PH4, and 1PH7 with encoder are usually supplied with encoder ERN1381 without C/D tracks.

Increment encoder

- 0: Pulse number is not a power of 2.
Pulse number in P144 is applicable.
- 9: $2^9 = 512$
- 10: $2^{10} = 1024$
- 11: $2^{11} = 2048$
- 12: $2^{12} = 4096$
- 15: Increment = 2048 and zero pulse is not evaluated (from V1.24)

Motor encoder pulse number**Voltage supply encoder**

Is set at the factory to 5 V to match the standard encoder used with SIEMENS motors.
Also see function diagram 240.

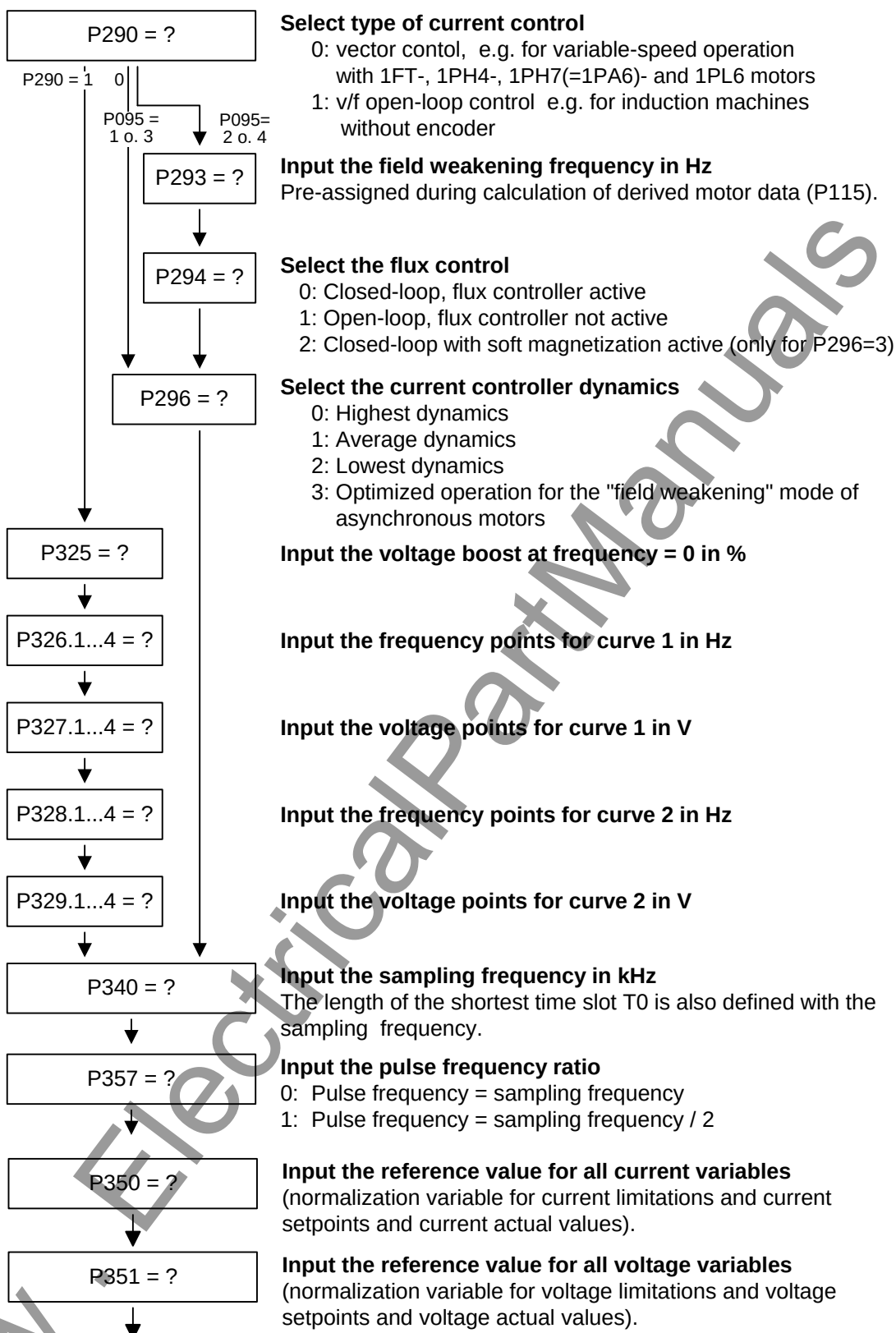
Setting of P147:

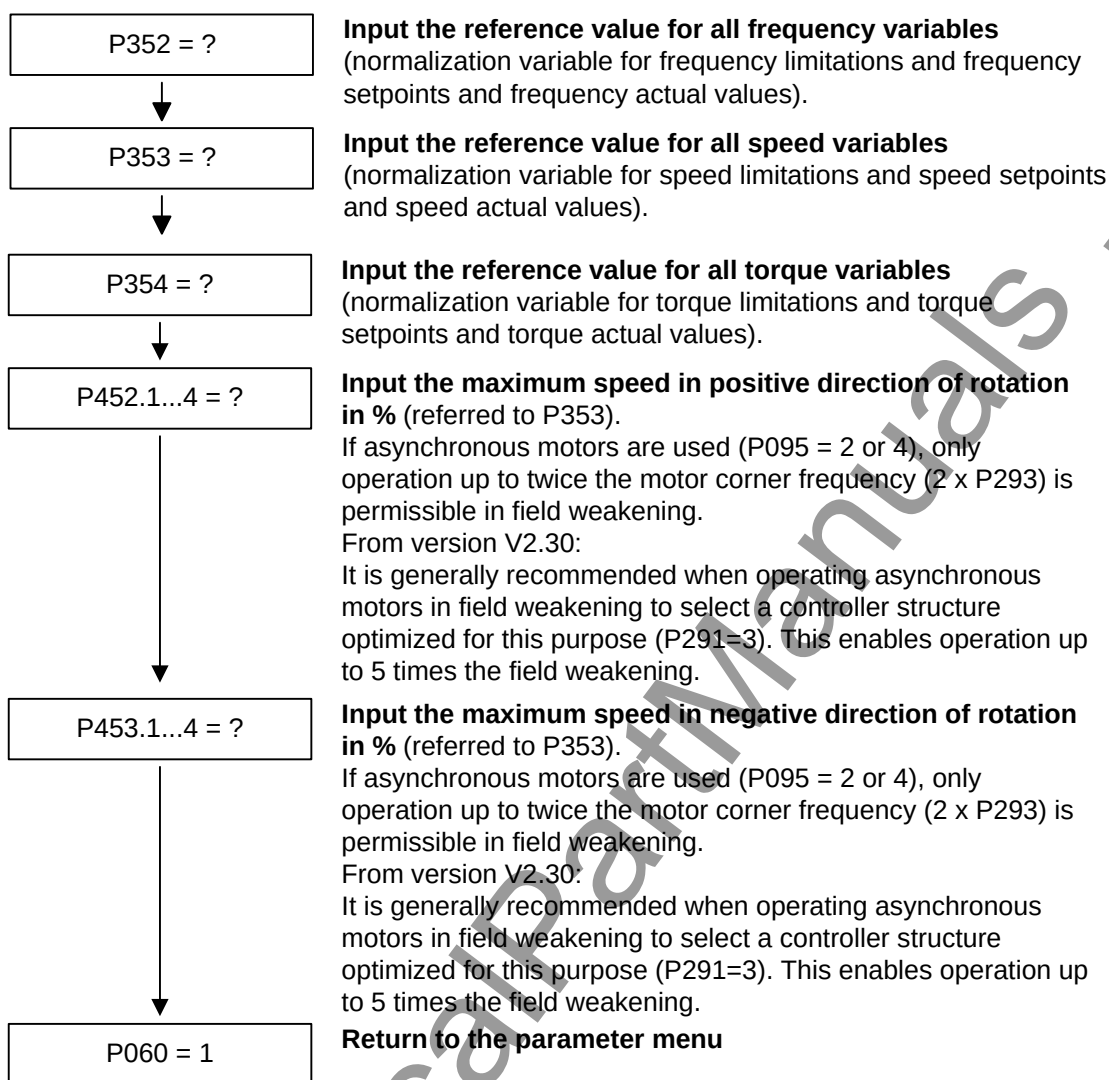
- 0: No standard encoder, parameterization in P148, P149
- 1: Encoder EQN1325 (Heidenhain) EnDat
- 2: Encoder ECN1313 (Heidenhain) EnDat
- 6: EnDat (Heidenhain)
- 7: EQI1325 (Heidenhain)
- 8: Encoder EQN1125 (Heidenhain) EnDat
- 9: Encoder ECN1113 (Heidenhain) EnDat

Zero offset in revolutions

Project planning note:
The travel range of the linear axes must be within the range of the encoder. Otherwise the mapping range will be shifted with the zero offset.

Increment multiturn encoder**Configuration Protocol**



**NOTE**

When leaving the "Drive setting" menu, the entered parameter values are checked for plausibility. Parameter settings that are not plausible will lead to a fault. The wrongly set parameters are entered in parameter r949 (fault value).

If current control is selected as the motor control, a suitable sensor board (SBx) must be inserted beforehand and a permissible motor encoder selected. If this is not carried out, the unit will also generate a fault when it tries to leave the "Drive setting" menu.

6.2.4 Motor identification

6.2.4.1 Standstill measurement

From Version V1.30 onwards, automatic motor identification is available. In the case of Siemens motors ($P095 = 1$ or 2) the motor type is first selected in $P096$ or $P097$. In the case of non-Siemens motors ($P095 = 3$ or 4), the rating plate data and number of pole pairs have to be entered, and then automatic parameterizing is called with $P115 = 1$.

After exit from the "drive initial start-up" status with $P060 = 1$, $P115 = 2$ is set and hence motor identification is selected. The converter must now be switched in within 30 s so that measuring can start. The alarm A078 is set during the 30 s.

WARNING



The motor shaft can move slightly during the measurement operation. The motor cables are live. Voltages are present at the converter output terminals and hence also at the motor terminals; they are therefore hazardous to touch.

DANGER



It must be ensured that no danger for persons and equipment can occur by energizing the power and the unit.

If measurement is not started within 30 s or if it is interrupted by an OFF command, error F114 is set. The converter status during measurement is "Motid-Still" ($r001 = 18$). Measurement is ended automatically, and the converter reverts to the status "Ready for start-up" ($r001 = 009$).

Depending on the motor size (rotor time constant), measurement will take 2 to 10 minutes.

In current-controlled mode ($P290 = 0$), automatic motor identification should **always** be performed during initial start-up.

6.2.4.2 No-load measurement

From version 2.30 onwards, a no-load measurement for asynchronous motors is available.

In order to perform this measurement, the motor shaft must not be connected to any mechanical loads as this would distort the measurement results.

For Siemens asynchronous motors (P095 = 1) first of all the motor type is selected in P097. For motors of other makes (P095 = 4) the rating plate data and the pole pair number have to be entered and then automatic parameterization has to be called up with P115 = 1. After leaving the "Drive initial start-up" state with P060 = 1 the unit achieves the „Ready for start-up" state (r001 = 009).

Now P115 = 4 is set and hence the no-load measurement is selected. The converter must now be switched in within 30 seconds so that the measurement can run. During these 30 s alarm A078 is set.

WARNING



The motor shaft turns during the measurement. The motor cables are live. Voltages are present at the converter output terminals and hence also at the motor terminals; they are therefore hazardous to touch.

DANGER



It must be ensured that no danger for persons and equipment can occur by energizing the power and the unit and as a result of the moving motor shaft.

If the measurement is not started within 30 s or it is interrupted with an OFF command, fault F114 is output. The converter status during the measurement is "Motid-Opt" (R001 = 19). The measurement is ended automatically, the converter returns to the "Ready for start-up" status (r001=009).

6.2.5 Function adjustment

Once the description of the hardware has been completed, function adjustment has to take place. The function blocks available in the unit are selected, interconnected and adjusted to suit the specific application. Parameterization is carried out in the parameter menu. The function diagrams serve as a basis for the function adjustment. Please refer to the parameter list for more detailed information on the individual parameters. The interconnectable binectors and connectors are listed in the relevant binector and connector lists.

The indicated lists are attached hereto.

6.3 Quick parameterization procedures

The following quick procedures are always used in cases where the application conditions of the units are exactly known and no tests and related extensive parameter corrections are required. Typical examples of applications for quick parameterization are when units are installed in standard machines or when a unit needs replacing.

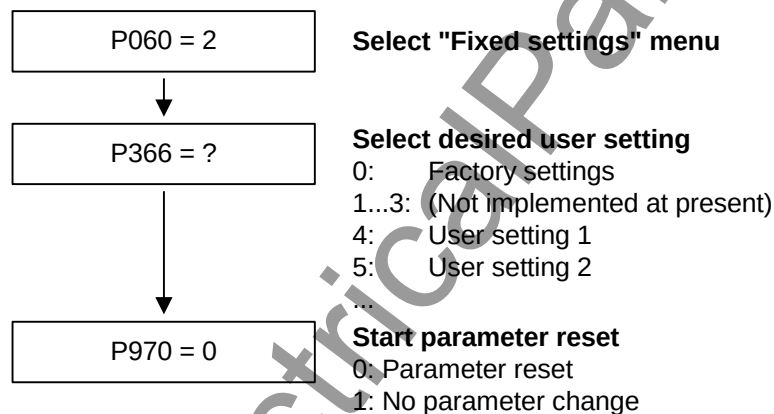
6.3.1 Parameterizing with user settings

During parameterization by selecting user-specific fixed settings, the parameters of the unit are described with values that are permanently stored in the software. In this manner, it is possible to carry out the complete parameterization of the units in one step just by setting a few parameters.

The user-specific fixed settings are not contained in the standard firmware; they have to be compiled specifically for the customer.

NOTE

If you are interested in the provision and implementation of fixed settings tailored to your own requirements, please get in contact with your nearest SIEMENS branch office.



Unit carries out parameter reset and then leaves the "Fixed settings" menu.

Fig. 6-4

Sequence for parameterizing with user settings

6.3.2 Parameterizing by loading parameter files (download P060 = 6)

Download

When parameterizing with download, the parameter values stored in a master unit are transferred to the unit to be parameterized via a serial interface. The following can serve as master units:

1. OP1S operator control panel
2. PCs with DriveMonitor service program
3. Automation units (e.g. SIMATIC)

The interfaces SCom1 and SCom2 with USS protocol of the basic unit (SCom2 not in the case of units of the Compact PLUS type) and field bus interfaces used for parameter transfer (e.g. CBP for PROFIBUS DP) can serve as serial interfaces.

Using download, all changeable parameters can be set to new values.

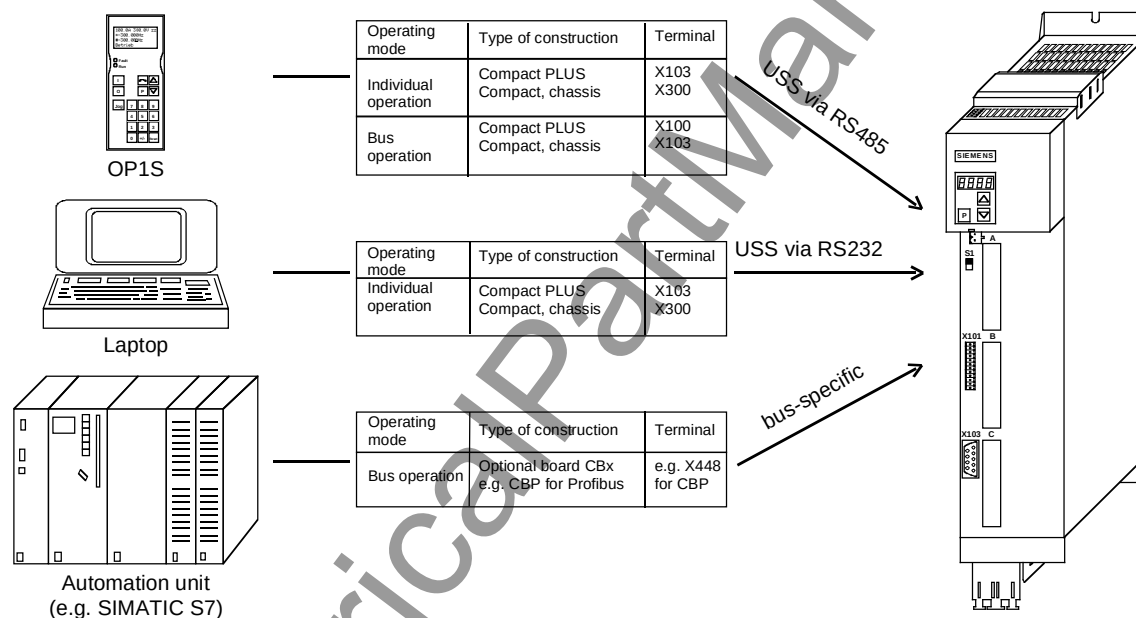


Fig. 6-5

Parameter transfer from various sources by download

Downloading with the OP1S

The OP1S operator control panel is capable of upreading parameter (Upread or Upload) sets from the units and storing them. These parameter sets can then be transferred to other units by download. Downloading with the OP1S is thus the preferred method of parameterizing replacement units in a service case.

During downloading with the OP1S, it is assumed that the units are in the as-delivered state. The parameters for the power section definition are thus not transferred. If a PIN has been entered to release optional technology functions, this is also not overwritten during downloading. (Refer to Section "Detailed parameterization, power section definition")

Parameter number	Parameter name
P060	Menu selection
P070	Order No. 6SE70..
P072	Rtd Drive Amps(n)
P073	Rtd Drive Power(n)
P700	IF bus address
P701	IF baudrate
P702	IF no. of PKW
P703	IF no. of PZD
U977	PIN

Table 6-5 Parameters you cannot overwrite during download

Download with DriveMonitor

By using the DriveMonitor PC program, you can upload parameter sets from the units, store them on the hard disk or on floppy disks and transfer them back to the units by download. You have the additional possibility of editing the parameters off-line and of creating parameter files especially for your application. These files do not have to contain the complete parameter scope. They can be limited to parameters that are relevant for the particular application.

On downloading with DriveMonitor, no power section data are written to protect the device. Similarly, overwriting of the communication parameters and the PIN enable are prevented.

Parameter number	Parameter name
P060	Menu selection
P070	Order No. 6SE70..
P072	Conv.current(s)
P073	Conv.power(s)
P700	IF bus address
P701	IF baudrate
P702	IF no. of PKW
P703	IF no. of PZD
P836	Optional card data download
P850 – P899	OP special parameters
P918	CB bus address
P952	Number of faults
P970	Factory setting
P971	EEPROM accept.
U976	Product number
U977	PIN

Table 6-6 Parameters that cannot be changed on downloading with DriveMonitor

NOTE

Successful parameterization of the units by download is only ensured if the unit is in the "Download" status when the data is being transferred. Transition into this status is achieved by selecting the "Download" menu in P060.

P060 is automatically set to 6 after the download function has been activated in the OP1S or in the DriveMonitor service program.

If the CU of a converter is replaced, the power section definition has to be carried out before parameter files are downloaded.

If only parts of the entire parameter list are transferred by download, the parameters of the following table must always be transferred too, as these automatically result during the drive setting from the input of other parameters. During download, however, this automatic adjustment is not carried out.

Parameter number	Parameter name
P109	Pole pair number
P352	Reference frequency = $P353 \times P109 / 60$
P353	Reference frequency = $P352 \times 60 / P109$

Table 6-7 Parameters that always have to be loaded during download

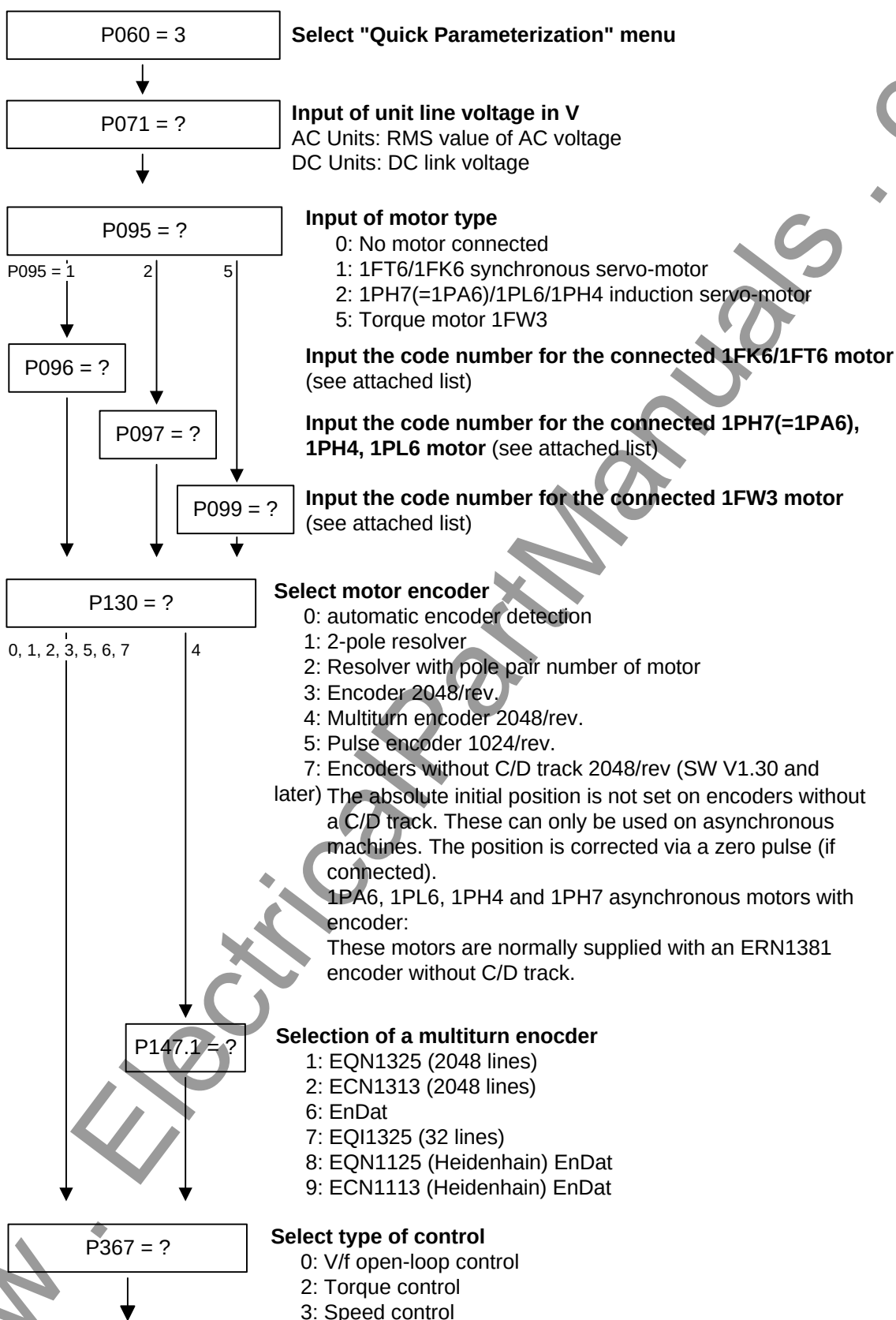
6.3.3 Parameterizing with parameter modules (quick parameterization, P060 = 3)

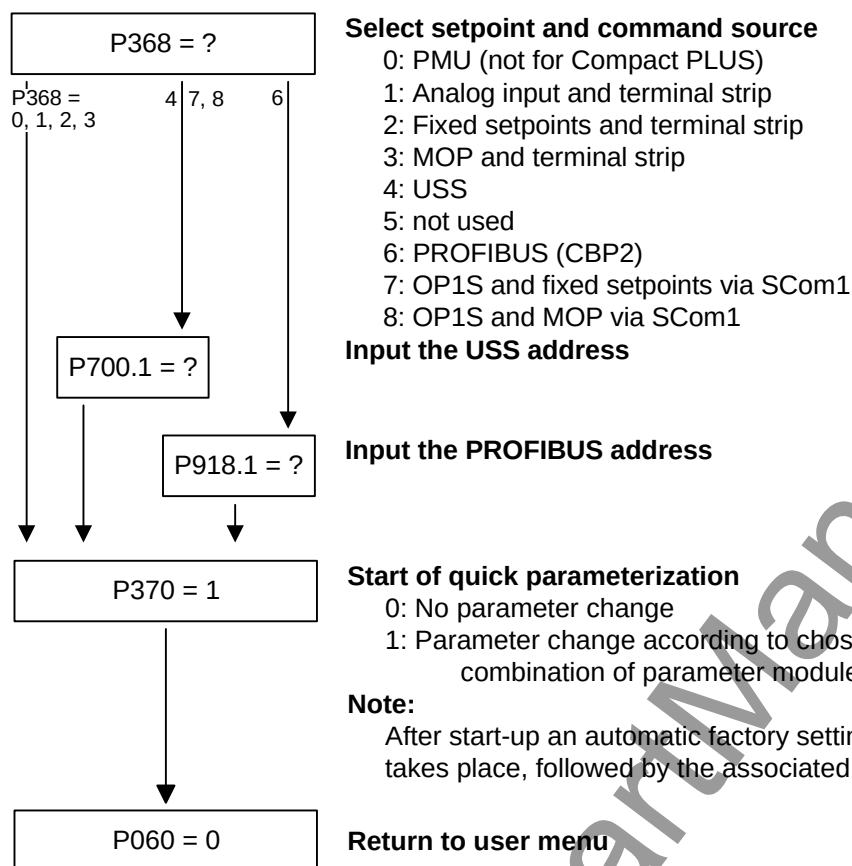
Pre-defined, function-assigned parameter modules are stored in the units. These parameter modules can be combined with each other, thus making it possible to adjust your unit to the desired application by just a few parameter steps. Detailed knowledge of the complete parameter set of the unit is not required.

Parameter modules are available for the following function groups:

1. Motors
2. Motor encoders
3. Control types
4. Setpoint and command sources

Parameterization is effected by selecting a parameter module from each function group and then starting quick parameterization. A parameter reset to the factory setting is performed and then, according to your selection, the required device parameters are set to achieve the required control functionality. The parameters necessary for fine adjustment of the control structure (all the parameters of the respective function diagrams) are automatically adopted in the user menu (P060 = 0).



**NOTE**

Parameterizing with parameter modules is carried out only in BICO data set 1 and in function data set 1.

If data set changeover is required, you will have to carry out a further detailed parameterization using parameter modules after having performed quick parameterization.

Quick parameterization is effected in the "Download" converter status.

Function diagram modules

Function diagram modules (function diagrams) are shown on the following pages for the parameter modules stored in the unit software. There is a module on every page beginning at the top for each of the following:

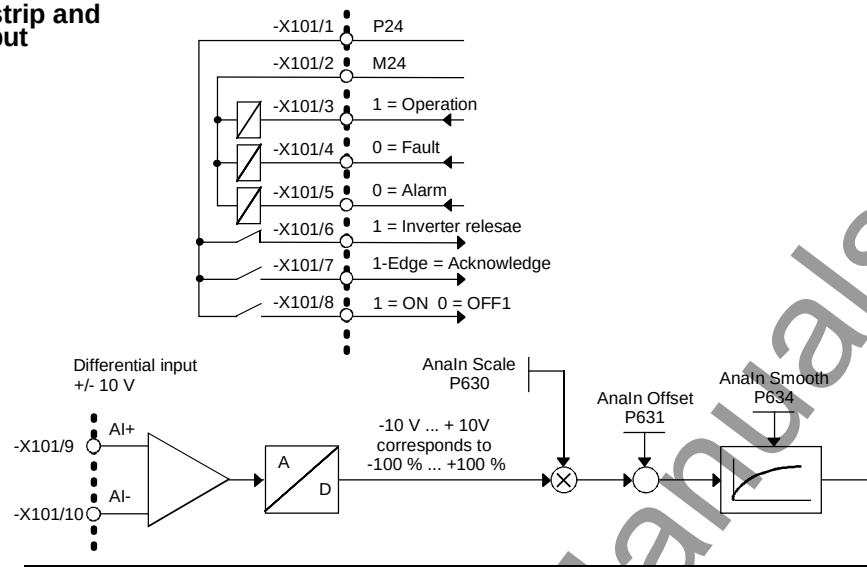
- ◆ Setpoint and command source,
- ◆ Control type and
- ◆ Motor encoder or the relevant sensor board (SBx)

Cutting lines have been marked in between the individual function diagram modules to enable the modules to be separated and then individually combined, i.e. you can create a function diagram to exactly fit the selected combination of parameter modules. This will give you an overview of the functionality parameterized in the units and of the necessary assignment of the terminals.

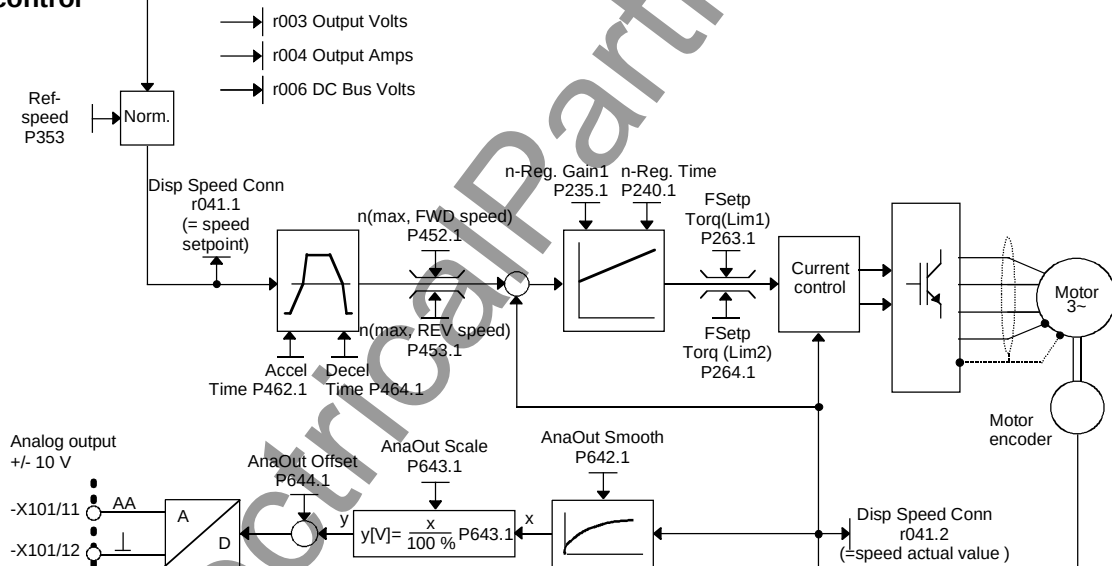
The function parameters and visualization parameters specified in the function diagrams are automatically adopted in the user menu and can be visualized or changed there.

www.ElectricalPartManuals.com

Setpoint and command source Terminal strip and analog input



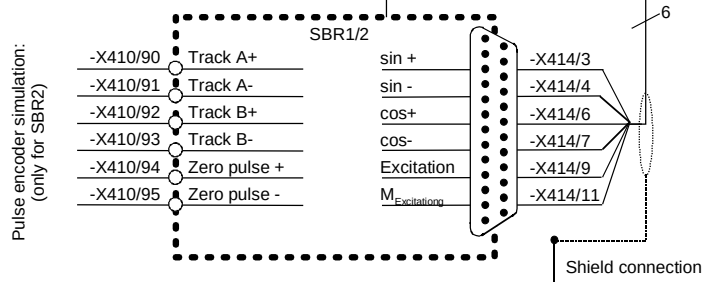
Control type: Speed control



Type of encoder: Resolver

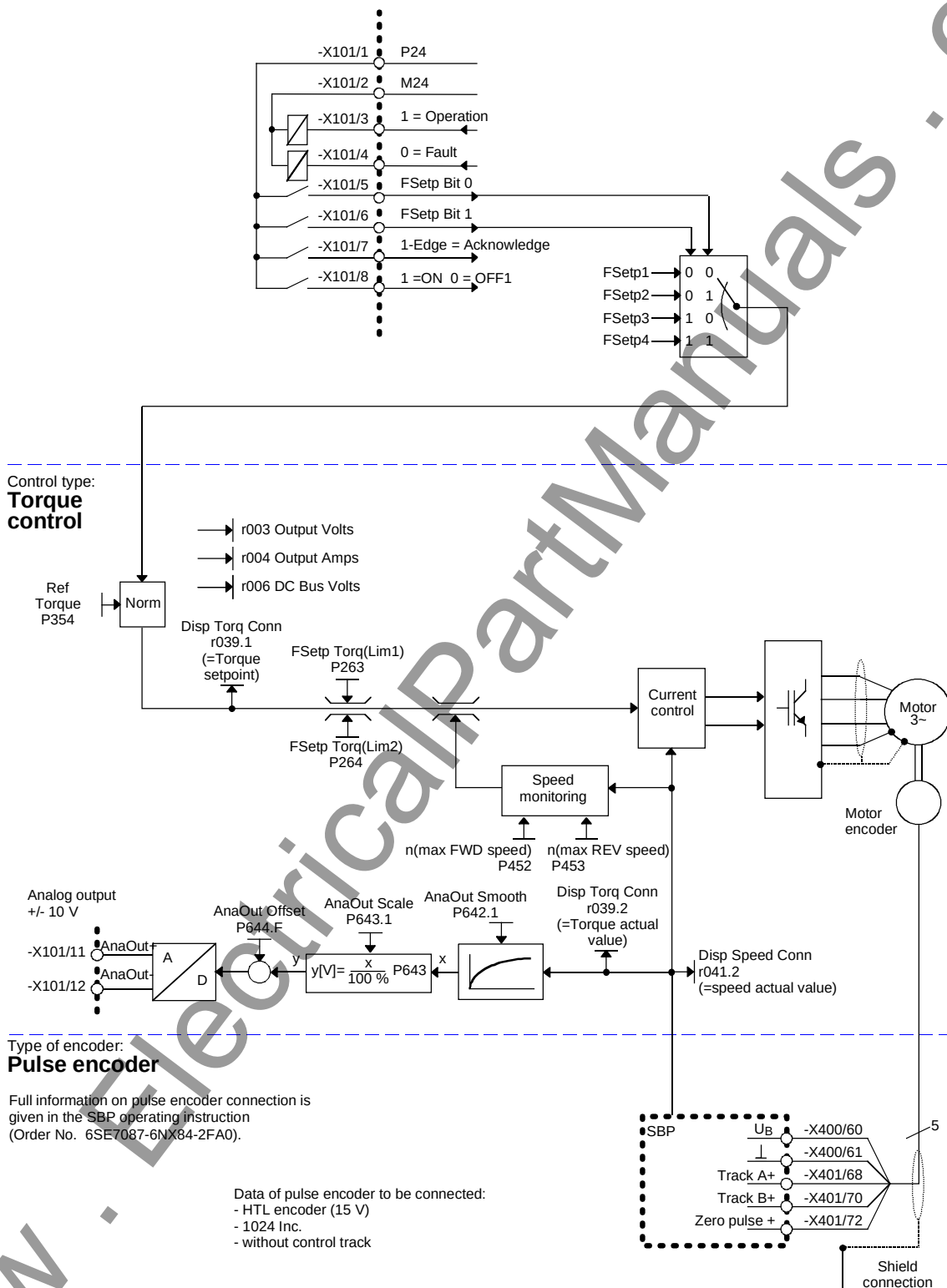
Data of resolver to be connected:
- 2-pole

Data of pulse encoder simulation:
- 1024 pulses/revolution



www.ElectricalPartManuals.com

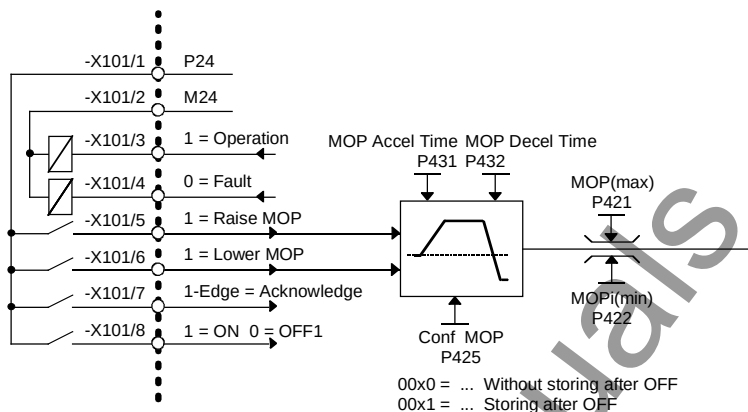
Setpoint and command source:

Terminal strip and fixed setpoints (FSetp)

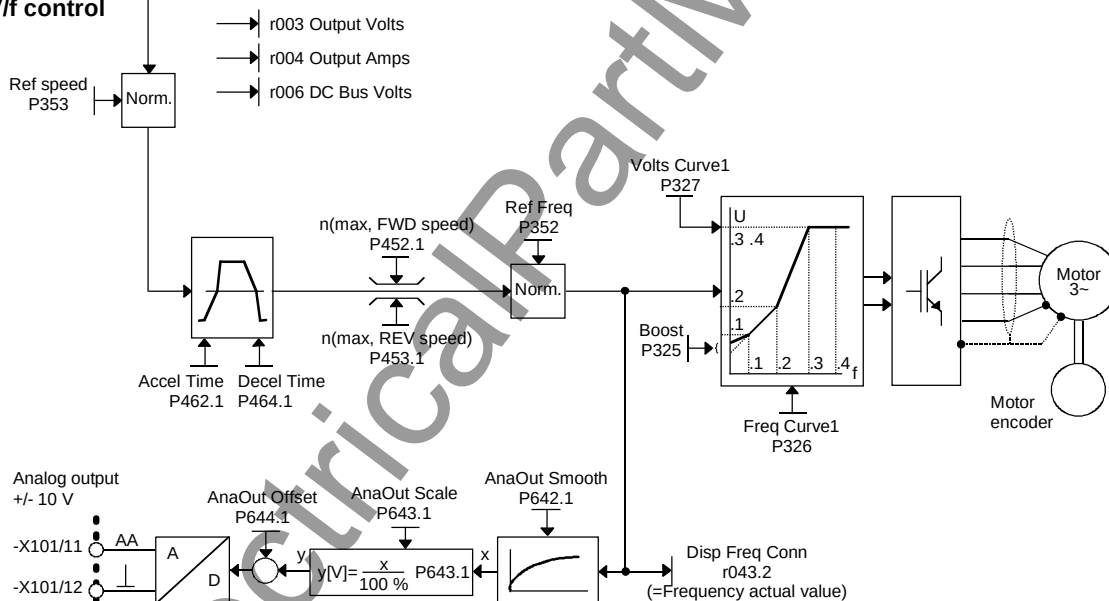
www.ElectricalPartManuals.com

Setpoint and command source

Terminal strip and motorized potentiometer



Type of control
V/f control



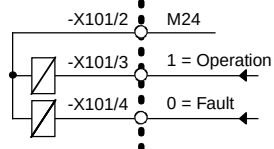
Type of encoder:

Without encoder

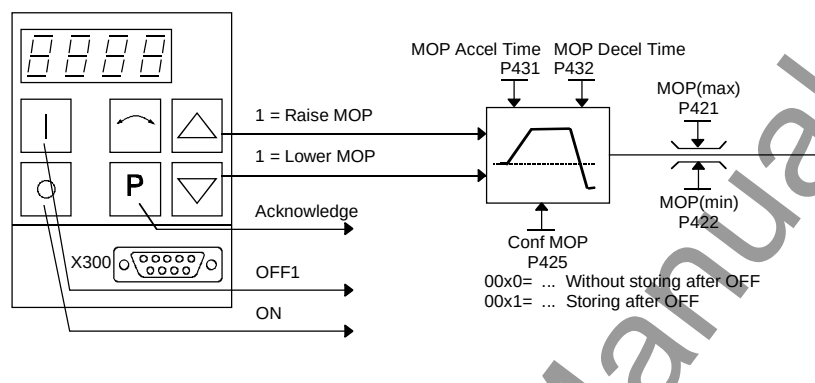
www.ElectricalPartManuals.com

Setpoint and command source

PMU (not for Compact PLUS units).



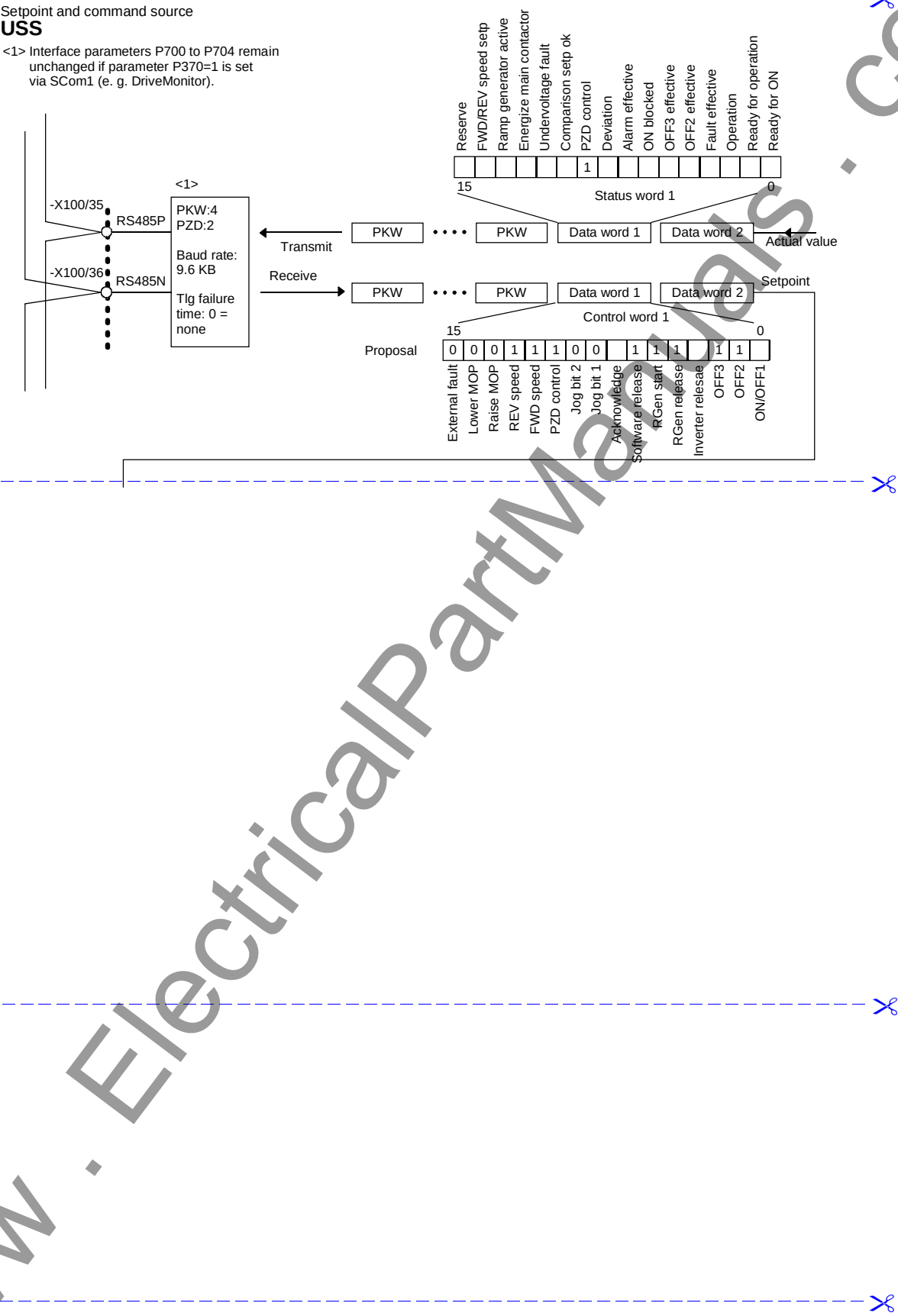
Note: The "Raise MOP" and "Lower MOP" keys are only effective if the operating display (r000) is selected.



www.ElectricalPartManuals.com

Setpoint and command source
USS

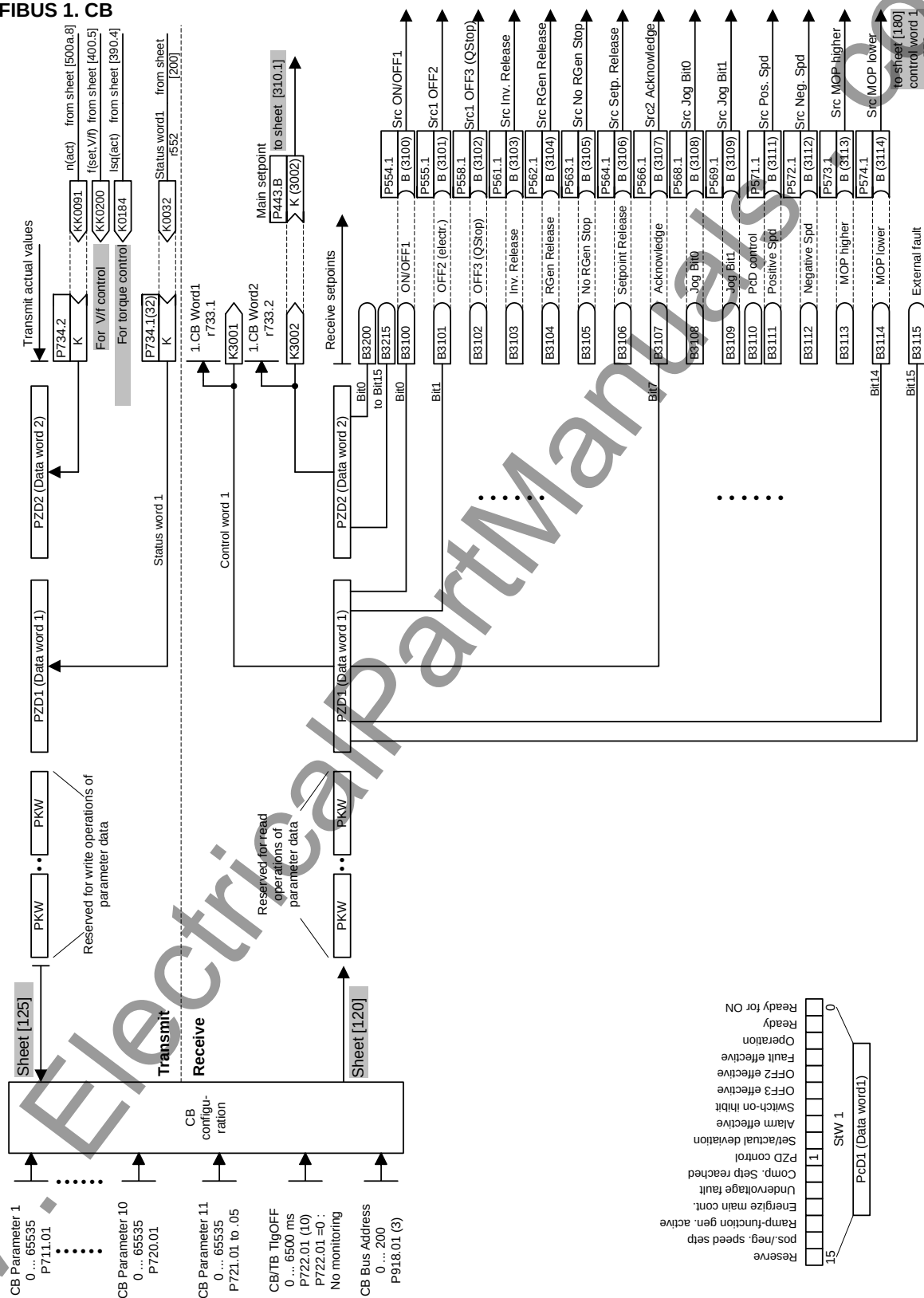
<1> Interface parameters P700 to P704 remain unchanged if parameter P370=1 is set via SCom1 (e. g. DriveMonitor).



www.ElectricalPartManuals.com

Setpoint and command source:

PROFIBUS 1, CB



www.ElectricalPartManuals.com

7 Functions

7.1 Basic functions

7.1.1 Time slots

The microprocessor system processes the function blocks sequentially. Each function block requires a certain calculating time and must be re-processed within a specified time. The microprocessor system therefore makes different times available to each individual function block. These times are designated as time slots.

A time slot is the period of time within which all output values of a function block are newly calculated.

NOTE

The following texts refer to the function diagram 702 "Setting and monitoring the sampling times and sampling sequence".

The terms "Time slot" and "Sampling time" have a synonymous meaning in the documentation and are interchangeable.

7.1.1.1 Time slots T0 to T20

T0 represents the shortest possible time slot within which a function block can be processed. The duration of time slot T0 is a function of the selected sampling frequency (P340), calculated as per the following formula:

$$T0 = \frac{1}{\text{Sampling frequency}}$$

This means:

- ◆ With a low sampling frequency (P340), the time slot T0 is longer. A large amount of calculating time is available for the individual function blocks. The reaction time is slower.
- ◆ With a high sampling frequency (P340), the time slot T0 is shorter. There is not much calculating time available for the individual function blocks. The reaction time is faster.

The time slot T0 forms the basis for all further time slots.

The time slots T1 to T10 and time slot T20 are available in addition to time slot T0. The time slots T1 to T10 are derived from time slot T0.

The time slot T20 is used as an archive for function blocks that are not needed. Function blocks stored in time slot T20 are not processed.

Overview of the time slots

Time slot	As a function of T0	Duration in ms at P340 = 5 kHz	Duration in ms as P340 = 7.5 kHz
T0	T0	0.2	0.133
T1	2 x T0	0.4	0.267
T2	4 x T0	0.8	0.533
T3	8 x T0	1.6	1.067
T4	16 x T0	3.2	2.133
T5	32 x T0	6.4	4.267
T6	64 x T0	12.8	8.533
T7	128 x T0	25.6	17.067
T8	256 x T0	51.2	34.133
T9	512 x T0	102.4	68.267
T10	1024 x T0	204.8	136.533
T20	none	archive	archive

7.1.1.2 Processing sequence

The time slots are processed in the sequence of their priority, whereby time slot T0 has the highest priority and time slot T10 the lowest priority. Each higher-priority time slot can interrupt a lower-priority time slot.

The sequence control of the converters and inverters starts every time slot automatically. If a higher-priority time slot is started, although another time slot is being processed, the time slot having the lower priority will be stopped and the time slot having the higher priority will then be processed before the interrupted time slot can be further processed.

Lower-priority time slots are lined up in a queue and are not processed until all higher-priority time slots are finished.

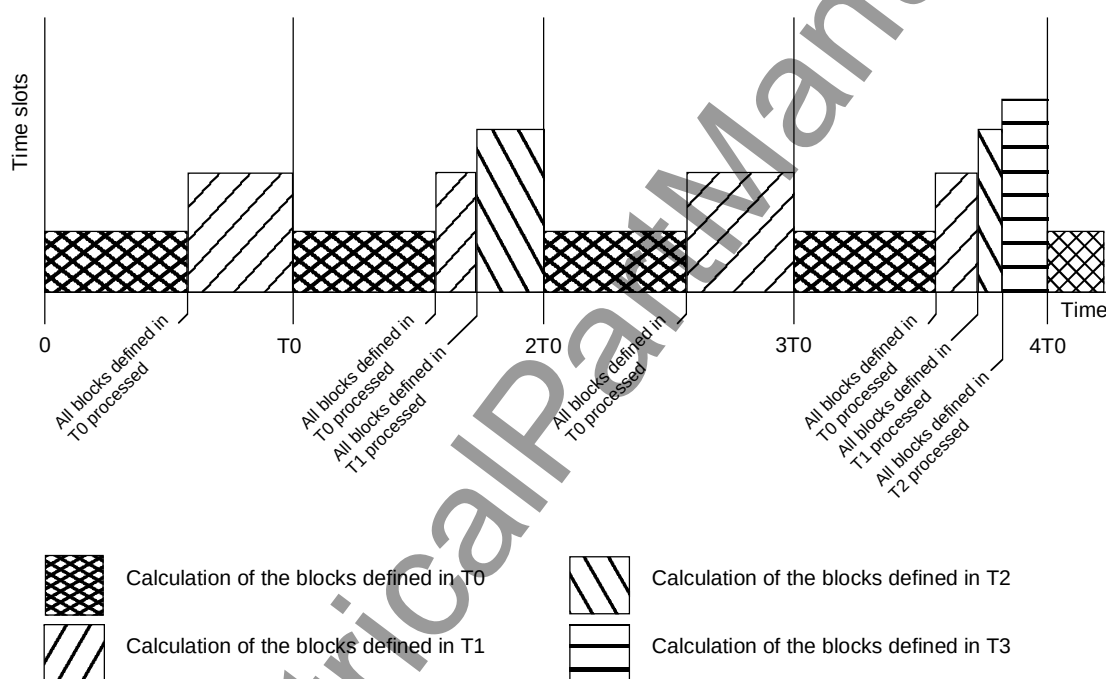


Fig. 7-1 Processing sequence of the time slots

7.1.1.3 Assignment of function blocks to time slots

To enable function blocks to carry out processing, a time slot (sampling time) is assigned to each function block. Assignment is effected by parameterizing in a table.

Some assignments are permanently stored in the firmware and cannot be changed. The two time slots T0 and T1 are reserved for time-critical function blocks. It is neither possible to remove function blocks from these time slots nor to assign further function blocks to these two time slots.

Time slot table

The time slot table consists of parameters U950 to U953. These parameters are indexed and have 100 indices each. Each index is assigned to precisely one function block. This means that the time slot in which the relevant function block is to be processed can be entered in the respective index.

The following applies regarding the assignment of the function block number to the parameter number with parameter index:

Parameter number	Parameter index	Assigned function block
U950	001	1
	...	...
	098	98
	099	99
U951	001	101
	...	...
	098	198
	099	199
U952	001	201
	...	...
	098	298
	099	299
U953	001	301
	...	...
	098	398
	099	399

Table 7-1 Time slots

The following assignment applies regarding the parameterization of time slots in parameters U950 to U953:

Time slot	Parameter value
T0	-
T1	-
T2	2
T3	3
T4	4
T5	5
T6	6
T7	7
T8	8
T9	9
T10	10
T20	20

Examples:

1. Function block 350 should be processed in time slot T4:
U953.50 = 4
2. Function block 390 should be processed in time slot T9:
U953.90 = 9
3. Function block 374 should not be processed:
U953.74 = 20

DANGER



When the units are delivered, time slots are already assigned to the function blocks. You should adjust these to suit your requirements once you have determined the interconnection of the function blocks.

Please note that an incorrect processing sequence will lead to uncontrolled axis movements!

7.1.2 Processing sequence of the function blocks

The function blocks are processed sequentially. For this reason, it is necessary to determine the processing sequence. The definition is carried out by means of parameterization in a table.

The processing sequence for some functions blocks is permanently stored in the firmware and cannot be changed. This concerns the function blocks that are defined in the two time slots T0 and T1.

Processing table

The processing table consists of parameters U960 to U963. These parameters are indexed and have 100 indices each. Each index is assigned to precisely one function block. This enables a processing number to be entered in the respective index for the function block concerned. Processing of the function blocks is then carried out in ascending order.

The following applies regarding the assignment of the function block number to the parameter number with parameter index:

Parameter number	Parameter index	Assigned function block
U960	001	1
	...	...
	098	98
	099	99
U961	001	101
	...	...
	098	198
	099	199
U962	001	201
	...	...
	098	298
	099	299
U963	001	301
	...	...
	098	398
	099	399

Examples:

1. Function block 350 should be processed in time slot T4 before function block 390:

U953.50 = 4
U953.90 = 4
U963.50 = 1000
U963.90 = 1010

2. Function block 350 should be processed in time slot T9 after function block 390:

U953.50 = 9
U953.90 = 9
U963.50 = 1050
U963.90 = 1010

DANGER



When the units are delivered, a processing sequence has already been determined. You should adjust this to suit your requirements once you have determined the interconnection of the function blocks.

Please note that an incorrect processing sequence will lead to uncontrolled axis movements!

7.1.2.1 Time monitoring

Depending on the number and frequency of the blocks to be processed, the microprocessor system of the units has a varying degree of utilization. In order to avoid any dangerous overloading, the operating system has a time monitoring facility, which

- ◆ monitors the system for its overall workload
- ◆ monitors the various time slots to ensure they are being completely processed within the allocated time
- ◆ generates alarms and fault messages if calculating times are not adequate.

7.1.2.2 Influencing the time response

The time response affects two different areas:

- ◆ Calculation workload
- ◆ Control response

Calculation workload

You can influence the calculation workload as follows:

- ◆ By changing the sampling frequency P340. With a high sampling frequency, less calculating time is available per time slot. With a low sampling frequency, more calculating time is available per time slot.
- ◆ By assigning function blocks to different time slots.

If you assign too many function blocks to one time slot, it is no longer possible to process all function blocks within the specified time. The time monitoring facility generates an alarm and de-energizes the unit if the alarm occurs repeatedly.

Control response

You can influence the control response as follows:

- ◆ By changing the sampling frequency P340. With a high sampling frequency, the reaction time is shorter. With a low sampling frequency, the reaction time is longer.
- ◆ Change parameter P357. Setting parameter P357 to 1 will also impair the reaction time as current control only takes place in time slot T1 which corresponds to the sampling frequency P340 being halved.
- ◆ By assigning function blocks to other time slots.
- ◆ By changing the processing sequence.
- ◆ By changing time-relevant parameters.

If you assign a function block to a slow time slot (e.g. T10), the result of this function block is seldom re-calculated, i.e. the long processing time acts on the control circuit as a delay element. If you change the processing sequence of two consecutive function blocks by having one output block calculated before its associated input block, you will have integrated a delay element of the duration of one time slot into the control circuit.

Rules

You should observe the following rules regarding the assignment of function blocks to the time slot table and the processing table.

- ◆ Function blocks that can be combined to form a function group (with a mutual task) should be processed in the same time slot.
- ◆ Function blocks should be processed in the fastest necessary time slot, not in the fastest possible time slot.
- ◆ The sequence in which the function blocks are entered into the processing table should correspond to the signal flow.

NOTE

With the introduction of parameter P357<>0 (from V2.20) the pulse frequency is no longer directly connected with the sampling frequency.

Changing the pulse frequency by halving it (P357 = 1) is only possible if the Power Extension PIN F02 has been enabled (relevant for power sections above 75 kW), see Chapter 11.10 "Power Extension PIN F02".

7.2 Technology functions

DANGER



It is the user's responsibility to make allowance for the safety-relevant requirements when using the technology functions as free blocks and to interlock them.

7.2.1 Comfort ramp-function generator

Adaptation

- ◆ The adaptation function has no effect with ramp-up/ramp-down times in 'min' and 'h'.
- ◆ The resolution is 11 bits = 0.2 %.
- ◆ The adaptation function only has an effect for ramp-up/ramp-down times, not for rounding off.
- ◆ With an adaptation factor of 0 %, the rounding time at least has an effect.

Rounding

- ◆ The rounding function has no effect with ramp-up/ramp-down times in 'min' and 'h'.
- ◆ The rounding function also has an effect during zero passage.
- ◆ There is no overshooting over zero.
- ◆ The rounding function can be altered during ramp-up/ramp-down.
- ◆ Initial rounding is always limited to at least 10 % of the final rounding.

Calculating time

The following calculating times apply to the comfort ramp-function generator:

- ◆ Without rounding:
65 to 79 µsec
- ◆ With initial rounding = final rounding:
96 to 105 µsec
- ◆ With initial rounding <> final rounding:
105 to 114 µsec
- ◆ With initial rounding <> final rounding and adaptation:
123 to 132 µsec

Sampling time

The ratio of the sampling time to the ramp-up, ramp-down and rounding times is as follows:

- ◆ With 1 : 100, a very good result
- ◆ Example.: When $T_{sa} : T_{round} = 1 : 10$, the ramp-up/ramp-down time can be false by a maximum of 10 %
- ◆ Max. sampling time: 200.00 msec

Priorities

The priorities of the commands of the comfort ramp-function generator are as follows:

1. Enable
2. Quick stop
3. Set
4. Ramp down to zero
5. Stop

RFG tracking The ramp-function generator tracking (limiting) always has an effect, i.e. even if the RFG is blocked. The positive limiting value should always be greater than the negative limiting value, otherwise the limitation cannot be correctly calculated. If the positive limiting value is negative, the output is limited to this negative value even if the RFG has been blocked!

RFG bridging Bridging of the ramp-function generator has the following effect:

- ◆ The output value y is equal to the input value x , irrespective of the commands ramp down and stop.
- ◆ In the case of quick stop, however, the quick-stop time has an effect.

7.2.2 Technology controller

Calculating time The following calculating times apply to the technology controller:

- ◆ PI controller with K_p adaptation: 38 μsec
- ◆ PID controller with K_p adaptation and smoothing: 48 μsec
- ◆ With all (droop, precontrol, RFG at output): 58 μsec

K_p adaptation Due to negative factor, a sign reversal is possible!

I component

- ◆ The I component is always set, irrespective of whether the controller is blocked or enabled.
- ◆ If the controller is blocked, the controller output is always zero, even when the I component is set
- ◆ When $T_n = 0$, the I component is deleted, the effect is equal to $T_n = \infty$
- ◆ Tracking of the I component only occurs when the output is limited ($B0555 = 1$) and $T_n \neq 0$ and the controller is enabled and the I component is not set.

Controller block Blocking of the technology controller results in the following:

- ◆ The limiting ramp-function generators are reset
- ◆ The setpoint, actual value, smoothing and controller input are calculated
- ◆ The D component is calculated
- ◆ Droop and precontrol are added
- ◆ P component and controller output are zero
- ◆ No setting of the I component $\Rightarrow$ I component is deleted
- ◆ The output limitation is calculated (with $B+ = B- = 0$)

Smoothing

- ◆ No sub-sampling, i.e. each value is used
- ◆ With a smoothing time of zero, the smoothing element is set to the input value
- ◆ Smoothing time : scan time < 500
 $\Rightarrow$ max. 1 % error in smoothing time
- ◆ Smoothing time : scan time = 3000
 $\Rightarrow$ max. 10 % error in smoothing time
- ◆ Smoothing time : scan time > 20000
 $\Rightarrow$ should not be set

RFG output limitation

- ◆ Ramp time : scan time < 500
=> max. 1 % error in ramp time
- ◆ Ramp time: scan time = 1500
=> max. 10 % error in ramp time
- ◆ Ramp time: scan time > 10000
=> should not be set
- ◆ Always B- ≤ B+
- ◆ The upper limit (U370.1) always has higher priority than the lower limit (U370.2)

7.2.3 Basic positioning

Principle

The basic positioner can be used for "basic" positioning tasks. It is implemented using three free function blocks and provides the necessary operating modes and functionality to move an axis from A to B under position control.

As can be seen in the "overview" in function diagram 788a, the basic positioner consists of the three free function blocks that are completely pre-wired together in the factory setting for the "Basic positioner with motor encoder" function (these three blocks can also be used individually for further applications).

All that therefore has to be done is to modify the desired inputs in function diagram 788a (shown in more detail in function diagram 789a) and connect the outputs (see function diagrams 788a and 789c) as recommended (for motor encoder).

Enabling (ENABLE POS/ REF) is implemented in the factory setting using the "position controller enabled" checkback signal, i.e. best way to enable the basic positioner is by means of the user-selectable source "enable position controller" (P210, [340.4]).

Function diagram 788 also provides a graphical overview of the embedding.

As practically all the variables of the basic positioner (including those between the three free function blocks) are binector or connector inputs/outputs, the function can be controlled both using only one signal or with the desired sequence/interlocking.

Consequently, the desired function/movement must be ensured by the user.

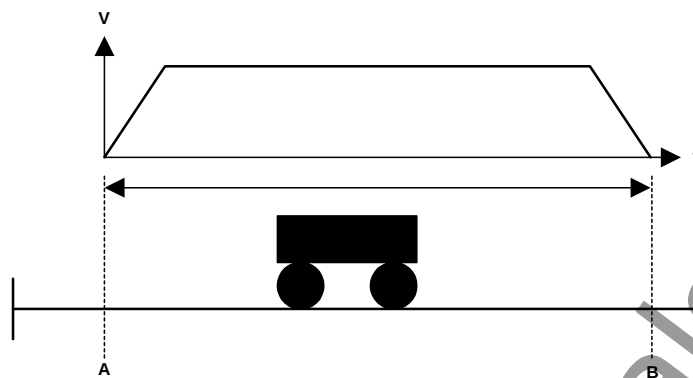


Fig. 7-2 Moving from A to B

NOTE

Standard applications are available including parameter assignment and documentation. These can be obtained from your regional SIEMENS AG office and are available from the Application Center for Production Machines.

Characteristics**What the basic positioner offers:**

Positioning POS (absolute/relative) of linear and circular axes with motor encoder or machine encoder

- ◆ SETUP (stepless)
- ◆ Homing REF (homing movement/homing on the fly)
- ◆ Software limit switch SWE (for linear axis, connector inputs from V2.3)
- ◆ Stop cam (from V2.3)
- ◆ Static STOP cam (from V2.4)
- ◆ Backlash compensation (with preferred position)
- ◆ Exact positioning window (POS_OK window + delay time)
- ◆ Window evaluation for post-homing
- ◆ Rate-of-change limitation (adaptable)
- ◆ Continuous or triggered setpoint transfer
- ◆ Mode change on the fly (REF, POS, SETUP)
- ◆ Setpoint change on the fly possible using PZD

Advantages of basic positioner:

- ◆ Cost neutral (with basic unit functionality)
- ◆ Easy to understand (basic commissioning)
- ◆ Continuous setpoint evaluation (during constant transfer)
- ◆ Control/checkback interface using BICO technology (e.g. PLC connection)
- ◆ Mode change on the fly (REF, POS, SETUP)
- ◆ Lower calculation time loading
- ◆ Lower project engineering costs
- ◆ Greater freedom for applications
- ◆ SIMATIC S7 not absolutely necessary

Differences from existing technology option F01:

- ◆ No automatic processing of blocks
- ◆ No automatic lag monitoring (this can be implemented using free blocks if required, see Section "Script files with project examples")
- ◆ No fixed error messages or warnings (these can be implemented using free blocks if required)
- ◆ No remaining traverse path in the case of relative positioning (in the sense of F01)

The basic positioner, referred to in the subsequent text as positioner or by the abbreviation BPos (in the parameters), is implemented using three free function blocks. The factory settings of these all match and the function blocks are already wired ready for operation with a motor encoder. The user simply has to make the connections to the basic unit (see function diagram 789c).

Overview of the three function blocks**Setpoint transfer and mode management [FD789a]**

Setpoint transfer block with mode management and edge-controlled setpoint transfer for consistent data transfer.

Setup/positioning [FD789b]

Setup/positioning block that traverses a specified path relatively or absolutely using the specified deceleration, acceleration and speed.

Correction value/homing [FD789c]

Correction block that provides the position correction and position setpoints for linking to the position controller and also the position detection (see overview diagram FD788, FD788a for linking to the basic unit).

Recommendations

Function diagrams 788 to 789c are necessary for an understanding of these instructions.

For the standard user, it is sufficient to work with function diagram 788a and to use these instructions for reference if required.

NOTE

The control/checkback signals have **positive** logic (up to RESET SET setpoint).

Application

The basic positioner consists of the three free blocks and is available at no extra cost with the same functionality as the basic unit to provide a solution for **"basic" positioning applications**.

(In the factory setting, the three free blocks are **completely** pre-wired for the "basic positioner with motor encoder" application.)

The positioner leaves all options open to the user when designing a solution for a positioning task. Thus, the positioner can be expanded or modified by its own BICO interconnections from the blocks available in the unit.

DANGER



The user is responsible for taking account of and providing interlocks for the relevant safety requirements when implementing a positioning task using the basic positioner.

NOTE

The positioner only operates as a pure position controller. The positioner is enabled using B0220 on U866.1 ENABLE_POS_REF (enable basic positioner).

It is expected that parameter assignment of the basic unit has already taken place before the commissioning of the positioner.

Process data traffic is not linked to predefined jobs (e.g. from an S7), but can be implemented using the PKW/PZD basic unit functions (see project examples).

Using the communication options of the basic unit (SIMOLINK, USS-BUS, PROFIBUS-DP, etc.), the position setpoint, maximum speed, acceleration and deceleration and the control word can be processed directly as setpoints and the actual values and statuses can be read back.

The **setup/positioning** block is a setpoint generator. The position and speed setpoints required for a positioning process are formed from the target position, the maximum speed and the maximum acceleration or deceleration.

These parameters are used to calculate the speed and position setpoints such that the target position is reached without violating the specified limits.

The setpoint generator can also be used as a pure ramp function generator and setpoint generator for a control process (setup) or as lead value source for synchronous tasks.

The **setpoint transfer block** carries out the setpoint transfer and interlocks the possible modes of homing, positioning and setup. This also generates a homing movement that processes the core block as a ramp generator (SETUP) using reversing cams.

The **correction block/homing command** provides the setpoints with speed precontrol, the position correction signals and handles the measured value memory.

The blocks must always be configured so they have the shortest possible interrogation time (< 5 ms). If too slow an interrogation time is chosen, setpoint jumps or uneven running of the axis may occur.

Preferably, the interrogation time should be the same as that used for technology option F01 (interrogation time T4).

U953.60 = 4

U953.61 = 4

U953.62 = 4

7.2.3.1 Functions

Set setpoint transfer Setpoint transfer and mode management (function diagram 789a)

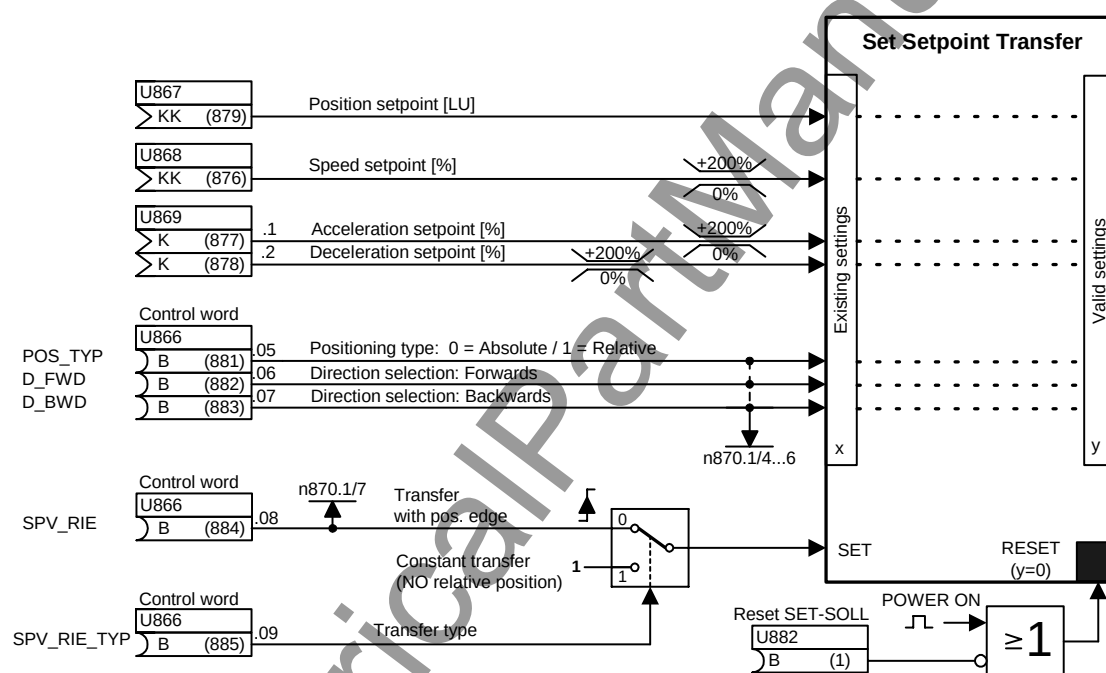


Fig. 7-3 Setpoint transfer input

The effective setpoints, such as position setpoint [LU], speed [%], acceleration [%] and deceleration [%], and the binary signals for type of positioning (absolute or relative) and for direction selection (forwards or backwards) all form part of the "set setpoint" transfer.

NOTE

The speed, acceleration and deceleration setpoints are percentage values that must always be positive (negative setpoints are limited to 0 %).

Set setpoint transfer with edge control [SPV_RIE_TYP] = 0

If edge-controlled transfer is selected, the active setpoint values are always valid at the same time as the rising edge on SPV_RIE from 0 → 1.

After transfer of the setting values, the SPV_RIE_ACKN transfer acknowledgement is set as checkback for the user. An acknowledgement-controlled setpoint transfer can be implemented with the checkback signal SPV_RIE_ACKN (see section "Acknowledge transfer").

Set/setpoint transfer with constant transfer [SPV_RIE_TYP] = 1

If constant setpoint transfer is selected, all the pending setting values are transferred immediately as valid values. A rising edge on SPV_RIE 0 → 1 no longer has any effect on this. The checkback signal SPV_RIE_ACKN has no function in this mode of setpoint transfer and therefore remains at logical zero.

NOTE

The following limitation applies to constant setpoint transfer:

The relative positioning type (POS_TYP = 1) is locked in the case of constant setpoint transfer (SPV_RIE_TYP = 1).

The valid speed setpoint is set to 0%.

Therefore, **no relative positioning** is possible with constant setpoint transfer.

Reason:

For a relative travel movement, the travel path is started as an incremental dimension with a rising edge on POS_ON. Therefore, only an edge-controlled transfer makes sense.

Acknowledge transfer [SPV_RIE_ACKN]

With edge-controlled transfer, the signal [SPV_RIE_ACKN] is set as checkback for the user after transfer of the setting values. If the control signals and setpoints are transferred from a higher-level system (e.g. PC, SIMATIC S7) in conjunction with the communication capabilities of the basic unit, the acknowledge transfer can be employed by the user to initiate the [SPV_RIE] signal under acknowledgement control.

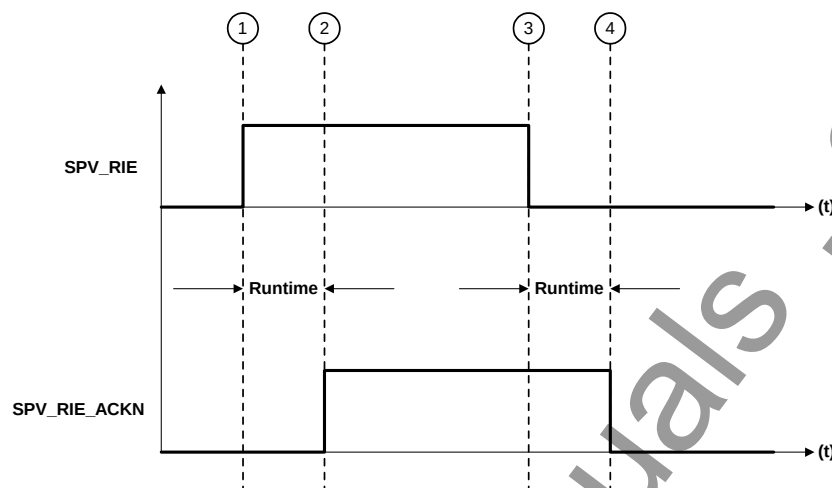


Fig. 7-4 Signal sequence of acknowledgement-controlled setpoint transfer

Description:

- ① The block transfers the pending setting values from the user when a rising edge is detected on the SPV_RIE signal (from 0 to 1).
- ② Checkback signal SPV_RIE_ACKN = 1
Acknowledges the transfer of the setpoints from the positioner in the form of a checkback signal for the user
- ③ With control signal SPV_RIE = 0
acknowledge transfer is confirmed by the user
- ④ Checkback signal SPV_RIE_ACKN = 0
as conclusion of setpoint transfer

Reset set setpoint (U882)

This input is low active and is connected to logical 1 in the factory setting as an auxiliary input. All outputs of the basic positioner (i.e. binectors and connectors) of the set/setpoint transfer are set to zero with logical "0" (KK0874, KK0875, K0872, K0873, B0874, B0875, B0876 = 0) as is the case after Power ON.

This gives the user the option to delete the valid outputs.

DANGER



An abrupt compensating movement of the axis may occur if the RESET signal is output without precautionary measures having been taken.

Mode management

The mode management and setpoint transfer block can be found on function diagram 789a and is assigned to a time slot by means of U953.60.

The mode management function block interlocks the homing, positioning and setup modes against each other. This interlocking ensures the priority of the modes.

Priorities:

REF_ON	→ homing	= highest priority
POS_ON	→ positioning	= second highest priority
SETUP_ON	→ setting up	= lowest priority

Transfer between the modes takes place on the fly. A mode change can be carried out without a drop in speed. The priorities are always taken into account, even if modes are selected simultaneously.

Example:

If all modes are selected simultaneously REF_ON = 1 with REF_TYP = 1, POS_ON = 1, SETUP_ON = 1, homing movement always has priority. If homing movement is deselected (REF_ON = 0 with REF_TYPE = 1), the positioning mode is effective.

NOTE

Homing on the fly (REF_TYPE = 0) is always effective with REF_ON and has no effect on the priority.

If positioning is also deselected, the setup mode immediately becomes active.

Mode management also safeguards the sequence control of the homing movement, i.e. homing movement with preferred direction evaluation and reversing cams.

This mode permits the positioning block to travel to and fro in SETUP = 1 using the reversing cams REF_BWD_STOP, REF_FWD_STOP until REF_STOP [ARFD] = 1 is achieved.

See "Homing movement" section for further information.

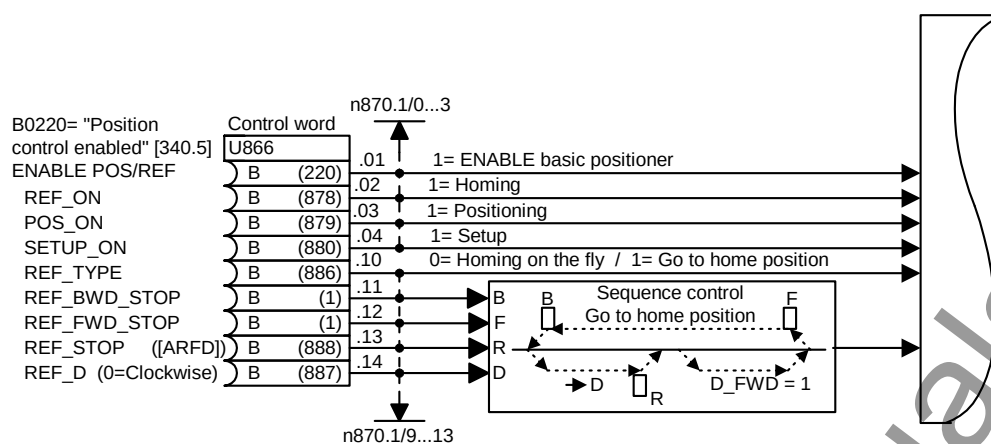


Fig. 7-5 Mode management input on control word U866

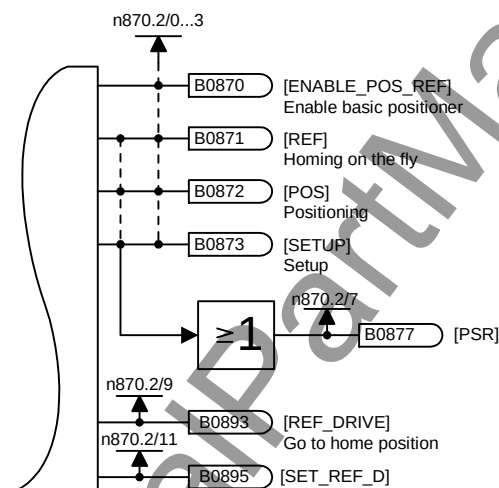


Fig. 7-6 Mode management outputs

The mode management outputs become the control signals for the position ramp function generator and the correction block.

Binector B0870 ENABLE_POS_REF = 1 enables the series-connected function blocks (positioning block, correction block).

Binector B0877 PSR (**POS/SETUP/REF**) shows that one of the three modes POS, REF or SETUP is active.

Binector B0893 REF_DRIVE = 1 shows that "homing movement" with reversing cams is active.

Binector B0895 SET_REF_D = 1 shows the preferred direction REF_D (input Src U866.14).

Homing on the fly		Input (control signals)										Output	
		ENABLE	REF_ON	POS_ON	SETUP_ON	REF_TYP	REF_STOP	REF_D	D_FWD	D_BWD	POS_TYP		
REF_TYP = 0													
ENABLE_POS_REF	B0870	1	x	x	x	x	x	x	x	x	x	1	
REF	B0871	1	1	x	x	0	x	1	x	x	x	1	
POS	B0872	1	x	1	x	0	x	x	x	x	x	1	
SETUP	B0873	1	x	0	1	0	x	x	0	1	x	1	
D_FWD_ACT	B0875	1	x	0	1	0	x	1	1	0	x	1	
D_BWD_ACT	B0876	1	x	1	0	0	x	0	0	1	x	1	
PSR	B0877	1	1	1	1	x	x	x	x	x	x	1	
REF_DRIVE	B0893	1	x	x	x	0	x	x	x	x	x	0	
POS_TYP_ACT	B0874	1	0	1	x	0	x	x	x	x	0	0	
		1	0	1	x	0	x	x	x	x	1	1(V=0)	
SET_REF_D	B0895	x	x	x	x	x	x	1	x	x	x	1	
Homing													
REF_TYP = 1													
ENABLE_POS_REF	B0870	1	x	x	x	x	x	x	x	x	x	1	
REF	B0871	1	1	x	x	1	0	1	x	x	x	1	
POS	B0872	1	1	x	x	1	x	x	x	x	x	0	
SETUP	B0873	1	1	x	x	1	0	x	0	1	x	1	
D_FWD_ACT	B0875	1	1	x	x	1	0	1	1	0	x	1	
D_BWD_ACT	B0876	1	1	x	x	1	0	0	0	1	x	1	
PSR	B0877	1	1	x	x	x	0	x	x	x	x	1	
REF_DRIVE	B0893	1	1	x	x	1	0	x	(1)*	(1)*	x	1	
POS_TYP_ACT	B0874	1	0	1	x	1	x	x	x	x	0	0	
		1	0	1	x	1	x	x	x	x	1	1(V=0)	
SET_REF_D	B0895	x	x	x	x	x	x	1	x	x	x	1	

x= don't care

*) Depending on selection of direction D_FWD_ACT / D_BWD_ACT

Table 7-2 Mode truth table

Status signals n870 Status signals of monitoring parameter n870
n870 Index 1: setpoint/mode input (-> K0886)

Bit 0	U866.1	ENABLE_POS/REF	Enable basic positioner
Bit 1	U866.2	REF_ON	Homing on
Bit 2	U866.3	POS_ON	Positioning on
Bit 3	U866.4	SETUP_ON	Setup on
Bit 4	U866.5	POS_TYP	Positioning mode
Bit 5	U866.6	D_FWD	Positive direction
Bit 6	U866.7	D_BWD	Negative direction
Bit 7	U866.8	SPV_RIE	Transfer with positive edge
Bit 8	U866.9	SPV_RIE_TYP	Transfer type
Bit 9	U866.10	REF_TYP	Transfer type
Bit 10	U866.11	REF_BWD_STOP	Positive direction reversing cam
Bit 11	U866.12	REF_FWD_STOP	Negative direction reversing cam
Bit 12	U866.13	REF_STOP	Terminate homing
Bit 13	U866.14	REF_D	Homing approach direction

Table 7-3 BPos STW status of monitoring parameter n870.1: setpoint/mode input

The BPos STW status input is the same as connector 0886.

n870 Index 2: setpoint/mode output (->K0887)

Bit 0	B0870	ENABLE_POS_REF	Enable basic positioner
Bit 1	B0871	REF	Homing on the fly
Bit 2	B0872	POS	Positioning
Bit 3	B0873	SETUP	Setting up
Bit 4	B0874	POS_TYPE_ACT	Valid positioning type
Bit 5	B0875	D_FWD_ACT	Valid positive direction
Bit 6	B0876	D_BWD_ACT	Valid negative direction
Bit 7	B0877	PSR	POS or SETUP or REF active
Bit 8		-----	
Bit 9	B0893	REF_DRIVE	Homing active
Bit 10	B0894	SPV_RIE_ACKN	Transfer acknowledge
Bit 11	B0895	SET_REF_D	Homing approach direction

Table 7-4 BPos STW status of monitoring parameter n870.2: setpoint/mode output

The BPos STW status output is the same as connector 0887.

Special case when selecting direction D_FWD and D_BWD

Case 1: A rotary axis (U858.1 $\neq$ 0) is positioned absolutely (POS_TYP = 0).

In this case, in addition to the mode, the direction of movement must be determined through the two control binector inputs D_FWD or D_BWD:

[D_FWD]	[D_BWD]	
0	0	= shortest path
1	0	= always positive direction
0	1	= always negative direction
1	1	= no selection of direction of movement (or: axis is stopped on ramp)

Case 2: The homing mode is selected with a circular or linear axis.

In this case, in addition to the mode, the direction of movement must be determined through the two control binector inputs D_FWD or D_BWD:

[D_FWD]	[D_BWD]	
0	0	= no selection of direction of movement (or: axis is stopped on ramp)
1	0	= positive direction
0	1	= negative direction
1	1	= no selection of direction of movement (or: axis is stopped on ramp)

Case 3: A linear axis (U858.1 $\neq$ 0) is positioned absolutely (POS_TYP = 0) or relatively (POS_TYP = 1).

In the case of relative positioning, the sign of the position setpoint determines the direction of movement. With absolute positioning, the direction of movement is determined by the difference between the position setpoint and actual values. Activation of the control binector inputs D_FWD or D_BWD has **no** effect.

7.2.3.2 Normalization

The purpose of normalization is to establish the relationship between the mechanical (e.g. mm) and electronic (LU) representation.

In positioning, the path dimension unit is called a **LENGTH UNIT LU**. This means that LU can be mm, inches, degrees, etc.

LU = **LENGTH UNIT** is a neutral length dimension.

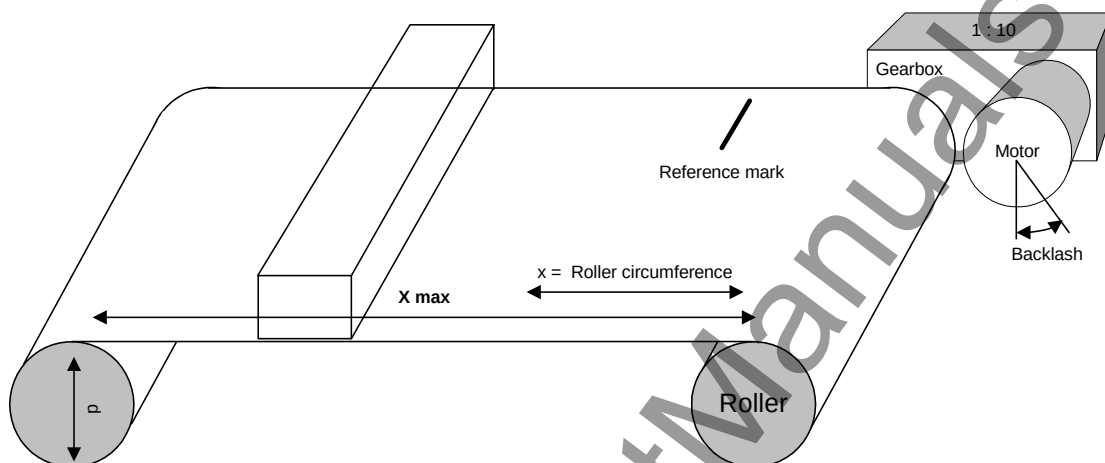


Fig. 7-7 Typical positioning application

The arrangement shown in Fig. 7-7.

NOTE

The factory setting values are pure motor-related normalizations.

Example:

Encoder: 131 072 pulses per motor revolution [resolution in 2^n ($n = 17$)]

with a gearbox having 1:10.00 ratio (i), this gearbox factor must be included in the calculation.

$131072 \times 10 = 1\,310\,720$ pulses per roller revolution

The diameter of the drive roller (d) is 300.000 mm

$$x = d \times \pi = 300 \text{ mm} \times 3.1415 = 942.477 \text{ mm}$$

The following is to be represented: **1LU = 1 μ m**

The circumference of the roller (x) is therefore 942477 LU.



Actual-value weighting factor

AVWF = actual-value weighting factor as conversion constant/factor.

For example, if 1 LU = 1 µm, a decimal setpoint of 1000 LU represents a travel distance of 1000 µm, or 1 mm.

Without the AVWF factor, the length information always refers to the encoder pulses depending on the resolution of parameter P171 (motor encoder) 2^{P171}.

e.g. P171 = 12 → 4096 LU/revolution

The AVWF is calculated from this as follows:

$$\text{AVWF} = \frac{\text{Path distance per motor revolution}}{\text{LU per motor revolution}} \quad \text{or}$$

$$\text{AVWF} = \frac{\text{Roller circumference}}{\text{LU per motor revolution} \times \text{gearbox factor (i)}}$$

The aim should be to have a resolution of about 1 µm / encoder increment. Table 7-5 shows how large the factors are and hence how high a resolution is to be selected.

In the example, the AVWF factors result from a roller diameter of 300.00 mm (circumference = 942477 µm). This must then be multiplied by the gearbox factor (AVWF*).

P171	Resolution	AVWF	AVWF*
12	4096	230.096924	23.0096924
13	8192	115.048462	11.5048462
14	16384	57.524231	5.7524231
15	32768	28.7621155	2.87621155
16	65536	14.3810577	1.43810577
17	131072	7.19052887	0.719052887
18	262144	3.59526443	0.359526443
19	524288	1.79763222	0.179763222
20	1048576	0.89881611	0.089881611
21	2097152	0.44940805	0.044940805

Table 7-5 AVWF factors resulting from a roller diameter of 942477 µm

This results in an AVWF factor (which should be less than 1) of **0.71905288** or, as a fraction

$$\frac{\text{Numerator}}{\text{Denominator}} = \frac{942477}{1310720} = 0.71905288$$

as gearbox factor or factor for the AVWF.

More encoder increments per revolution should therefore be specified (P171) than there are LUs per revolution.



NOTE

The AVWF, which is entered in P169 for the figures before the decimal point and in P170 for the figures after the decimal point, can be entered as a decimal with a maximum of 8 figures. Alternatively, the AVWF parameters can be entered as a gearbox factor (fraction) numerator/denominator, P180.1, P180.2 of the motor encoder.

Rated speed (U856)

The rated speed is a reference value used to represent speed.

The AVWF is also used to calculate the rated speed U856 which, in the factory setting, is 12288.00 [1000 LU/Min]. If this value is changed, the basic unit parameter P205 (rated V) must also be set to the same value, ignoring the figures after the decimal point.

Rated V = resolution x AVWF factor x reference speed x 10⁻³

Rated V: U856

Resolution: P171 [FD 330.3]

AVW factor: P169, P170 or P180, P181 [FD 330.3]

Reference speed: P353 [FD 20.5]

The rated V is specified in 1000 LU/min and the reference speed in rpm.

In the factory setting, the resolution of 2¹² = 4096 increments/revolution, an AVWF = 1.0 and 3000 rpm results in a rated speed of 12 288 000 LU/min. This speed then corresponds to 100% in the equipment.

In the example, the rated V derived from:

131 072 [LU/rev.], an AVWF of **0.71905288 [AVWF]** and 3000 rpm rated motor speed results in a rated speed of 282743097 [LU/min].

$$\begin{aligned} \text{U856} &= \text{rated V} = 2^{17} \times 0.71905288 [\text{IBF}] \times 3000 [\text{rpm}] \times 10^{-3} \\ &= \mathbf{282743.096} \text{ in } 1000 \text{ LU/min} \end{aligned}$$

The value is entered in 1000 LU/min with 2 decimal places
= 282743.09 in 1000 LU/min

Rated acceleration (U857)

The acceleration represents the change of speed (LU/min) within a time unit.

The rated acceleration is defined as the rate at which the drive accelerates from v = 0 to the rated speed (rated V) in t = 1 sec.

The following equation is used to calculate this:

Rated acceleration = U857 (in 1000 LU/s²)

Rated speed = U856 (in 1000 LU/min)

$$(\text{U857}) \text{ rated acceleration} = \frac{\text{rated speed (U856)}}{60 \times t}$$

Example (referred to factory setting):

Rated $V = 12288.00$ [1000 LU/Min],

results in the following with a desired runup time of 1 second from
 $0 \rightarrow 100\%$

$$U_{857} = \frac{12288.00 \text{ [1000 LU / Min]}}{60 \times 1} = 204.80 \text{ [1000 LU / s}^2\text{]}$$

the value calculated for the rated acceleration is transferred to parameter U857. It is used to normalize the acceleration setpoint (U852.1 or U869.x) and the deceleration setpoint (U852.2 or U869.x), which are specified in percent.

NOTE

Rated acceleration/deceleration $U_{857} = 0$ or
acceleration/deceleration setpoint in per cent via $U_{869.1} / .2 = 0$ means
maximum acceleration / deceleration.

NOTE

If an absolute encoder and odd-order gear factors are used, position
tracking has to be used (see Chapter 9.4.9 and 9.4.10).

You will find the basic description on the use of absolute encoders in
Chapter 9.4.6.

7.2.3.3 Operating modes

Setup mode (SETUP_ON)

In setup mode, the functionality of the setup/positioning block becomes available. This function block can be found in function diagram 789b and is assigned to a time slot using parameter U953.61.

Setup means position-controlled jogging.

In setup mode (SETUP_ON or SETUP = 1), the axis is moved under position control by means of the direction selection [D_FWD] and [D_BWD] taking into account the set values for acceleration and deceleration and the speed.

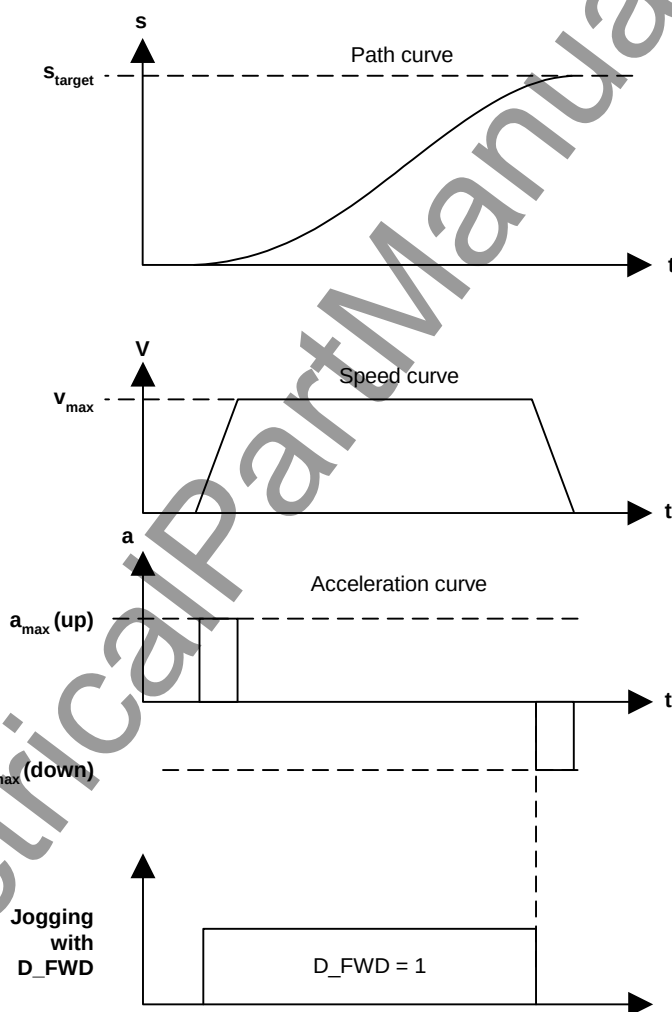


Fig. 7-8 Setup process SETUP = 1 and D_FWD = 11

To activate the setup mode, the enable positioner/homing command ENABLE_POS_REF must be set. Because of the order of priority, neither positioning (POS_ON) nor homing movement (REF_TYP = HIGH) may be set.

Homing mode (REF_ON)

In the setup mode with linear axis, the software limit switches become effective after homing.

The cycle length of a rotary axis is entered in the parameter U858 (axis cycle) in LU. U858 should be set to 0 in the case of a linear axis. The value (-1) in index 2 means that the value for index 2 = corresponds to the value in index 1.

The correction and homing block always provides the functionality. The associated function block can be found in function diagram 789c and is assigned to a time slot using parameter U953.62.

The homing mode is employed when incremental path encoders are used, as there is no relationship between the measuring system (incremental path encoder) and the mechanical position of the axis when the drive is switched on.

In this mode, a basic distinction is made between two types of homing. The type of homing depends on the selected homing type REF_TYP, which is set using parameter U875.10 or source U866.9.

The following settings are possible:

- ◆ REF_TYP = 0: Homing on the fly (post-homing)
- ◆ REF_TYP = 1: Homing movement in preferred direction and sequence control with reversing cams.

Homing on the fly REF_TYP = 0

Homing on the fly means setting the position setpoint and actual value simultaneously. This is implemented in the basic unit by position correction of the position detection and of the position setpoint of the position controller.

A reference position REF_setpoint is specified for this in U874.2. Alternatively, this reference position can be transferred via the connector to parameter U877.3.

From V1.6

The correction of position setpoint and position actual-value is always performed irrespective of the direction of rotation, unless (from V1.6) the direct setting of REF_D_REF is activated at the correction block (function diagram 789c) with REF_D_REF_EN = 1.

Signal SET_REF_D (B0895) always shows the status of REF_D (input Src U866.14 preferred direction). This is linked with the input REF_D_REF at U878.5 (factory setting), which also enables the direction to be evaluated during homing on the fly (function diagram 789c).

If the current direction of rotation corresponds to [SET_REF_D] (B0895), (1 = counter-clockwise / 0 = clockwise) and a valid measured value is received, a correction is carried out by means of "Start Ref." taking the "skip window" into account.

Homing movement REF_TYP = 1

Homing movement is initiated by a rising edge on REF_ON with homing movement selected by REF_TYP = 1 and the selection of a preferred direction using D_FWD or D_BWD. If the homing movement is terminated with REF_STOP and REF_ON is then removed, the active modes then take effect in their order of priority. REF_TYP is no longer taken into account. Homing movement is only initiated again with a rising edge on REF_ON.

Homing movement is initiated by a rising edge on REF_ON and the preferred direction selected using D_FWD or D_BWD. The reversing cams REF_STOP_FWD and REF_STOP_BWD reverse the movement until the homing signal [ARFD] on the input REF_STOP (U866.13) terminates the movement or the homing enable is removed.

For example, the homing signal is transmitted from a proximity switch to the interrupt-enabled digital inputs 4 or 5, thus saving the actual position at the moment the interrupt occurred in the motor encoder position measurement memory (function diagram 330). The source position measurement memory enable must be connected (P179 = 891) to the BPos measurement memory enable (function diagram 789c). The position measurement memory enable B0891 is controlled by the BPos.

The outputs from the position measurement memory pass to the correction block/homing of the BPos (function diagram 789c).

Actual position at interrupt U877.4 = 122 (position measurement memory)

Start referencing U878.3 = 212 (measurement valid)

No correction is performed if the deviation between the reference position U877.3 and the actual position is less than the innermost window F1 (U879.1). If the deviation is between F1 and F2 (U879.2), the position setpoint and actual values are corrected by the deviation. If the deviation is greater than F2, B0892 is set (print mark outside window 2).

In the factory setting, binector B0888 (axis referenced) is connected to the control signal REF_STOP (U866.13, function diagram 789a). If the input REF_STOP is set, the homing movement is stopped.

The axis then stops on the ramp, which means that it has not reached the homing position but was stopped ahead of or behind it, depending on the direction in which it was moving. If required, POS_ON can be used to carry out an absolute movement (POS_TYP = 1) to this homing point.

The binector input REF_FWD_STOP restricts the homing movement in the FWD direction and reverses the direction that was previously selected using D_FWD and Ref ON (or that is indicated on REF_STOP_BWD after the reversal).

The REF_D signal specifies the direction in which a "coarse pulse" is to be evaluated. This means that the evaluation of the "coarse pulse" on the measurement memory (position detection) in the opposite direction with respect to [REF_D] will be ignored and that the measurement memory will not be enabled by the BPos until the homing direction corresponds to that indicated by [REF_D].

Sequence of homing movement D_FWD = 1

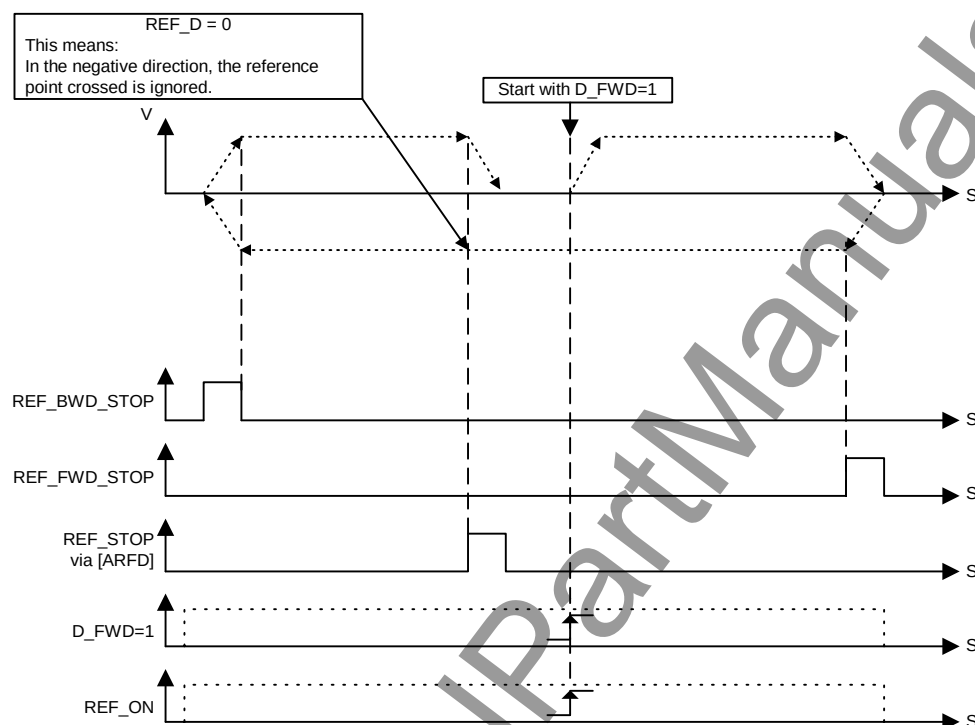


Fig. 7-9 Example of homing movement sequence start D_FWD to the right from the Proximity switch and homing direction REF_D = 0 (positive)

The setting values for acceleration and deceleration/delay are maintained during all movements. In the case of a constant transfer (U875.8 SPV_RIE_TYP = 1) these values can be changed with each rising edge on U875.7 SPV_RIE, even during the homing movement. If the axis is in the homing mode [ARFD] = HIGH, this process can be repeated as often as required (note: 1 sampling time delay!). No user intervention is necessary as REF_STOP (and consequently ARFD) are reset to zero with the rising edge on REF_ON.

Homing movement with coarse pulse and zero pulse evaluation

This can be implemented if required by means of the basic unit functionality (see position detection, function diagram 330) in the speed controller.

Positioning mode (POS_ON)

The setup/positioning block provides the positioning functionality. This function block can be found in function diagram 789b and, as for setup, is assigned to a time slot using parameter U953.61.

The positioner is a position controller whose position difference ($\Delta S[LU]$), which is derived from the setpoint position and actual position, is reduced to "0" using the specified acceleration and deceleration and maximum speed values.

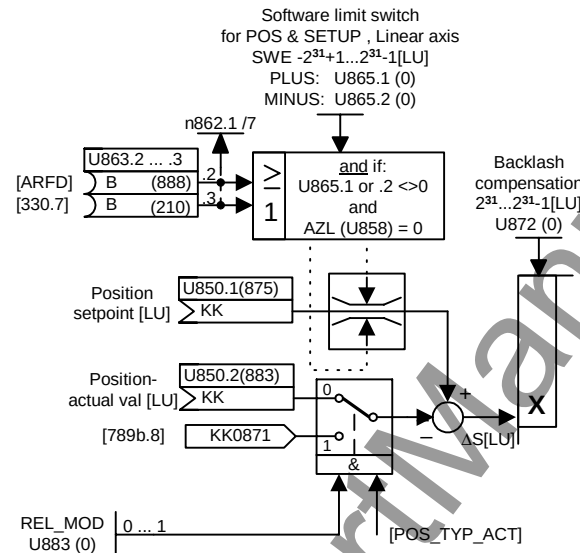


Fig. 7-10 Position controller: derivation of ΔS position difference

The positioner can be operated both in the block configuration described here or as a stand-alone function. The block then behaves as with constant transfer from the set setpoint module in function diagram 789a. The triggered variant is implemented by the upstream connection of the set setpoint block.

Changes to the setpoint are effective immediately.

The positioner consists of a position controller that controls the position difference ΔS to $\Delta S = 0$ while maintaining the specified acceleration and deceleration and the speed setpoint. In doing this, the position controller operates correctly according to established control technology principles.

Possible overrunning of the target position is not prevented, as the compensation movements are carried out within the limits of the specified setpoints (acceleration/deceleration).

Example: If a target position is not reached within the set ramp, braking takes place on the ramp and the axis travels to the target position in the opposite direction.

Behavior in accordance with correct control principles also means possible overrunning of the specified target position (oscillation).

DANGER



DANGER

In the case of constant transfer $SPV\_RIE\_TYP = 1$ (or with stand-alone operation of the function block) and active POS or SETUP with D_FWD_ACT or D_BWD_ACT , a movement is initiated when the controller is enabled.

There is **no** START enable or read-in enable; setpoints are evaluated immediately.

Software limit switches (with connector input from V2.3)

The software limit switches are only active in the case of a linear axis.

Reason: The image of a rotary axis only represents part of the range of movement over several axis cycles; there is therefore no point in having a limit within the axis cycle.

Recommendations:

If, for instance, a rotary table is to have a limited range of movement, it can have parameters assigned as for a linear axis.

Up to V2.2: To activate the software limit switches, parameter value $U865.1$ must be $\neq 0$ or $U865.2$ must be $\neq 0$

From V2.3: To activate the software limit switch the value at the connector input must be $U850.7$ or $U850.8 \neq 0$.

With the factory setting ($U850.7 = 898$, $U850.8 = 899$) this is effected by the parameter value $U865.1 \neq 0$ or $U865.2 \neq 0$.

The software limit switches have the effect of limiting the range of movement of the setpoints and take into account any possible backlash compensation (end position $\pm$ backlash) that may be set.

The software limit switches only affect a referenced ("homed") linear axis.

The checkbacks for this are already predetermined in the factory settings of parameters $U863.2$ and $U863.3$.

The sources are ARFD (function diagram 789c.7, B0888) from the homing using the basic positioner or "acknowledgement reference point detected" (function diagram 330.7, B0210) from the reference point detection function in the basic unit in n control mode (see basic unit description - position detection).

If a software limit switch is approached, it is only possible to move away from it in the opposite direction. This is possible by specifying a new target position outside the limits of the software limit switches or by jogging in SETUP mode in the opposite direction to the software end limit.

NOTE

The effect of the software limit switches can be deactivated or reconfigured by the user (using BICO technology).

Example:

SWE_MINUS : 1000 LU

SWE_PLUS : 150000 LU

Permissible range of movement from 1000 to 150000 [LU]

Stop cam (from V2.3)

The stop cam function is activated with value 1 at binector input U866.1 (SC_ON = 1).

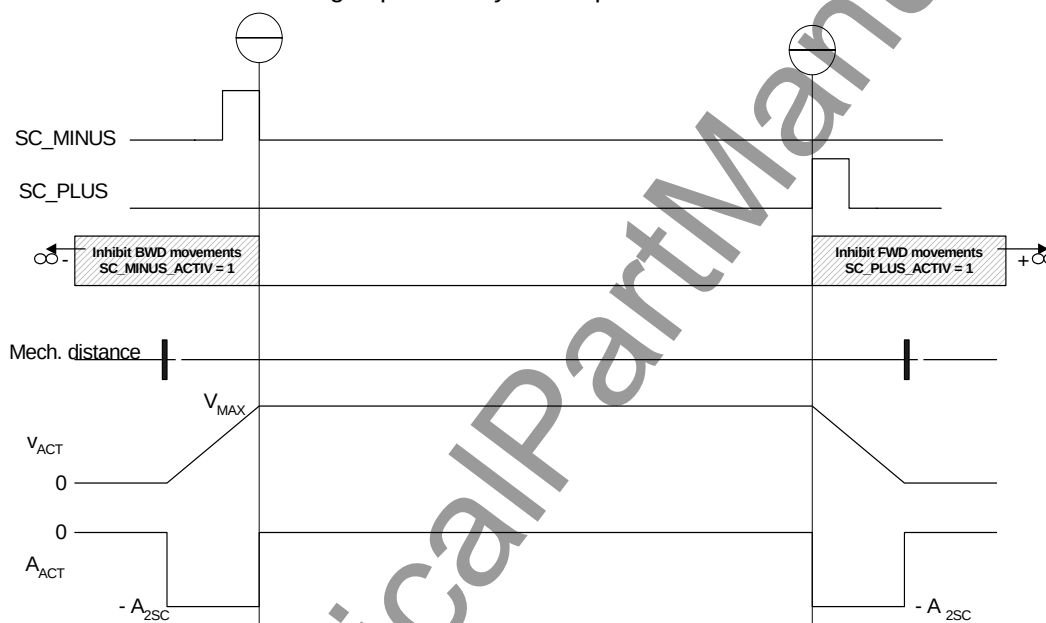
After enabling (SC_ON = 1) the respective movement is stopped with the "deceleration setpoint A_{2SC} " via STOP cam (SC_PLUS = 1 and/or SC_MINUS = 1).

The "deceleration setpoint A_{2SC} " must be selected such that sufficient mechanical distance-to-go is available at maximum speed.

Depending on the actuated STOP cam, SC_PLUS_ACTIV = 1 or SC_MINUS_ACTIV = 1 is checked back.

The selected operating modes remain. Only the valid speed setpoint is set to zero.

The following path diagram shows the behavior on leaving the travel range specified by the stop cam.



Only movements that lead out of the range of the STOP cams are permissible.

After leaving the respective limit switch by a falling edge in the valid direction of movement, SC_PLUS_ACTIV = 0 or SC_MINUS_ACTIV = 0 is checked back.

If both STOP cams are available (SC_PLUS = 1 and SC_MINUS = 1) no movement is carried out as long as SC_ON = 1 (Enable STOP cam).

Static STOP cam (from V2.4)

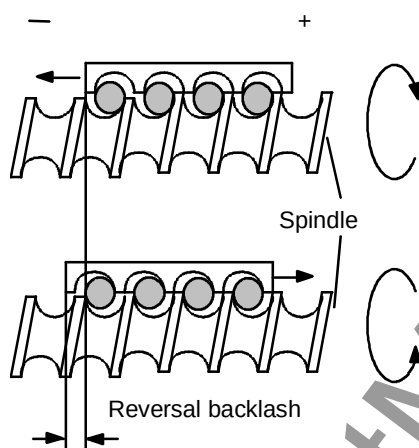
The static STOP cam function is activated with SC_ON = 1 and U887 = 1. The activated STOP cams are evaluated purely statically in this operating mode.

Overtravel of the cams is not supported which is why the cams have to be executed up to the end stop!

Reversal backlash compensation (U872)

To activate reversal backlash compensation, parameter U872 must be $\neq 0$.

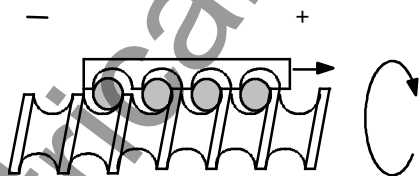
Reversal backlash compensation is used to compensate for mechanical reversal backlash. In the case of an indirect measuring system (path encoder on motor), the mechanical backlash is first traveled before the effective (real) axis movement begins. The result is position errors. There is no minimum travel distance.



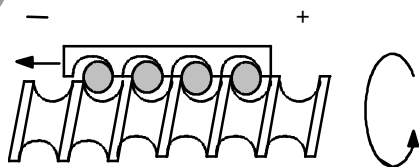
The sign gives the preferred position for the reversal backlash compensation.

This means:

Positive value = positive preferred position → on the first positive movement after switching on the converter, no reverse backlash is taken into account.



Negative value = negative preferred position → on the first negative movement after switching on the converter, no reverse backlash is taken into account.



Parameter value = 0:

No reversal backlash compensation takes place.

The backlash compensation is taken into account in the software limit switches as follows:

If movement takes place over the software limit switch in the direction of the backlash, the axis stops at the software limit switch plus backlash.

This means that the unit itself is at the software limit switch but the position setpoint is beyond it.

Example:

Negative position (preferred position neg.)		Positive position (preferred position plus)
Software limit switch plus:	100000 LU	100000 LU
Software limit switch minus:	50000 LU	50000 LU
Backlash:	-100 LU	200 LU
Specified setpoint:	150000 LU	150000 LU
Output setpoint:	100100 LU	100000 LU
Actual-value:	100100 LU	100000 LU
Specified setpoint:	0 LU	0 LU
Output setpoint:	50000 LU	49800 LU
Actual-value:	50000 LU	49800 LU

Absolute positioning

In the case of absolute positioning, absolute equality is created between the position setpoint and the actual position value.

Setpoint = actual value

The POS input enables the positioner and movement takes place to the specified setpoint using the positioning movement. A new target position can be set at any time by changing the position setpoint.

If the POS input is reset during absolute positioning, movement is stopped immediately while maintaining the specified acceleration and deceleration and the specified speed setpoint. If the POS input is set again, the setpoint becomes valid again and the position setpoint is approached again.

With linear axis

In the case of a linear axis, a position setpoint in the range -2^{31} to $+2^{31}-1$ is possible, i.e. the range can be used to its full extent.

The software limit switches can be used with a linear axis.

In general, the use of a linear axis makes sense for limited movement paths. Nevertheless, it must be ensured that the path representation fits into the range.

Example:

A path is to have a resolution of $1/1000 \text{ mm} = 1 \text{ } \mu\text{m}$:

4294.967297 could be represented with a 32 bit position setpoint of $-2^{31} = -2147483648 \text{ [LU]}$ to $+2^{31}-1 = 2147483647 \text{ [LU]}$.

Because of the resolution in 2^n steps and the mechanical coupling, a actual-value weighting factor (AVWF) usually results that allows the position setpoint to be converted to mechanical μm or [LU].

With rotary axis

In the case of a rotary axis ($U858.1 \neq 0$), we speak of an axis cycle. The axis cycle can lie in the counting range from 0 to $+2^{31}-1 = 2147483647$ [LU]

Also in the case of a rotary axis, absolute equality is created between the setpoint position and actual position value within an axis cycle.

This means that the target position only lies within the axis cycle.

The movement is specified by the direction: shortest path, only positive or only negative.

See: evaluation of direction selection D_FWD / D_BWD

The setpoint is calculated and corrected (MODULO) within the axis cycle for both negative and positive setpoints.

Example:

-5000 becomes 3192 with $AZL = 4096$ ($5000 \text{ MOD } 4096$)

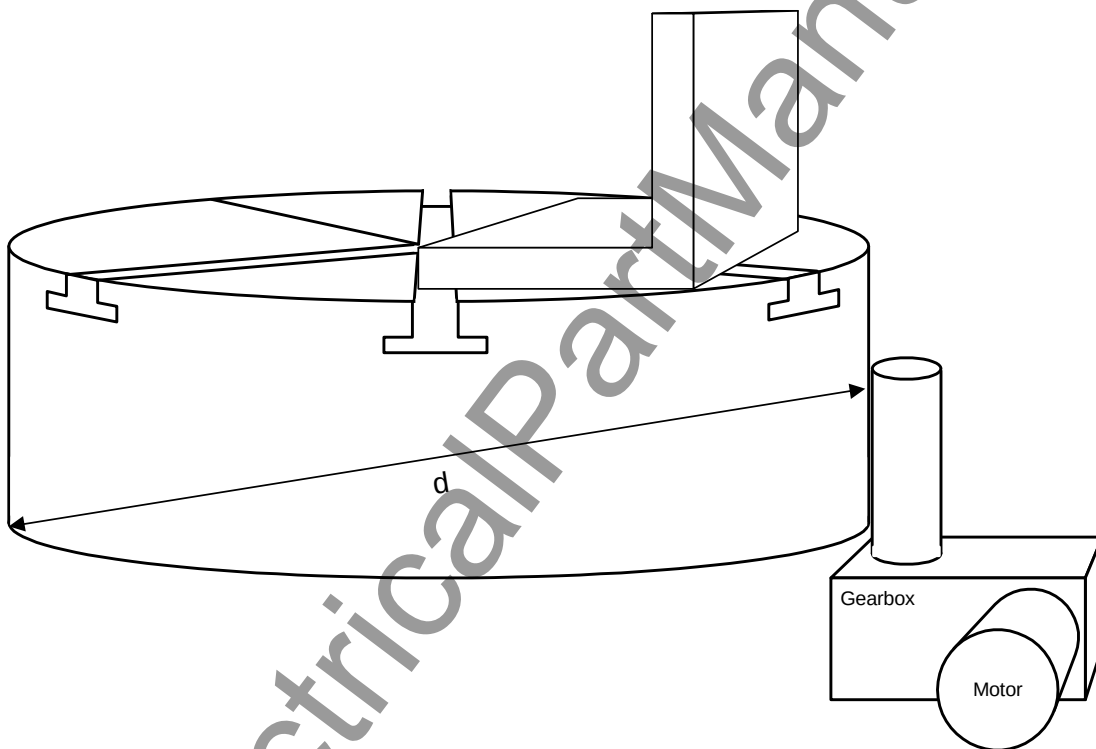


Fig. 7-11

Example of rotary axis, the rotary table

Relative positioning

In the case of relative positioning, also known as incremental travel, movement takes place over a specified distance. This movement is initiated by a positive edge on POS_ON (enable positioning) or transfer with positive edge in transfer type SPV_RIE_TYP = 0.

Re-triggering of this movement is possible at any time. In this case, the remaining distance is disregarded and a new relative distance is traveled from the time of triggering.

Example: Actual value = 12567
Setpoint = 5000
New target position: $12567 + 5000 = 17567$

Setting the homing on the fly changes the actual value. As a result, the target position reached depends on the displacement of the measuring system caused by homing on the fly. This can, however, be selected to be dependent on the RELMOD mode (U883) (see next section).

The software limit switches refer to the sum of the distance (actual value) and interrupt relative positioning when the software limit switch is reached. Further movement can only take place in the direction opposite to that in which the software limit switch was approached.

NOTE

The relative movement is interrupted if the POS-ON is removed before the target position is reached or a positive edge is given on SPV_RIE. In this case, the remaining distance is deleted.

Recommendations:

if a relative movement is to be interrupted, i.e. without again travelling the total relative distance, the speed setpoint on U851 can be set to 0% by an analog switch. Stopping then takes place on the ramp. When switched again to a valid setpoint, the axis travels to the old setpoint position, i.e. the remaining distance will then be traveled.

Relative positioning mode

RELMOD = 0
(U883=0)

For relative positioning (POS_TYP_ACT = 1), the actual value from the position value content of source U850.2 is used, as in all other modes.

This means that, when homing on the fly, the corrected setpoint is taken into the travel distance calculation ($SET = ACT$).

NOTE

The positioner behaves in accordance with correct control principles. In the case of a rotary axis, the correction is carried out by the shortest path. This may mean that a reversal of direction is possible, especially if the correction value $> 1/2 \times$ axis cycle.

**RELMOD = 1
(U883=1)**

In the case of relative positioning (POS_TYP_ACT=1), the internal position value S_pos (KK0871) is used.
This means that, when homing on the fly, the corrected setpoint is not taken into the travel distance calculation (SET <> ACT).

NOTE

In some circumstances, the software limit switches no longer refer to the actual value of the measuring system, as the internal position value is used.

The travel distance is not corrected, the distance that is specified is traveled.

With linear axis

In the case of relative positioning with a linear axis, the valid position setpoint is traveled as the distance.

The movement is only limited to the range
(from -2^{31} to $+2^{31}-1$).

The software limit switches can be used with a linear axis.

With rotary axis

In the case of relative positioning with a rotary axis, the valid position setpoint is traveled as the distance.

The movement is otherwise limited to the range
(from -2^{31} to $+2^{31}-1$).

The software limit switches cannot be used with a rotary axis.

Auxiliary inputs:

The purpose of the auxiliary inputs is to set the positioner output and to implement tracking mode.

Setting value

As standard, the setting value is the actual position value (KK0120 motor encoder) to enable the actual position value to track to the position setpoint (jump suppression).

Position setting value

The position setting value can also be from another source (motor encoder/machine encoder) depending on the application.

Set trigger

With [ENABLE_POS], output KK0871 tracks the position setting value source U850.3 (statically) and all statuses are reset.

ENABLE_POS

With [ENABLE_POS], output KK0871 tracks the position setting value source U850.3 (statically) and all statuses are reset.

ENABLE_REF

With [ENABLE_REF], output KK0882 tracks the position setting value source U877.2 (statically) and is corrected in the axis cycle. This implements the rotary axis representation of the actual position value in tracking mode.

Position of current actual value

Here, the actual value of the POS-OK representation is read; the actual position value can come from the motor encoder or machine encoder, but can also be from any other source.

The purpose of these auxiliary inputs is to delay the POS_OK signal output and to influence it if necessary.

**Window width
Pos OK**

For the POS_OK representation, a window width (exact positioning window) in LU is invoked. The POS_OK checkback is set when the target position is reached.

POS_OK delay time	POS_OK forms the delay time from the time parameter 0 to 100.00 s after reaching the target position in the window. A target position is expected after a time of 0 to 100.00 s to be certain that the position is held.
External position OK	To influence POS OK externally or to ensure a handshake with another unit or to hide the checkback, the signal is evaluated in parallel.
Setpoint mode output:	
B0871: REF homing	If the control binector "homing ON" [REF_ON] is active and travel takes place in the selected direction REF_D, REF is set to HIGH to enable the homing function on function diagram 789c. Note: This signal is independent of status U866.1 ENABLE_POS/REF so that, when the positioner is enabled again through status U866.1 ENABLE_POS/REF, status ARFD = 1 is not reset. (See the "Mode management" section.)
B0872: POS positioning	If the block is enabled (ENABLE_POS/REF = 1) and the "positioning ON" binector [POS_ON] is active and if no homing movement (REF_DRIVE = 1) is valid, status binector POS = 1 is shown. (See the "Mode management" section.)
B0873: SETUP	If the block is enabled (ENABLE_POS/REF = 1) and the "setup ON" binector [SETUP_ON] is active and if no positioning or homing movement (REF_DRIVE = 1) is active ([POS_ON] and [REF_ON] = LOW), status binector [SETUP] = 1 is shown. (See the "Mode management" section.)
B0877: PSR positioning/homing/setup active	If the "positioning" status binector [POS] or the "homing" status binector [REF] or the "setup" status binector [SETUP] is active, this is shown through the status binector [PSR] = HIGH.
B0893: REF_DRIVE homing movement active	If the block is enabled (ENABLE_POS/REF = 1 and REF_TYP = 1 and REF_ON = 1), REF_DRIVE is set to 1.
B0895: SET_REF_D preferred direction	SET_REF_D (B0895) = REF_D (source U866.14), irrespective of all operating modes.
B0874: POS_TYP_ACT current POS_TYP	The "current POS_TYP" status binector [POS_TYP_ACT] always shows the signal level of the last valid, i.e. transferred, status binector [POS_TYP]. See section about [POS_TYP] control binector. [POS_TYP_ACT] = 0: ABSOLUTE positioning through the [POS_TYP] control binector is transferred/valid. [POS_TYP_ACT] = 1: RELATIVE positioning through the [POS_TYP] control binector is transferred/valid.

B0875: D_FWD_ACT Positive direction active	The "D_FWD active" status binector [D_FWD_ACT] always shows the signal level of the last valid, i.e. transferred, status binector [D_FWD]. See section about [D_FWD] and [D_BWD] control binectors.
B0876: D_BWD_ACT Negative direction active	The "D_BWD active" status binector [D_BWD_ACT] always shows the signal level of the last valid, i.e. transferred, status binector [D_BWD]. See description of [D_FWD] and [D_BWD] control binectors. See "Setup/positioning output" section.
B0860: POS_OK Position OK	The binector has a HIGH signal: ◆ if positioning is switched on ([POS] = HIGH) and ◆ if, as in the case of ABSOLUTE and RELATIVE positioning, the current actual position value is within the assignable parameters of the position window (U859; $\pm$ tolerance in [LU]), referred to the expected position (valid position setpoint).
B0861: POS_RUN Positioning running	The status binector "positioning running" is only HIGH if the axis moves (ramp runs) with positioning selected ([POS] = HIGH). i.e. [POS] = HIGH AND [AXS_RUN] = HIGH
B0862: RFG_RUN Axis in motion	The "Axis in motion" status binector will always be HIGH when the axis is moving (ramp-function generator running). This may be the case during homing and/or positioning or when slowing down to a stop (positioning/homing disabled; invalid input [POS_TYP] with constant SET setpoint transfer, etc.).
B0863: RU_ACT Acceleration active	The binector will only be set HIGH when the RFG is accelerating, irrespective of the mode of operation (positioning, homing, etc.).
B0864: RD_ACT Deceleration active	The binector will only be set HIGH when the RFG is decelerating, irrespective of the mode of operation (positioning, homing, etc.).
B0876: POS_DELTA Position not yet reached	The binector will be set HIGH when positioning has been activated (POS = 1) but the target position has still not been reached (valid position setpoint). If a new SET setpoint is transferred [SPV_RIE], [POS_DELTA] will also be reset again. The remaining traverse path will be deleted.
B0866: FWD_RUN Forwards running	The binector will only be set HIGH when the RFG has generated a positive movement, irrespective of the mode of operation (positioning, homing, etc.).
B0863: BWD_RUN Backwards running	The binector will only be set HIGH when the RFG has generated a negative movement, irrespective of the mode of operation (positioning, homing, etc.).
B0868: SW_E_PLUS Plus software limit switch	The binector will only be set HIGH when the software limit switches are active (approached); the movement has been limited by the Plus software limit switch.

**B0869: SW_E_PLUS
Minus software limit
switch**

The binector will only be set HIGH when the software limit switches are active (approached); the movement has been limited by the Minus software limit switch.

**B0888: ARFD axis
referenced****Positioner/correction value and homing**

The binector is only set HIGH following a successful **homing** operation (valid measurement in the enabled travel direction [REF_D] taking account of the masking window function or active correction performed).

The signal remains HIGH until the ARFD binector is reset through REF_ON being set again.

The signal will automatically be reset if there is a POWER-OFF on the converter or [REF] is set again, unless an absolute value encoder is present (i.e. [ARFD] **remains** HIGH following the first valid reference mark if [REF] is on and a number of coarse pulses have been received (reference mark).

The signal is not continuously set to HIGH in the case of absolute value encoders.

NOTE

If an absolute value encoder is used, the software limit switches which require the signal [ARFD] = High at U863.2 for their activation, can be activated by manually setting the signal to high at U863.2.

CAUTION

The user is responsible for monitoring the signal and ensuring the appropriate interlocks are in place!!!

**B0892 : F_REF_WD
fault Reference
point correction
outside window 2**

Binector for masking window homing
(see "Masking window for homing" section)

Status binectors/connectors/visualization parameters

The parameter U862 BPos RM-Signal shows the status of the basic positioner as status signals.

Index 1 : BPos (K0888) input

BIT0 = ENABLE_POS
BIT1 = RESERVED
BIT2 = POS
BIT3 = SETUP
BIT4 = POS_TYP_ACT (was: ABS_REL)
BIT5 = D_FWD_ACT
BIT6 = D_BWD_ACT
BIT7 = EXT_REF_OK B0888 or B0210 = 1
BIT8 = EXT_POS_OK
BIT9 = SET_TRIG
BIT10 = Internal POS_OK (reached position)

Index 2 : BPos output and homing command (K0889)

BIT16 = B0860 [POS_OK]
BIT17 = B0861 [POS_RUN]
BIT18 = B0862 [RFG_RUN]
BIT19 = B0863 [RU_ACT]
BIT20 = B0864 [RD_ACT]
BIT21 = B0866 [FWD_RUN]
BIT22 = B0867 [BWD_RUN]
BIT23 = B0865 [POS_DELTA]
BIT24 = B0868 [SW_E_PLUS]
BIT25 = B0869 [SW_E_MINUS]
BIT26 = B0888 [ARFD]
BIT27 = B0892 [F_REF_WD]

7.2.3.4 Preprocessing of position setpoint

The correction block is used to preprocess the position setpoint. The correction block can be found in FD789c and is inserted into the time slot using U953.62.

Its function is to provide the corresponding signals for the position controller and the position detection.

Example of the connections to the basic unit when the motor encoder is being used:

Position control [FD340]

position setpoint	P190 = 882
Speed setpoint	P209 = 881

Position detection [FD330]

correction value	P174 = 885
Correct position POV/NOV	P175.1 = 889, P175.2 = 890
Enable measurement memory	P179 = 891

If a rotary axis is in use, the actual position has to be corrected by means of the control inputs in order to detect the position correctly.

In the case of homing, the correction value is taken from the position measurement memory.

Any rate-of-change limitation required is provided at this point in order to pass the corrected actual value to the positioner (without rate-of-change limitation) during homing. This (KK0833) causes the position controller to function as a closed control loop and carry out homing corrections without any jerks or sudden changes.

Smoothing adaptation (rate-of-change limitation)

This programmable rate-of-change limitation is estimated using an adaptive connector input (U881).

Rate-of-change limitation is disabled if the parameter has a value of 0 or is assigned a value of 0% from an adaptive connector input.

The values and parameters can be changed if the ramp function generator is not active (U876 V- set IN = 0).

NOTE

"Computed" smoothing as found in the "deluxe" ramp function generator has been dispensed with owing to the amount of computing time required.

Masking window for homing

Setting of a homing point on the fly can be influenced through two movable windows. The points of reference of the windows are based on the reference position of U874.2 or the source on U877.3 and define the permissible deviation between the reference position and the measured position.

The windows allow setting of the homing point to be suppressed if the deviations are too small or too large.

NOTE

The contents of the windows are only used when the axis is referenced (ARFD = 1). The first sensing of the reference mark following runup leads to a setting of the homing point, irrespective of the window settings.

If the deviation is within the innermost window (window 1), the homing point is not set. Neither is the homing point set if the deviation is outside the outermost window (window 2).

The output B0892 error homing point proximity switch outside window 2 is activated.

This output signal stays on until a homing point mark is next evaluated.

Both windows can be disabled individually by assigning them values of 0.

Fault and warning messages

The free BASIC POSITIONER blocks do not generate any fault or warning messages. The positioning software does not respond to messages caused by incorrect parameters either (e.g. violation of limit values).

Definitions

Def. homing:

Homing on the fly, also referred to as post-homing, appears in the correction value/homing block, function diagram 789c, with window evaluation.

Homing as homing movement mode with reversing cam appears in the setting value/setpoint block function diagram 789a.

7.2.3.5 Application example

Definitions and warnings

Qualified personnel

within the scope of the documentation are individuals who are familiar with the erection, installation, commissioning, operation and maintenance of the SIMOVERT MASTERDRIVES product and who have the appropriate qualifications to perform their activities, such as:

- ◆ trained and authorized to energize, de-energize, ground and tag circuits and equipment in accordance with established safety procedures.
- ◆ trained in the proper care and use of protective equipment in accordance with established safety procedures.
- ◆ trained in rendering first aid.

This document does not contain any explicit warnings. However, reference is made to the warnings contained in the operating instructions for the relevant product in the MASTERDRIVES range.

The application examples are provided free of charge. They may be copied, modified and used and passed on to third parties. They may only be passed on in a complete and unmodified state together with all patent rights notices. The commercial distribution to third parties (e.g. as shareware or freeware) is only permitted with the prior written permission of Siemens AG.

NOTE

As the application examples are provided to you free of charge, the authors and copyright holders are unable to entertain any warranty claims. You use them at your own risk. The authors and copyright holders shall only be liable for their premeditated actions and gross negligence. All other claims are excluded. In particular the authors and copyright holders accept no liability for any defects or consequential damages. Please let us know if you find any errors in the application examples.

Standard applications

Basic applications in the form of scripts including full documentation

are available, e.g.:

- full functionality of the basic positioner through PROFIBUS interface;
- basic positioner through terminal strip interface,
- etc..

NOTE

Standard applications are available including parameter assignment and documentation. These can be obtained from your regional SIEMENS AG office and are available from the Application Center for Production Machines.

Lag monitoring

An example of lag monitoring with variable limit values [in LU] on U015 for Standstill Lag and U016 for Travel Lag.

If F148 trips, the drive shuts down with pulse disable (drive coasts).

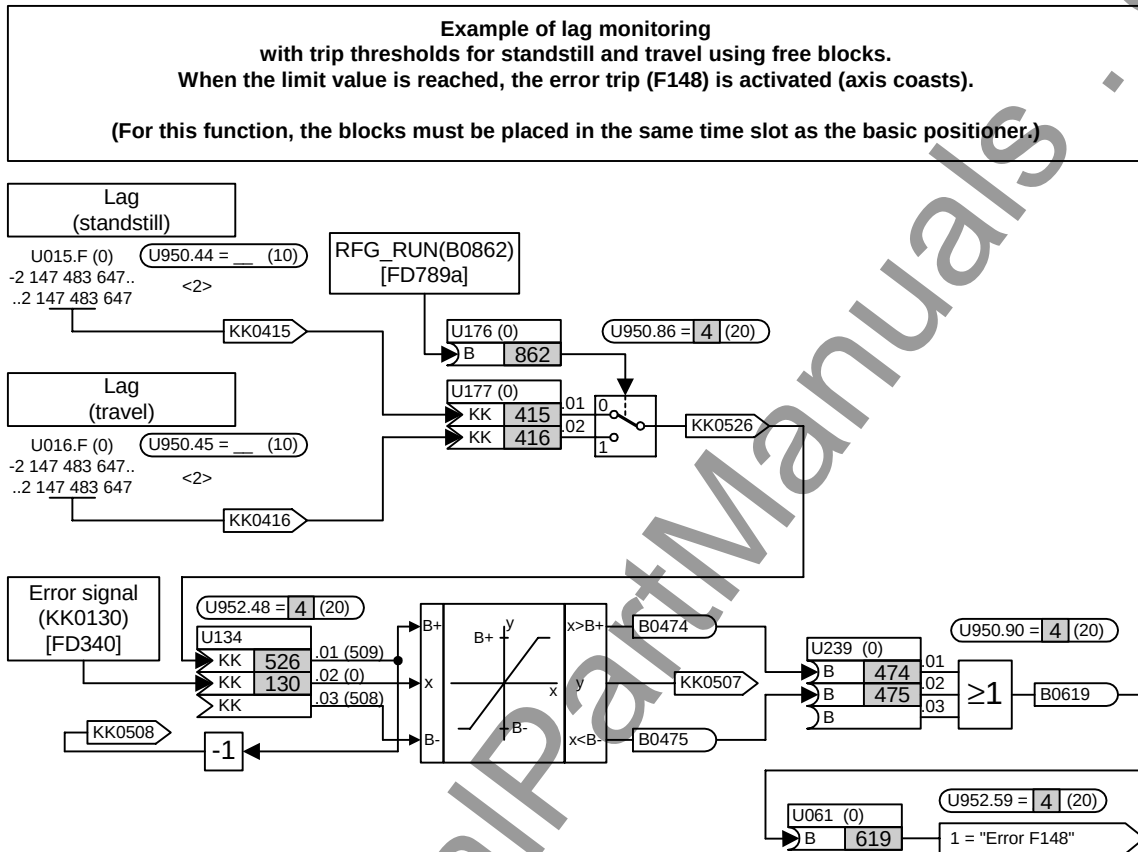


Fig. 7-12 Implement lag monitoring using free blocks

NOTE

The user should determine the shutdown strategy for the device using the appropriate operating mode OFF1, OFF2, OFF3 with brake control or pulse disabling through fault message triggering (as in the example).

Script file: Example of parameters for lag monitoring

```

SET LOG ON
REM *****
REM * File name: EP_Schlepp.txt *
REM * Script file for MASTERDRIVES Compact Plus and *
REM * MASTERDRIVES Compact *
REM * Date : 27.09.2000 *
REM *
REM * Example of lag monitoring using free *
REM * blocks. *
REM * Standstill Lag 100 LU *
REM * Travel Lag 1000 LU *
REM *
REM *****

REM *****
REM ** Fixed setpoints for Standstill Lag [FD705] **
REM *****

REM ---Insert in time slot
WRITE 2950 44 4
REM ---Standstill Lag
WRITE 2015 0 100

REM *****

REM *****
REM ** Fixed setpoints for Travel Lag [FD705] **
REM *****

REM ---Insert in time slot
WRITE 2950 45 4
REM ---Travel Lag
WRITE 2016 0 1000

REM *****
REM ** Analog signal selector switches [FD750] **
REM *****

REM ---Insert in time slot
WRITE 2950 86 4
REM ---Switch of RFG_RUN axis running
WRITE 2176 0 0x862
REM ---Standstill Lag
WRITE 2177 1 0x415
REM ---Travel Lag
WRITE 2177 2 0x416

```

```

REM *****
REM ** Limiters with threshold checkback signal [FD735] **
REM *****

REM ---Insert in time slot
WRITE 2952 48 4
REM ---Pass limit value from switch
WRITE 2134 1 0x526
REM ---Control difference from position controller as input [FD340]
WRITE 2134 2 0x130

REM *****
REM **   OR with limit value signal  [FD765]   **
REM *****

REM ---Insert in time slot
WRITE 2950 90 4
REM ---Evaluate upper B+ from limiter
WRITE 2239 1 0x474
REM ---Evaluate upper B- from limiter
WRITE 2239 2 0x475

REM *****

REM *****
REM **   Trigger fault message [FD710]   **
REM *****

REM ---Insert in time slot
WRITE 2952 59 4
REM ---Wire up OR output
WRITE 2061 0 619

REM *****
SET LOG OFF

```

7.2.3.6 Change history

V1.60

- ◆ Selection of speed-dependent enable "Homing on the fly" U878.5 / U878.6, see Section 7.2.3.3 "Operating modes".
- ◆ Measured value _OK U878.7 for correction block / homing newly introduced.

NOTICE

Factory setting for motor encoder (B0070)
If a machine encoder is used B0071 (measured value valid / machine encoder) has to be parameterized.

V2.10

- ◆ An additional speed input U850.5 has been introduced as a percentage input for improving the quality of the checkback signals.
- ◆ The value range of the software limit switch has been adjusted to $\pm 2^{31}-1$.

7.3 Converter functions

7.3.1 Friction characteristic function (function diagram 399)

7.3.1.1 Friction characteristic

The friction characteristic is made up of 10 intermediate points each with a speed value (x axis) and a torque value (y axis). The pair of values (U215, U216) defining the intermediate point is to be given in % of the reference speed and in % of the reference torque.

The characteristic refers to the absolute value of the input variable (U214, e. g. KK0091 actual speed value). The derived torque (K0615) is output as a positive or negative value according to the current sign of the input. Weighting of the friction characteristic can be performed with the aid of parameter U217. BICO parameter U218 serves for selecting the source from which the friction characteristic is switched in (switched out => K0615 = 0).

7.3.1.2 Friction characteristic recording (automatic procedure)

Start of measuring BICO parameter U219 is used to select the source for starting automatic recording of the friction characteristic. When the status of this source changes from 0 to 1 friction characteristic recording is started, feedforward control of the friction characteristic is prevented internally (K0615 = 0).

In the first step, the necessary enables and assignments are checked:

Prescribed assignments:

- ◆ Master drive (P587 = 0)
- ◆ P260 = 153 or P262 = 153;
- ◆ P228 = 152;

Necessary enables:

- ◆ Enable pulse
- ◆ Enable speed control
- ◆ Enable direction of rotation (positive and/or negative)
- ◆ Positive and negative speed limits (P452, P453) chosen so that, with due regard to the direction of rotation enabled, the characteristic takes full effect in the pertinent direction of rotation.

If incorrect assignments are made or an enable is absent, fault F099 occurs.

If the assignments and enables are correct, the converter displays alarm A72 and waits for the ON command in order to begin with rotary measuring. If the ON command is not given within 30s, friction characteristic recording is interrupted with fault F099.

NOTE

Measuring can be started only from converter status °008 and °009.

Rotary measuring

After the ON command, the converter approaches all characteristic points independently. The ramp time is independently determined by the converter, but is not less than 2 s. If both directions of rotation are enabled, the positive and negative speed values of the characteristic points are approached alternately. The mean is derived from the measured friction torques.

If it is not possible to approach all characteristic points in the positive and negative directions of rotation, the converter displays alarm A74. At the end of friction characteristic recording, the drive is stopped, the values recorded (provided that no fault occurs, or recording is not interrupted) are transferred to parameter U216 and the converter is switched off. Binector B0690 changes to 1 and indicates the end of friction characteristic recording. After the friction characteristic recording command (U219) is cancelled, binector B0690 is reset to 0.

Interruption of measuring

Measuring can be interrupted at any time by canceling the ON command or by a fault (with the exception of F099).

The converter then displays alarm A73 and waits to be switched ON again. After being switched ON (or fault acknowledgement and switching ON) the converter continues measuring from the point at which the interruption occurred. If there is a delay of longer than 5min before the converter is switched ON, the converter ends friction characteristic recording with fault F099.

Termination of measuring

Friction characteristic recording can be interrupted by the converter itself on occurrence of fault F099. Fault F099 while measuring is in progress is caused by:

- ◆ BICO change, or function dataset change (not allowed during friction characteristic recording)
- ◆ Change/cancellation of direction of rotation enable
- ◆ Speed setpoint not reached
- ◆ Measured value not plausible (e.g. < 0 in the case of positive direction of rotation)
- ◆ Cancellation of friction characteristic recording command (U219 = 0)

The values acquired up until termination of measuring are **not** transferred to parameter U216.

Friction characteristic faults and alarms

F099:

Measuring has been terminated, the reason is given by the fault variable (P949).

Meaning of the fault variable:

Bit	Val.	Meaning
0	1	Positive direction of rotation not possible
1	2	Negative direction of rotation not possible
2	4	Releases absent
3	8	Assignment not allowed
4	16	Termination through cancellation of the record command
5	32	Dataset switchover
6	64	Time exceeded (switch ON or switch ON again)
7	128	Measuring fault: Measuring point not reached or measured value not plausible.

A72:

Waiting for ON command, max. 30 s.

A73:

Waiting for ON command when measuring interrupted, max. 5 min.

A74:

Measuring of **all** characteristic points in the positive **and** negative directions of rotation is not possible.

7.3.2

Torque constant adaptation for synchronous motors (function diagram 393)

The function "torque constant adaptation for synchronous motors" serves to improve the absolute torque accuracy for control of synchronous motors. Variations in the magnetization of the permanent magnets arise from manufacturing tolerances and temperature fluctuations.

This "kT estimator" function adapts the torque constant k_T [Nm/A] in the control to the instantaneous magnetization.

It makes sense to use the k_T estimator only in combination with the friction characteristic because the k_T estimator corrects only the internal torque of the machine. The friction losses must be compensated by an additional torque derived from the friction characteristic.

Using the kT estimator

The kT estimator needs the most exact possible motor parameter values in order to attain high torque accuracy. Before use is made of the kT estimator, motor identification (P115 = 2) must therefore be performed to define the values for P119, P120 and P121. The motor should be at room temperature for identification.

The motor temperature is needed by the estimator for tracking the temperature-dependent variables. If no motor temperature sensor is wired up, good accuracy is attained only during operation at the temperature at which motor identification was performed.

The kT estimator is not activated until a definite speed (P091.1) is reached. The voltage at the converter terminals is always subject to minor errors, introduced by voltage drops at the semiconductors, etc. The lower the speed and hence the output voltage, the more the estimate is disturbed by minor voltage errors. That is why the estimate is turned off below a definite speed (factory setting: 20 % of rated speed). When the speed falls below this level, the last-estimated value is frozen.

The estimator is activated by setting the maximum deviation (P091.2) to a value greater than 0 %. To turn on the estimator, this value can be set to 30 %.

The standstill torque constant (P098) is preassigned with the stored value in the case of a Siemens motor, and with the value $M_{\text{rated}}/i_{\text{rated}}$ in the case of a non-Siemens motor. The value can vary as a result of manufacturing tolerances. If the estimator has been activated and the motor speed is higher than the selected speed, a corrected standstill torque constant value can be read from visualization parameter r088. This value can then be entered in P098.

The temperature dependence (P090.2) of the magnet material is set to 12 % at the factory. This means that magnetization declines by 12 % when the temperature of the rotor rises by 100 K. This is the usual value for the neodymium-iron-boron magnet material now used. If no temperature sensor is connected, the temperature adaptation is turned off.

The temperature adaptation is also effective when the estimated value is frozen or when the estimator was turned off with P091.2 = 0.

7.3.3 Tr adaptation function (function diagram 394)

The Tr adaptation function serves to improve torque accuracy in the case of induction machine control. The rotor time constant (T_r) has a substantial effect on determining the slip frequency, and hence on the calculated field angle.

Because it includes the rotor resistance, the rotor time constant depends strongly on temperature and can therefore vary by up to 50 %. Such variations lead to wrong orientation of the dq system and hence to an error in the injected torque variable.

Initial start-up of the Tr adaptation

Temperature tracking for the rotor time constant is based on a voltage model that has to operate with the most precise possible motor parameters. Before the Tr adaptation is used, motor identification ($P115 = 2$) **must** be performed to define the values $P111.1$ to 10, $P121$, $P122$, $P123$. The motor should be at room temperature during identification. In the case of a non-Siemens induction machine ($P095 = 4$), automatic parameterization ($P115 = 1$) is necessary before motor identification starts.

Tr adaptation is turned on by setting $P092$ to a value greater than 0 %.

Because of the underlying principle, the voltage model returns sensible results only at rotor frequencies greater than 3 Hz and at load currents greater than $0.15 \times$ motor rated current. If these conditions are not satisfied, Tr adaptation is turned off internally and the last-calculated value is frozen. Visualization parameter $r093$ indicates the actual rotor time constant value effective in the flux model referred to $P124$.

7.3.4 Position test function

On synchronous motors, the torque control must be able to detect the position of the rotor in the motor so that the current is always injected at the correct position. That position is supplied by the encoder, which is installed in the motor (resolver, multiturn encoder, encoder). The encoder is mounted on the rotor with the correct alignment in the factory. However, if the encoder is replaced as the result of a fault or if a non-Siemens synchronous motor is operated on the MASTERDRIVES converter, the alignment of the encoder and the direction of rotation must be checked and corrected, if necessary.

The position test consists of injecting a current with a fixed angular position into the stator of the synchronous motor. The rotor must be free to move so that it can align itself toward the injected current with its permanent magnet (if necessary, release the holding brake and decouple from the mechanical system).

1. Switching on

The converter switches to status "Operation with position test" when the ON command is output and while parameter P115 = 8 or value 1 is pending on binector input "Source position test" (P549).

2. Setpoint

The user must set a current setpoint so that the motor aligns itself (e.g. 100 % at P260 "source M(set)" or 1 % speed setpoint, which also results in 100 % torque setpoint via the speed controller).

The current causes the free moving motor to align itself to the fixed electrical angle. Depending on the number of poles of the motor, this will be one of several possible mechanical angular positions within one revolution of the motor. (For example, for a six-pole motor there are three mechanical angular positions within one revolution into which the motor can lock. It is of no consequence for the adjustment into which of the three positions the motor locks.)

3. Encoder adjustment

The incorrect orientation of the encoder can be read off in mechanical degrees in r286 "position test angle" (function diagram 390). A value of $\pm 1^\circ$ mechanical for motors with a low number of poles (≤ 8 poles) lies within the range of measuring inaccuracy and need not be altered. (This degree of inaccuracy can even be induced by turning the motor shaft slightly by hand during measurement.)

When the encoder is replaced, correction should preferably be made mechanically, i.e. by disconnecting the encoder, turning it until the "position test angle" r286 shows the value 0° and then tightening it again so that the encoder is again installed in the position as set in the factory. If it is not completely safe to align the encoder mechanically during operation, the deviation can also be corrected by parameterizing an offset in P132 "angular offset" to a position test angle r286 of 0° . Making the correction in the parameters is especially recommended for motors of non-Siemens suppliers who install their encoders in a standard position other than that of SIEMENS. **During operation with position test**, it is possible to change parameter P132 "angular offset" manually with the position test, or reduce it automatically with a falling edge at binector input "source position test" (P549) by the current displayed value r286 "position test angle".

In both cases, the position test angle r286 must finally show the value 0° .

4. Checking the direction of rotation

Not only the angular position but also the direction of rotation of the encoder can be checked:

If the position test is performed during operation, the current pointer can be turned with bit 11 in control word 1 "enable positive direction of rotation" (see function diagram 180, P571 "Q.positive DR").

If the bit is switched from 0 to 1, the injected current pointer slowly turns to the right by one "electrical revolution" (approx. 1 to 2 s). This turns the motor one pole division to the right. Connector KK0186

("theta(I cont.)" function diagram 390) then turns one full revolution in the positive direction ($0\% > 100\% > +199\% / -200\% > -100\% > 0\%$). (If the bit is switched from 1 to 0, the procedure is reversed.)

If KK0186 does not turn a full revolution, either the parameterized number of pole pairs (P109) is incorrect (how far did the motor turn mechanically, does it correspond to the pole pair number?) or has the encoder been incorrectly parameterized (incorrect number of increments?).

If KK0186 completes a full revolution during the test but in the opposite direction, either a track has been reversed on the encoder or two phases have been reversed on the motor (→ correct on motor or on encoder and repeat point 3 "encoder adjustment").

5. Normal operation

NOTE

Do not forget to reset parameters P115, P260 or the binector at P549 so that normal current-controlled operation can continue!

NOTE about removing the encoder on SIEMENS motors

If an encoder or multiturn encoder is to be completely removed, a screw is required to force off the encoder! The encoder has a conical shaft end that is inserted into the motor shaft. Even after all fixing screws have been removed, the encoder shaft is usually fixed so tight in the motor shaft that the encoder can only be removed without damaging it by forcing it out with a special screw.

Depending on the design of the encoder shaft, different screws will be required for forcing off the encoder (see Fig. 7-13 and Fig. 7-14).

DANGER



If the encoder is adjusted incorrectly, the motor can overspeed.

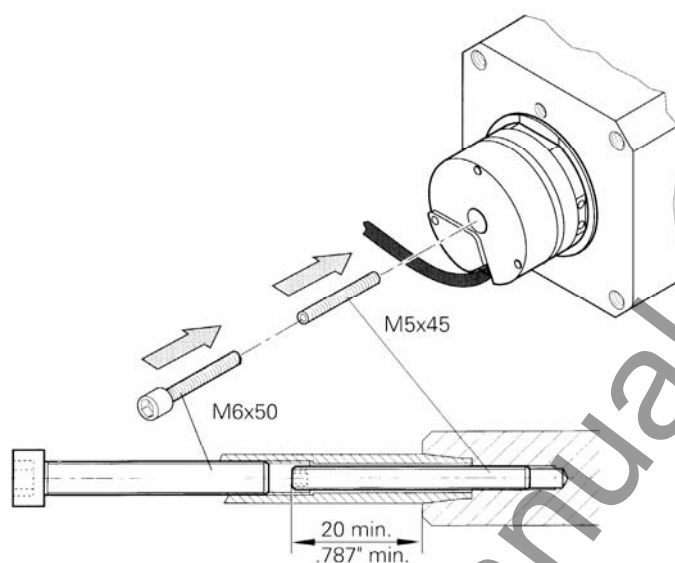


Fig. 7-13 Forcing off an encoder of old design

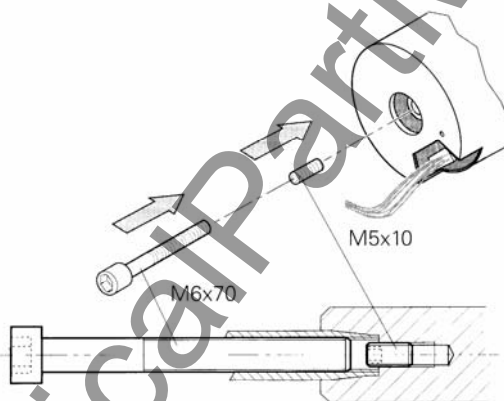


Fig. 7-14 Forcing off an encoder of new design

7.3.5 Function "PRBS signal with recording" (function diagram 796)

Recording

The free block "PRBS signal with recording" generates a pseudo random binary sequence which can be scaled with a selectable amplitude (U477 "PRBS-Ampl"). This signal is available on K0630. Connector K0630 supplies a noise signal with frequency components of 0.6 Hz to 625 Hz at a sampling frequency of 5 kHz (and recording of 2 single connectors). The noise signal is called "pseudo random" because even though the bit sequence of a cycle has no repetitions, the noise generator always supplies the same bit sequence on each start.

The block can record one or two channels simultaneously, physically using the trace memory in the converter. If the noise generator is assigned to the time slot (U953.70 = 2), the trace must be removed (U953.72 = 20), as both are supplied via the same connectors and both parameters and use the same memory.

Recording is always performed in T2. Unlike the normal trace, with the PRBS signal a series of recordings (U478 "PRBS cycles") can be started (U478 "PRBS cycles") and those recordings then averaged through addition. As a result, random interference is averaged out and even small noise amplitudes produce good results. In order to prevent the value range from immediately overflowing, the DC value is derived by an additional noise cycle that is automatically transmitted before the measurement itself. However, if a counter overflow occurs, alarm A032 is set. The alarm is only reset when a measurement is restarted. Monitoring parameter n479 "PRBS cycles CntD" counts the set cycles back to 0 during measurement. In this way, measuring progress is measured and the occurrence of alarm A032 can be detected, if necessary.

Reading out

As recording is configured and started using the same parameters (U480, U481, U488) as normal trace, recording can be configured and started in DriveMonitor with the Trace menu item. (Please first ensure that the time slot for noise is activated and the time slot for the trace block is deactivated! U953.72 = 20 and U953.70 = 2).

In the case of DriveMonitor, item "Trace" in menu "Diagnosis" must be selected. In the window "Settings.." you can then select the connector(s) to be recorded. No more than the first two channels can be activated. The "recording interval" or "trigger settings" bear no relevance to the PRBS recording. Recording starts as soon as you click on the "Start" button.

The recorded data are automatically stored in file "C:\Siemens\DriveMonitor\P7vrvisx\Projects\Drives\MASTERDRIVES MC\ TRACE.TXT" when read out with DriveMonitor. Existing files are overwritten. The file is an ASCII file with integer values and commas as separators, which can be imported into commercial mathematical programs.

Evaluation

Evaluation of the data in a mathematical program is **the task of the user**, i.e. the user must first create a worksheet in a mathematical program in which he or she edits, evaluates, and graphically represents the data as required.

One suitable program is, for example, the mathematical software "Mathcad [®]" produced by the company MathSoft (<http://www.mathsoft.com>), which provides a relatively uncomplicated introduction to the subject area.

The following graphics show the amplitude and phase response of the closed speed control loop (factory setting, without optimization). The data was evaluated and displayed graphically using "Mathcad 8 [®]".

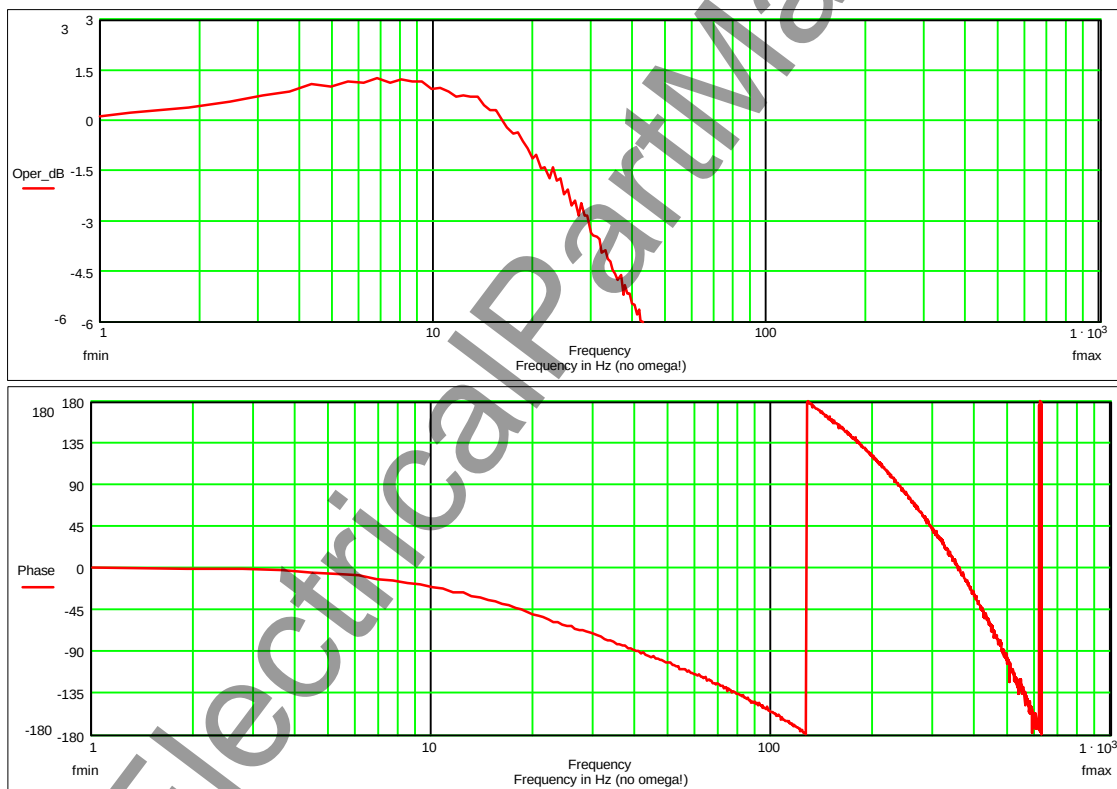


Fig. 7-15

7.3.6 Function "speed filter" (function diagram 361)

The free block "speed filter" contains three digital filters of the second order connected in series. They are always calculated in the time slot of the speed controller (T0) and can be wired freely. It is usual practice to insert the filter block in front of the speed controller (P252 = KK0152 and P228 = KK0158) when the PI controller is operated as a speed controller (P238 = 0 (default)) or downstream of the speed controller (P252 = K0153 and P260 = KK0158) when the PIR controller is employed as a speed controller (P238 = 1). The advantage of this is that vibrations that are injected by means of the speed setpoint (e.g. position controller output) are also acquired.

To prevent further deadtimes from resulting in the speed control loop in this configuration, the arithmetic sequence of the individual blocks must be changed such that the filter block is calculated after the setpoint/actual value difference has been generated. This can be set in parameter U963 (arithmetic sequence): U963.42 = 5 (filter), U963.43 = 2 (smoothing elements), U963.45 = 3 (setpoint/actual value difference generation).

Application and system analysis

It is possible to improve the stability of the control loop using filters. If the mechanical transmission elements contain disturbing frequencies they can be removed with the low-pass filters or band-stop filters. However, it is important to note that these filters increase the sum of the small delay times and the substitute delay time of the speed control loop and can therefore not be used in all cases.

First of all, the natural frequencies of the system must be determined. The simplest way of doing this is to inject or deactivate a torque setpoint in steps. The natural vibration of the system overlays the progression of the speed actual value after the step changes. These vibrations can be induced by triggering "AUS2" during a speed controlled acceleration and therefore deactivating the torque suddenly. If the vibrations are to be induced by injecting a square-wave additional torque setpoint, the speed controller must be set very slowly so that it does not intervene to compensate.

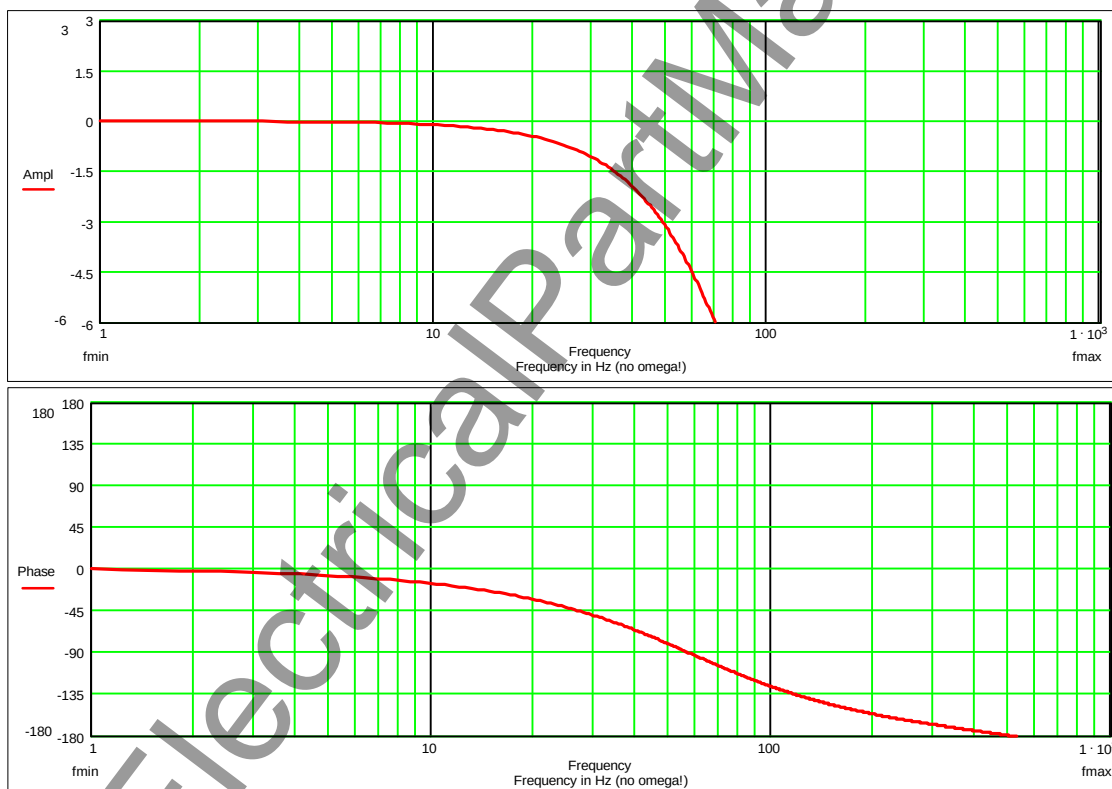
A good overview of the vibrations is provided by recording the speed actual value for a whole series of different speed actual values. The frequency spectra of the recordings are represented in a staggered 3D arrangement in a cascade diagram. In such a diagram it is possible to differentiate between the speed-dependent harmonic components and the natural frequencies of the system. The speed-dependent harmonic components which, for example, result from the unbalance or eccentricity appear as straight lines forming the origin in the cascade diagram. The natural frequencies of the system appear in the diagram as lines of constant frequency.

An examination of the transmission function of the open speed control loop (e.g. using the built-in noise excitation, function diagram 796) provides information about the amplitude and phase margin of the control loop for critical frequencies. As the filters always change the phase progression it is also possible to change the phase margin specifically by applying filters. For example, the phase of frequencies above the blocking frequency is raised by a band-stop filter, which can be utilized to increase the phase margin.

As low natural frequencies always have a detrimental effect on the dynamics and therefore fault suppression of the system, it is worth examining the source of natural frequencies in the mechanical system. By measuring the transmission functions with additional encoders, the measured natural frequencies can be assigned to the mechanical components and their frequencies increased specifically by using more rigid components or lighter masses.

Example 1

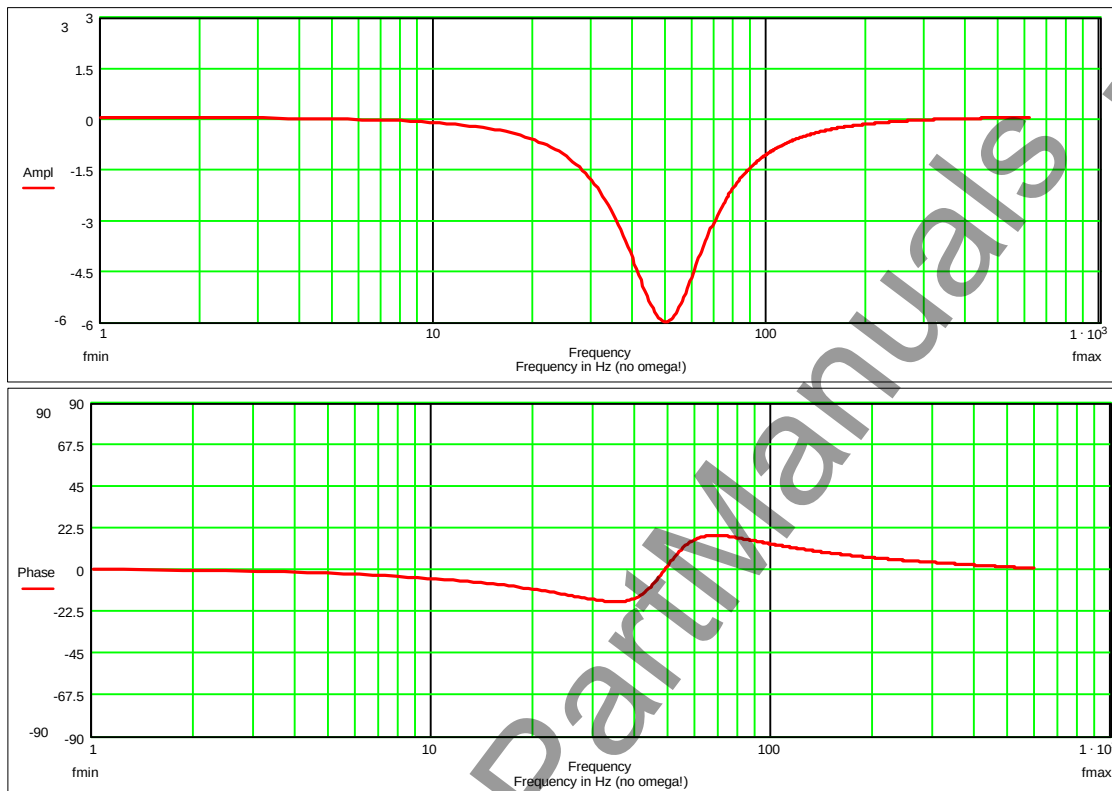
Low pass (P256 = 2; P254 = 50 Hz)



Example 2

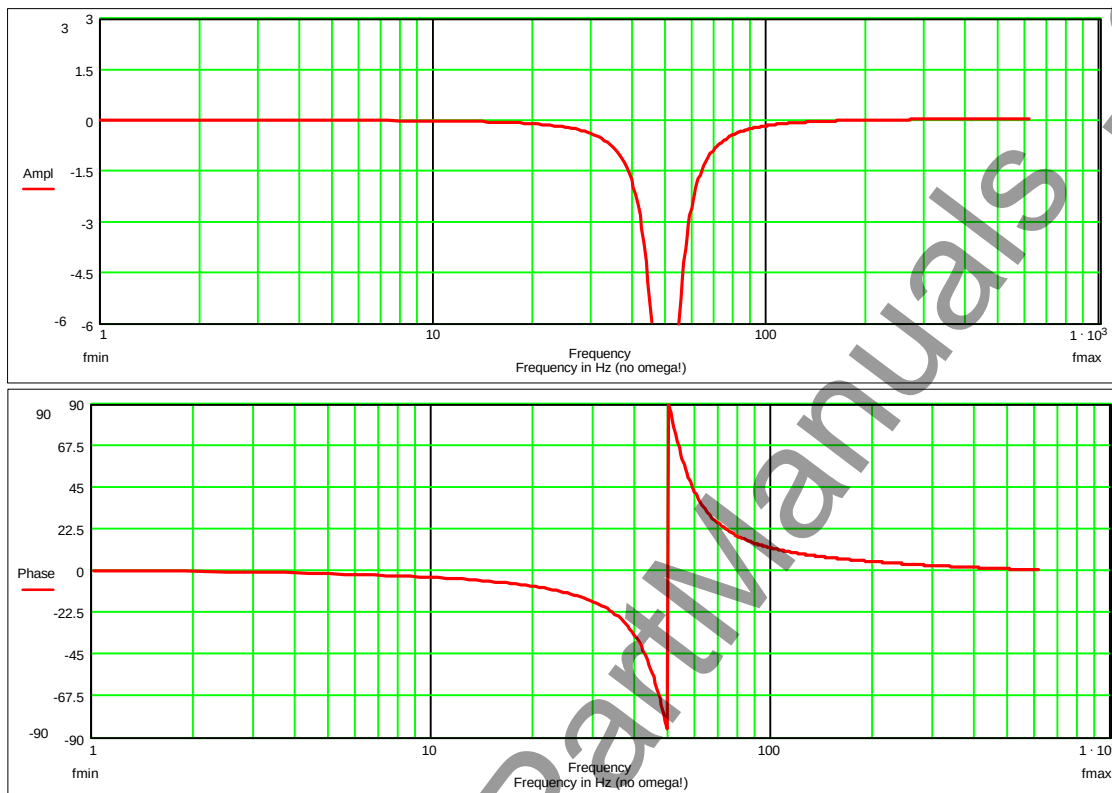
Band-stop filter of average quality and semi-suppression of the resonant frequency

(P256 = 1; P254 = 50 Hz; P253 = 1.0; P257 = 50 %)



Example 3

Band-stop filter with full suppression and high quality:
(P256 = 1; P254 = 50 Hz; P253 = 3.0; P257 = 0 %)



7.3.7 "Speed controller characteristic" function (function diagram 360)

You can select the speed controller characteristic via parameter P238.

P238 = 0: PI controller (default)

The speed controller is optimized in accordance with the known rules, e.g. Symmetrical Optimum.

While the controller is being optimized, e.g. according to the Symmetrical Optimum for a good fault response, overshoots occur in the response to setpoint changes. This should be reduced by appropriate setpoint smoothing (e.g. P221) or using the reference model (P238 = 1).

P238 = 1: PIR controller (reference model for the I-action component)

The response to setpoints by the speed controller can be improved (overshoot reduction) by means of the PIR controller characteristic (reference model). This is conditional on the controller being set according to PI controller conditions (see above P238 = 0). For a PIR controller (P238 = 1), the time constant of the reference model (P239) for the PIR controller must also be adjusted such that, for example, the smallest possible overshoot occurs in response to a setpoint step change.

Plant conditions permitting, you can do this in the following way: Set TN (P240) to a value of 0 (make note of original value!) and trace K0155 as the setpoint is changed, the time constant (P239) must be adjusted such that the areas above and below the zero line of K0155 are approximately equal in size. Then reset TN (P240) to its original value.

References relevant to reference model:

"Electrical Feed Drives in Automation"

SIEMENS AG; H.Groß, J.Hamann, G.Wiegärtner

(ISBN: 3-89578-058-8)

NOTES

- ◆ When you use the reference model (P238 = 1), the source selected in P228 (Src n (deviation)) is inoperative; KK0152 is linked permanently to the speed controller internally (as per default setting for P228).
- ◆ When you use the reference model (P238 = 1), the sampling time / sampling sequence of the speed controller itself is determined via U953.45 / U963.45 (not just the speed controller total as per P238 = 0); i.e. the sampling time / sampling sequence set via n959.52 / n969.52 is inoperative in this case.

7.3.8 "Field weakening for synchronous motor" function (function diagram 389)

The magnetic flux is constant in the base speed range of the permanent-field synchronous motor. The voltage requirement increases in proportion to speed until it exceeds the maximum converter output voltage.

In the field weakening range, the magnetic flux must be reduced in proportion to speed through application of a magnetic field which opposes the permanent magnet field.

In the field weakening range, an internal counter-voltage proportional to speed must be generated to weaken the effect of the permanent magnetic field on the terminal voltage (EMF).

NOTE

If the closed-loop control or power section of the synchronous motor fails at high speed, the motor regenerates power to the DC link at high voltage. To prevent damage to the converter caused by overvoltage under fault conditions, the VPM (Voltage Protection Module) can be used as an overvoltage protection mechanism. The VPM detects an excessive motor terminal voltage and shorts the three motor cables.

The regeneration voltage of the synchronous motor can be defined by the voltage constant (k_E) specified in the Engineering Information.

If $k_E \cdot \frac{n_{\max}}{1000} \cdot \sqrt{2} < 800 \text{ V}$, there is no need to provide overvoltage protection.

Commissioning the field weakening function

The motor and encoder data must first be entered in the "Drive Setting" menu (see Section 6.2.3 "Drive Setting"). The "Field weakening for synchronous motor" function is activated in parameter P300 SelectFieldWeak. A useful assignment of the field weakening activation speed P299 and the motor short-circuit current P105 MotPwrFactor can be obtained through the automatic parameterization run (P115 = 1). Both parameters define the flux setpoint control in the field weakening range. If the exact setting values of parameters P299 and P105 are known, they can, of course, be set manually.

IMPORTANT

The motor short-circuit is significantly higher than the rated motor current. Adjust parameter P128 Max Current accordingly.

Adjust the reference variables P350 ... P354 for field weakening operation.

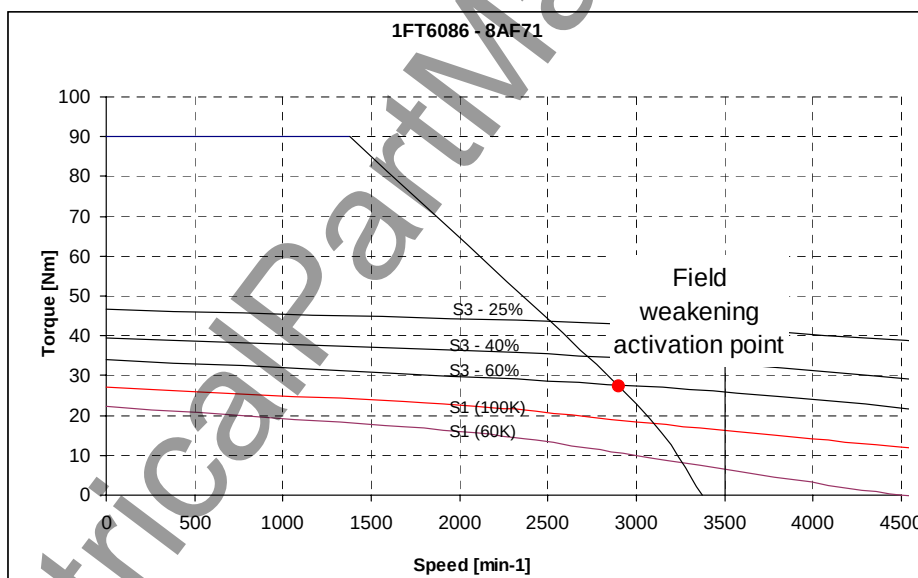
Manual optimization

When the parameters relevant to field weakening have been set via automatic parameterization, they do not need to be adjusted again for most applications. The default settings ensure that the motor enters the field weakening range at speeds above rated value. The terminal voltage is kept constant under no load in the field weakening range. The maximum torque decreases in proportion to $1/n$, the maximum output remains constant.

The settings can be optimized manually for special applications. If, for example, the required maximum torque is lower than the torque obtained with the default setting, the speed at which field weakening commences can be raised to reduce the thermal loading of the motor by the field weakening current.

How to optimize the settings:

1. Determine the speed at which field weakening commences on the basis of the desired torque value and the limit characteristics field from the Engineering Information:

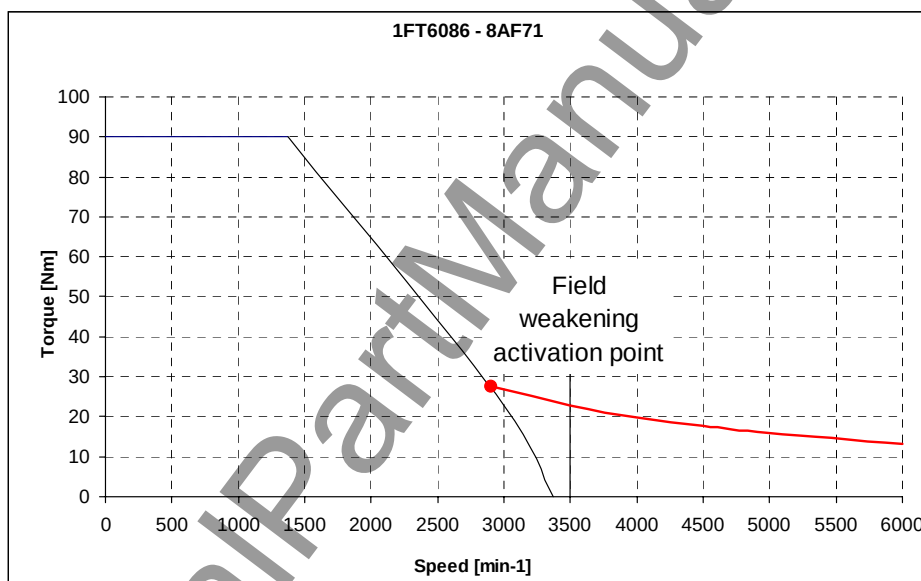


e.g. 1FT6086-8AF71, 27.5 Nm, 2900 rev/min

Enter the calculated field weakening activation speed in P299.

2. Set the characteristic to meet requirement "Maximum output in field weakening range remains constant":

The output voltage of the converter r003 under no-load conditions at the field weakening activation speed is recorded. With field weakening operation selected ($P300 = 1$), the motor must be made to approach the end value of the desired field weakening range. The output voltage at no load must be the same as the voltage at the field weakening activation speed. The voltage in the field weakening range can be varied by means of the short-circuit current ($P105$). The voltage drops as the short-circuit current increases and vice versa. The short-circuit current programmed by the automatic parameterization run corresponds to the application illustrated here which means that only minor adjustments are required.



e.g. 1FT6086-8AF71, 27.5 Nm at 2900 rev/min, 13.3 Nm at 6000 rev/min, $U_{(\text{no load } 2900 \text{ to } 6000)} = 296 \text{ V}$, $I_k(P105) = 37 \text{ A}$

3. Set the characteristic to meet requirement "Maximum torque in field weakening range remains constant":

The output voltage (r003) at the field weakening activation speed under no load conditions must be measured. The output voltage at the end point of the desired field weakening range must then be calculated according to the following formula from the maximum converter output voltage (depends on converter supply voltage) and the calculated output voltage at the field weakening activation speed:

$$U_{\text{Conv Out nEnd}} = \sqrt{U_{\text{Conv max}}^2 \cdot (1 - k_f^2) + U_{\text{Conv Out nAct}}^2 \cdot k_f^2}$$

where

$$U_{\text{Conv max}} = 350 \text{ V (when UDCL = 540 V)}$$

$U_{\text{Conv Out nEnd}}$ = Output voltage at end of field weakening range

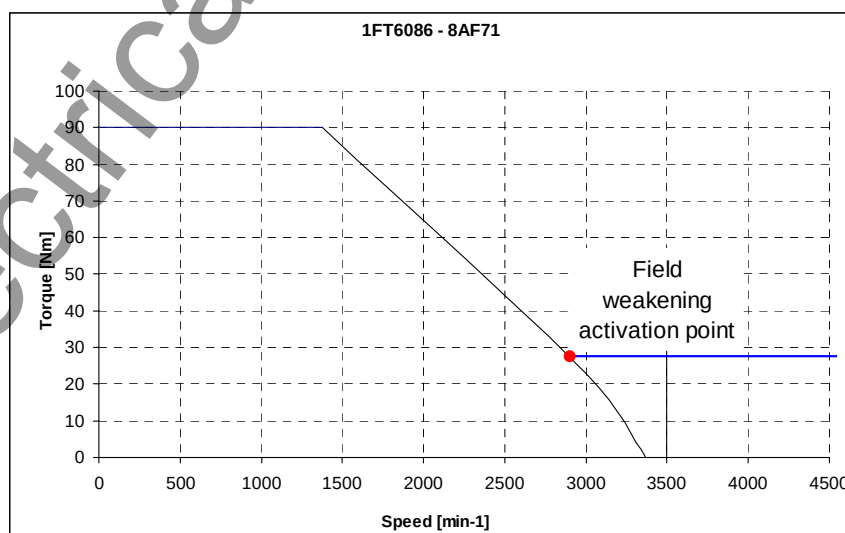
$U_{\text{Conv Out nAct}}$ = Output voltage at field weakening activation speed

$$k_f = \frac{n_{\text{End}}}{n_{\text{Act}}} \quad \text{Field weakening range : Ratio between final field}$$

weakening speed and field weakening activation speed

The formula is based on the simplified principle that the voltage drop across the winding resistance of the motor is negligible in the field weakening range. This applies to series 1FT6 and 1FK7 servo motors.

The output voltage calculated by this formula must then be set at the end speed of the field weakening range at no load via the short-circuit current. The current setting must be increased in relation to the automatically calculated value, as the automatic parameterization run defines the short-circuit current for operation at constant maximum output.



e.g. 1FT6086-8AF71, 27.5 Nm at 2900 rev/min to 4500 rev/min,
 $U_{(\text{no load } 2900)} = 296 \text{ V}$, $U_{(\text{no load } 4500)} = 174 \text{ V}$ $I_k (\text{P105}) = 60 \text{ A}$

NOTE

The maximum current limit of the motor and the converter restricts even brief periods of operation in the field weakening range. If the sum of squares I_q and I_d exceeds the maximum permissible current, the motor cannot operate according to the calculated characteristic.

Torque accuracy

The magnetization of the permanent magnets can vary as a result of manufacturing tolerances and temperature fluctuations. In order to improve the absolute torque accuracy in the control of synchronous motors, it is essential to use the "kT estimator" (function diagram 393). However, the kT estimator can be used meaningfully only in conjunction with the friction characteristic (function diagram 399) as it is capable of correctly only the inner torque of the machine. Friction losses must be compensated via an additional torque from the friction characteristic.

7.3.9 Vdmax closed-loop control (function diagram 610)

Description

The Vdmax closed-loop control function allows briefly occurring regenerative loading to be handled without the unit shutting down with fault message F006 "DC link overvoltage". In this case, the frequency is controlled (closed-loop) so that the motor does not excessively enter over-synchronous operation.

For a steady-state load, the converter output frequency must increase.

If regenerative loading occurs when the machine is decelerating too quickly (P464), this is automatically reduced so that the converter is operated at the voltage limit.

Function diagram 610 shows how the Vdmax closed-loop control function operates.

The Vdmax closed-loop control is also optimally suited for regenerative operation, which can occur when the speed stabilizes at the end of ramp-up.

Parameters for setting the Vdmax closed-loop control

P515 Vdmax controller

Value range 0 and 1

0: The Vdmax controller is inhibited.

1: The Vdmax controller is released.

P516 Dynamic response of the Vdmax controller

Value range 0 % to 200 %

This parameter enables the response of the PID controller to be influenced.

At 0 %, the Vdmax controller is shut down.

The factory setting is 25 %.

The controller output can be visualized via connector K0271.

The activation point for the Vdmax controller can be parameterized in P517.

7.3.10 Harmonic compensation

Description

This function is designed to compensate for torque ripple which has a functional correlation with an angular signal. This signal can be defined, for example, by the rotor position, the electrical angle or the angle of the external encoder. Torque ripple can also occur as a result of the specific design of a motor.

For torque ripple to be reduced effectively with this function, the ripple must exhibit the following properties:

- ◆ Frequency of the ripple must be a multiple of the input signal frequency
- ◆ Fourier decomposition of the torque ripple signal must indicate a dominant harmonic

This function is capable of compensating for two of the dominant harmonics identified by Fourier decomposition.

For this purpose, a signal in phase opposition is injected in the torque setpoint channel. The time response of the compensation signal is configured by means of the following parameters.

Parameters

P309: Order of harmonic to be compensated (referred to electric fundamental wave)

(Index 1 – 1st harmonic, index 2 – 2nd harmonic)

P310: Phase angle of harmonic (sine) referred to the start point of the fundamental wave (sine) in phase U with a clockwise phase sequence

(Index 1 – 1st harmonic, index 2 – 2nd harmonic)

The time response of the compensation signal defined by these parameters is mapped at connector K0272.

The severity of the torque ripple can vary as a function of the motor current. The variation in the compensation signal amplitude can be quantified in the two parameters below using a square-law growth function.

P311: Maximum current, i.e. current with maximum torque ripple

P312: R.m.s. value of compensation signal for

P312.1 $I_{\text{tot}} = 0 \% * P311$

P312.2 $I_{\text{tot}} = 50 \% * P311$

P312.3 $I_{\text{tot}} = 100 \% * P311$

(indices 4-6 analogous for 2nd harmonic)

A square-law interpolation calculation is performed between the three interpolation points. The signal value of the resulting compensation signal is mapped at connector K0273.

The compensation signal is injected as current upstream of the current limiter. This means that the configuration must allow for the functional correlations between torque ripple and compensation current.

The harmonic compensation function block is described in FD630.

7.4 Special functions

7.4.1 Loading firmware

The firmware supplied in the units is stored non-volatily in electrically erasable memory chips, so-called flash EPROMs. If required, the firmware can be erased and overwritten with new firmware.

It is necessary to import new firmware if

- ◆ an extended function scope is available in a new firmware version and this needs to be used, or if
- ◆ user-specific firmware needs to be loaded into the units.

The firmware can be loaded using a laptop or PC and the data is transferred into the units via the serial interface SCom or SCom1. A special cable is necessary for importing the firmware.

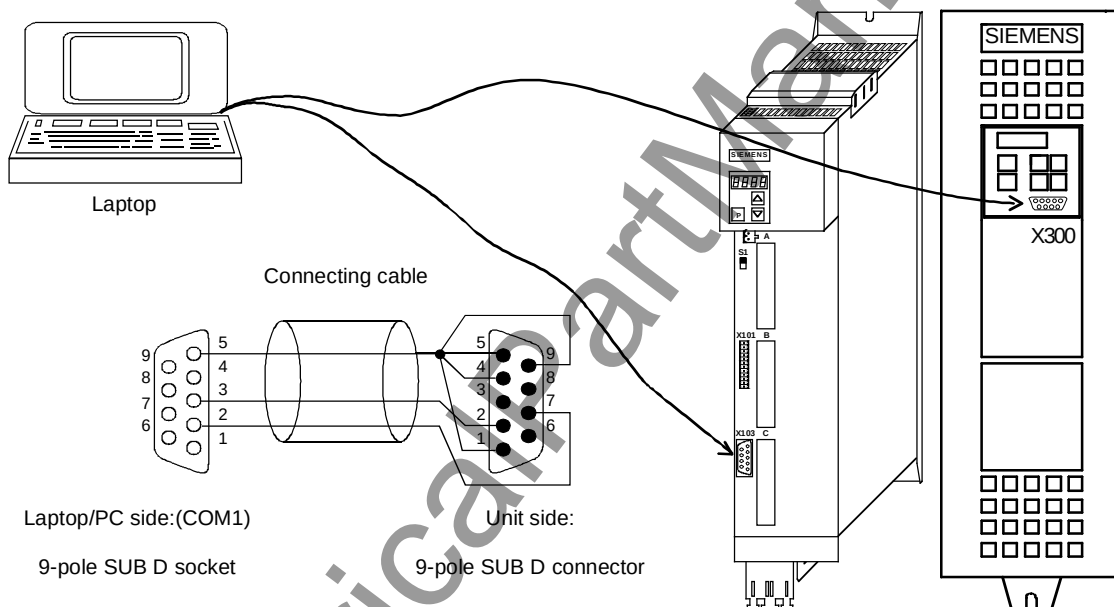


Fig. 7-16 Loading firmware by means of laptop or PC

Loading the firmware from a laptop/PC entails the steps listed below. All other programs using the same PC port (COM1 or COM2) (e.g. DriveMonitor) must be closed from the outset.

If problems should occur with the loading program bsl.exe under Windows NT, the program Win BSL.exe is also available after installation of DriveMonitor (Path: \Siemens\DriveMonitor\ P7vrvix\SYSTEM\WINBSL).

Before you load the software, save your parameter settings (upread with OP1S, or upload with DriveMonitor)!

If the position after the decimal point of the firmware changes (e.g. when upgrading from 1.3x to 1.4x), the module is reset in full. A CUPM signals status °000 and waits for the power section definition (input of the converter ID in P070).

The following parameters are read and stored when upload from DriveMonitor, but not written to the unit when downloaded for safety reasons:

Parameter number	Parameter name
P060	Menu selection
P070	Order No. 6SE70..
P072	Conv.current(s)
P073	Conv.power(s)
P700	IF bus address
P701	IF baudrate
P702	IF no. of PKW
P703	IF no. of PZD
P918	CB bus address
P952	Number of faults
P970	Factory setting
P971	EEPROM accept.
U976	Product number
U977	PIN

Table 7-6 Parameters that cannot be changed on downloading with DriveMonitor

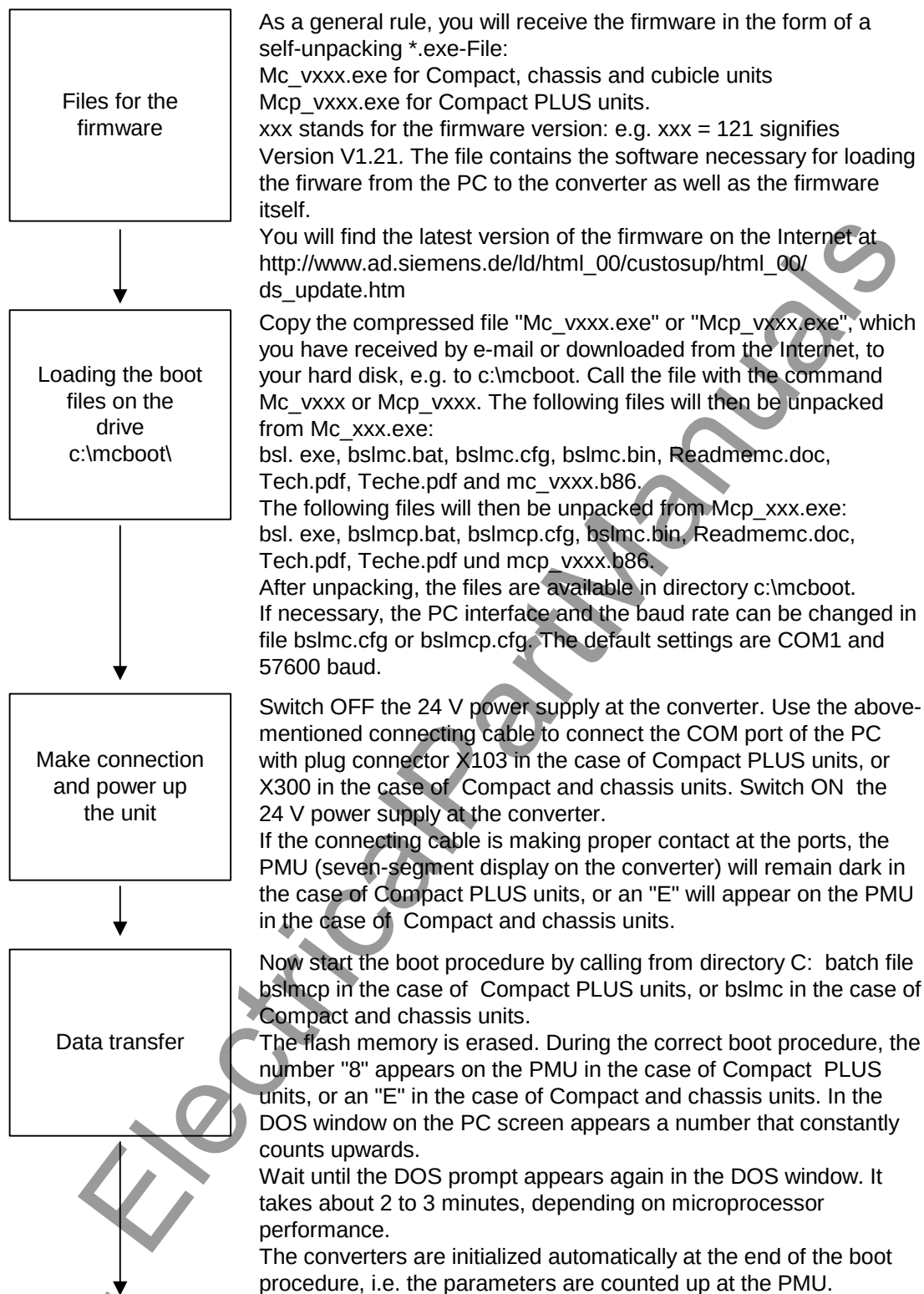
Where necessary, these parameters must be parameterized separately, for example, via the parameter menu of DriveMonitor; via a script file, or directly on the converter. The values of the specified parameters can be viewed by opening the backed-up download file in DriveMonitor "offline".

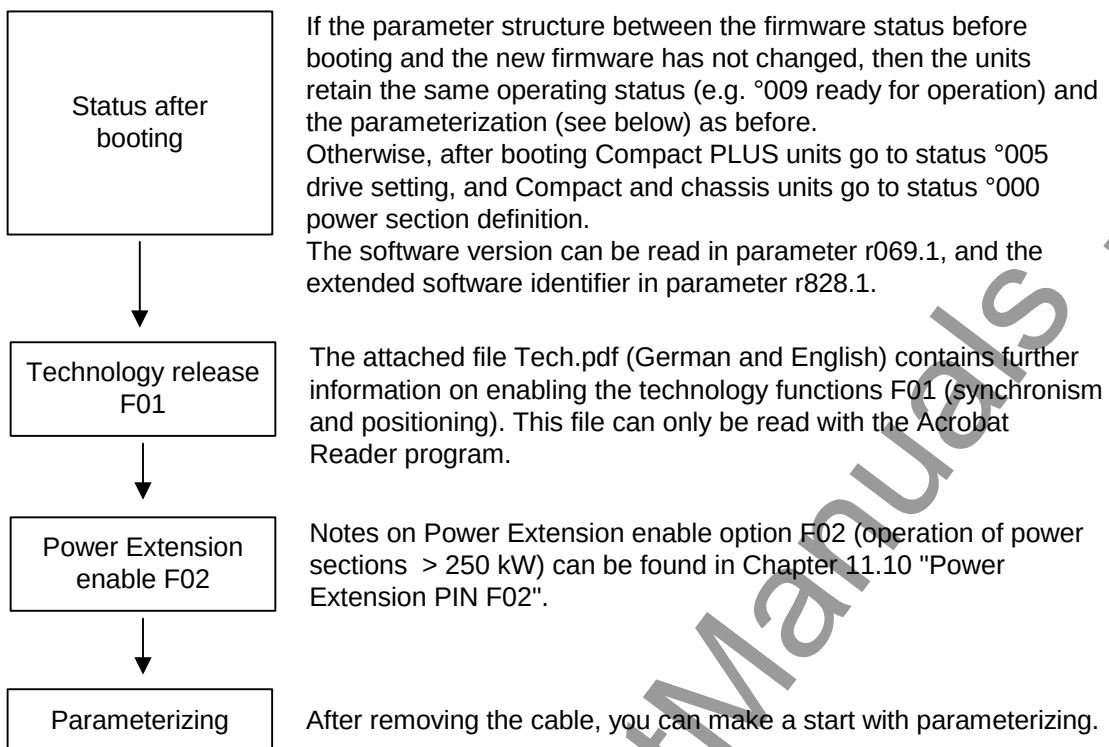
When **upgrading a MASTERDRIVES MC to a current firmware version**, we make a distinction between 2 cases:

1. Only the third position of the version number changes (e.g. from V2.00 to V2.01). In this case, the firmware can be loaded directly into the unit. The parameterization remains the same.
2. The second position of the version number changes (the third position is then of no significance, e.g. from V2.02 to 2.10). In this case, the parameterization must be saved with DriveMonitor. It is important to remember to use the differences from the factory setting **only** (under DriveMonitor: File, Upload, Base unit: changes only...).

The new firmware is then loaded into the unit. When initialization is complete, all parameters are reset to the factory setting. Now the file previously generated can be loaded back into the unit with DriveMonitor (under File, Download).

The first and second position of the firmware version can be read out in r069.01 and the third position corresponds to the first position after the decimal point in r828.01=0.1 (e.g. V2.01 corresponds to r69.01=2.0 and r828.01=0.1).





8 Communication

A differentiated communication concept makes it possible to use the correct communication medium for a specific requirement. The following communication interfaces are available:

- ◆ Integrated serial interface(s) with USS protocol for parameterization, operator control and visualization of the units with OP1S or PC
- ◆ Optional boards for various field bus interfaces (e.g. PROFIBUS DP) for integration into the automation
- ◆ Optional board for connecting up SIMOLINK for fast synchronous data transfer between technologically connected drives (e.g. angular synchronism).

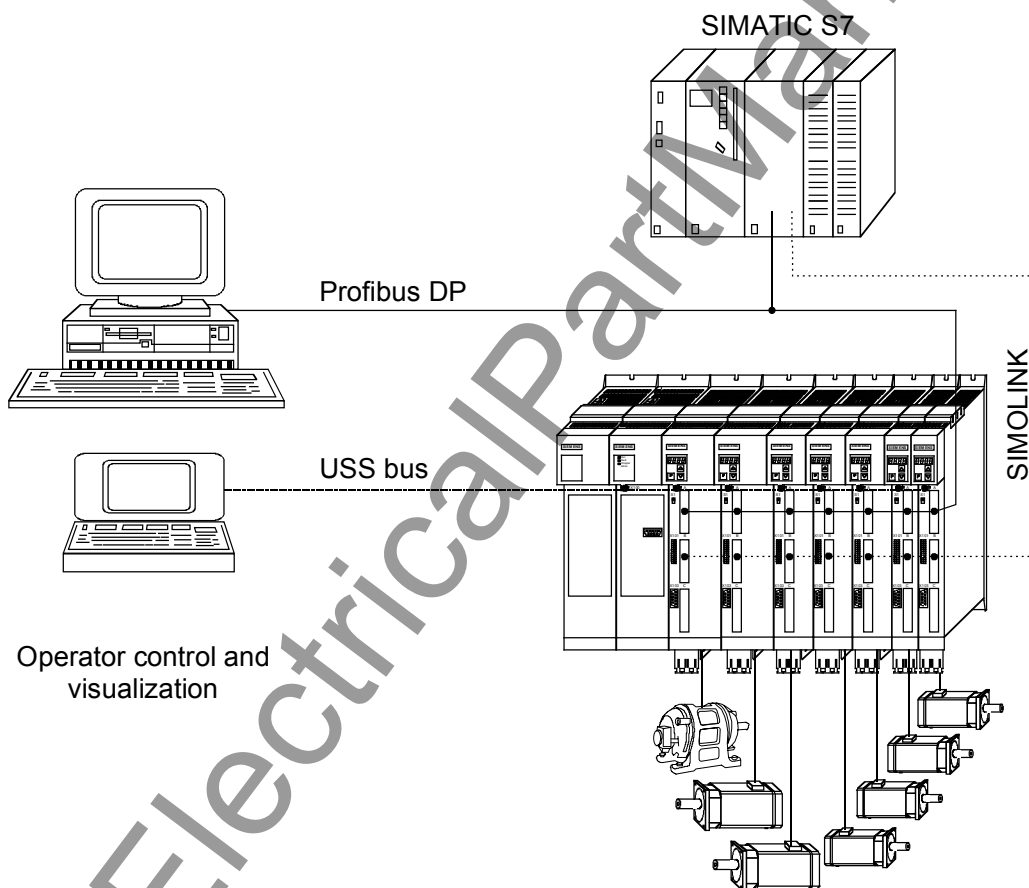


Fig. 8-1

Overview for communication

8.1 Universal Serial Interface (USS)

Introduction

This documentation describes the application of the Universal Serial Interface Protocol (USS) for SIMOVERT MASTERDRIVES MC and VC.

NOTE

The USS protocol is a simple serial data transfer protocol, defined by Siemens AG, which is fully tailored to the requirements of drive technology. A detailed description of the protocol specifications, the physical interface, the bus structure as well as a definition of the transferred net data for drive applications are documented in the specification "Universal serial interface protocol USS® protocol" (Order No. E20125-D0001-S302-A1).

Using the USS protocol, a user can establish a serial bus link between a higher-level master system and several slave systems. Master systems can be, for example, PLCs or PCs. SIMOVERT MASTERDRIVES drive converters are always the slaves on the bus system. Furthermore, SIMOVERT MicroMaster, SIMOVERT P 6SE21 and 6RA23 and 6RA24 drive converters can be operated as slaves on the USS bus.

The USS protocol allows the user to implement both automation tasks with cyclical telegram traffic (→ a fixed telegram length is necessary) as well as visualization tasks. In this case, the protocol with variable telegram length is advantageous, as texts and parameter descriptions can be transferred in one telegram without chopping up the information.

8.1.1 Protocol specification and bus structure

Features

The USS protocol has the following significant features:

- ◆ Supports a multi-point-capable link, e.g. EIA RS 485 hardware or a point-to-point link, e.g. EIA RS 232.
- ◆ Master-slave access technique
- ◆ Single-master system
- ◆ Maximum 32 nodes (max. 31 slaves)
- ◆ Operation with variable or fixed telegram length
- ◆ Simple, reliable telegram frames
- ◆ The same bus mode of operation as with the PROFIBUS (DIN 19245 Part 1)
- ◆ Data interface to the basic unit according to PROFILE variable-speed drives. This means that, when the USS is being used, information is transferred to the drive in the same way as with the PROFIBUS-DP.
- ◆ Can be used for start-up, service and automation
- ◆ PC-based service tools (e.g. DriveMonitor) for SIMOREG and SIMOVERT
- ◆ Can be easily implemented in customized systems

8.1.1.1 Protocol specification

Introduction

The USS protocol defines an access technique according to the master-slave principle for communications via a serial bus. The point-to-point link is included as a sub-quantity.

One master and a maximum of 31 slaves can be connected to the bus. The individual slaves are selected by the master using an address character in the telegram. A slave can never transmit without first being initiated by the master so that direct information transfer between individual slaves is not possible. Communication takes place in the half-duplex mode.

The master function cannot be transferred (single-master system).

The following illustration shows a bus configuration using drive technology as an example.

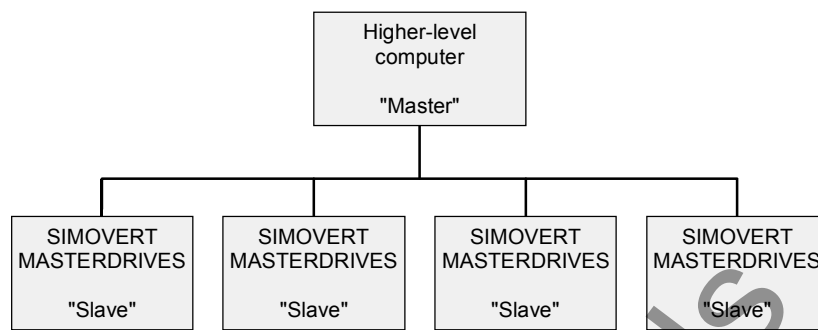


Fig. 8.1-1 Serial linking of SIMOREG/SIMOVERT drive converter (slaves) with a higher-level computer as the master

Telegram structure

Each telegram begins with the start character STX (= 02 hex), followed by the length information (LGE) and the address byte (ADR). The net characters then follow. The telegram is terminated by the BCC (Block Check Character).

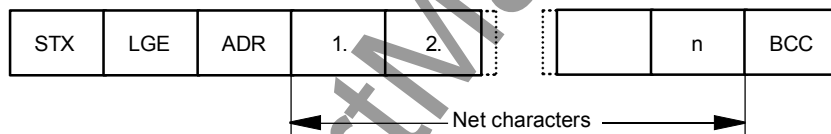


Fig. 8.1-2 Telegram structure

For single-word data (16 bit) in the net data block (= net character block), the high byte (first character) is always sent and then the low byte (second character). The same applies to double-word data: the high word is sent first followed by the low word.

The protocol does not identify tasks in the net characters. The contents of the net data for SIMOVERT MASTERDRIVES drive converters is dealt with in Section 8.1.3.

Data coding

Information is coded as follows:

- ◆ STX (start of text)
ASCII characters: 02 hexadecimal
- ◆ LGE (telegram length)
1 byte, contains the telegram length
- ◆ ADR (address byte)
1 byte, contains the slave address and the telegram type (binary coded)
- ◆ Net characters
Each one byte, contents are task-dependent
- ◆ BCC
1 byte, Block Check Character

Assigning the address byte (ADR)

In the address byte, information other than the node number is coded:
The individual bits in the address byte are assigned as follows:

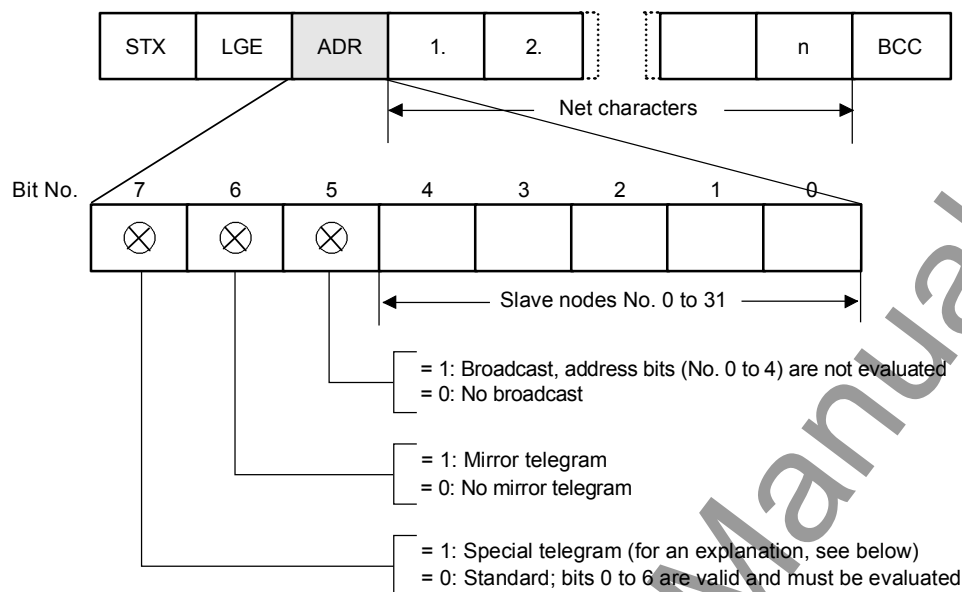


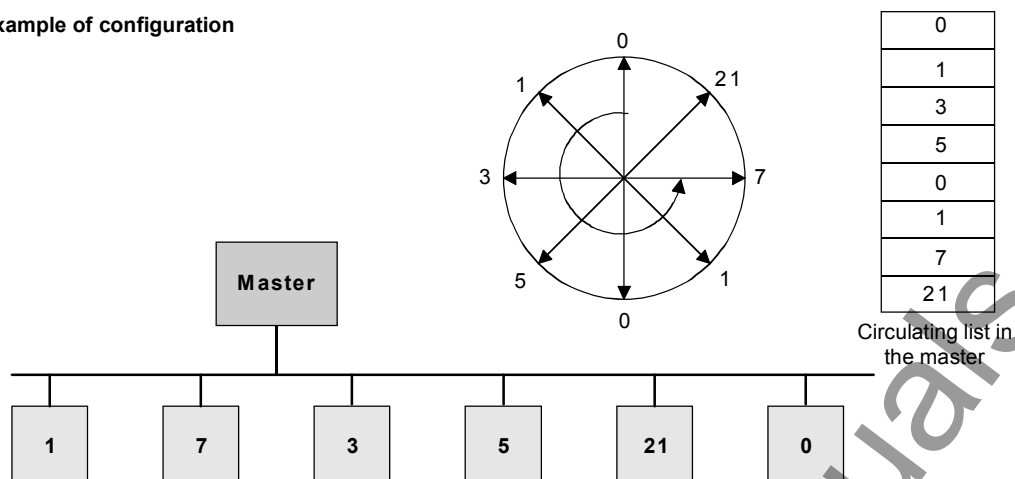
Fig. 8.1-3 Assignment of the address byte (ADR)

Data transfer procedure

The master ensures cyclical telegram data transfer. The master addresses all of the slave nodes one after the other with a task telegram. The addressed nodes respond with a reply telegram. In accordance with the master-slave procedure, the slave, after receiving the task telegram, must send the reply telegram to the master before the master can address the next slave node.

Handling data transfer

The sequence of the addressed slave nodes can be specified, for example, by entering the node numbers (ADR) in a circulating list (polling list) in the master. If it is necessary to address several slaves in a faster cycle than the other slaves, their node number can occur several times in the circulating list. A point-to-point link can be implemented by means of the circulating list, in which case only one node is entered into the circulating list.

Example of configuration

SIMOVERT MASTERDRIVES with the addresses 0, 1, 3, 5, 7 and 21
 Nodes 0 and 1 are signalled twice as often as others

Fig. 8.1-4 Circulating list

Cycle time

The length of a cycle time is determined by the time needed for the sequential occurrence of data exchange with the individual nodes.

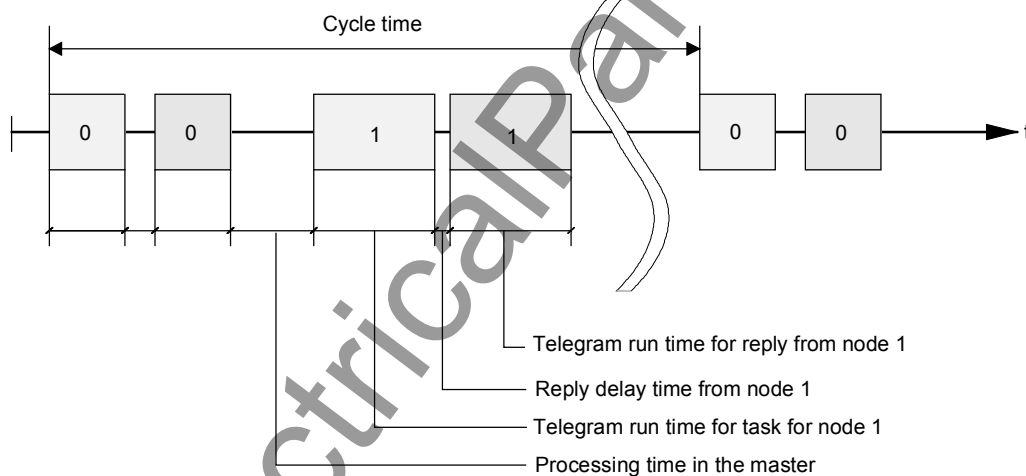


Fig. 8.1-5 Cycle time

Due to inconstant reply delay and processing times, the cycle time is not fixed.

Start interval

The STX start character (= 02 hexadecimal) by itself is not sufficient for the slaves to clearly identify the start of a telegram because the bit combination 02/hexadecimal can also occur in the net characters. For this reason, a no-character start interval of at least 2 character run-times before the STX is specified for the master. The start interval is part of the task telegram.

Baud rate in bit/s	Start interval in ms
9600	2,30 ms
19200	1,15 ms
38400	0,58 ms
76800	0,29 ms
93750	0,23 ms
187500	0,12 ms

Table 8.1-1 Minimum start intervals for various baud rates

Only an STX with a preceding start interval identifies the valid start of a telegram.

Data is always transferred in accordance with the diagram illustrated below (half-duplex mode):

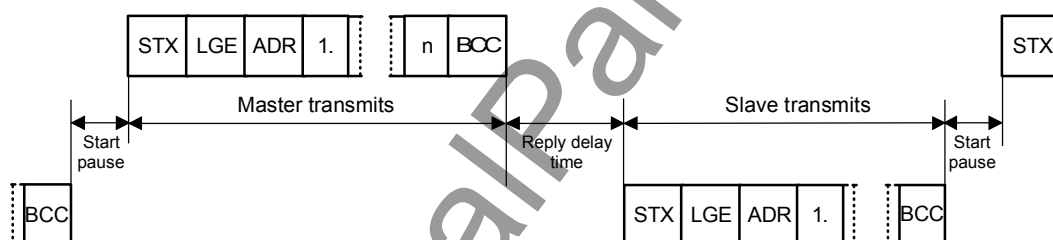


Fig. 8.1-6 Transmit sequence

Reply delay time

The time interval between the last character of the task telegram (BCC) and the start of the reply telegram (STX) is known as the **reply delay time**. The maximum permissible reply delay time is **20 ms, but it must not be less than the start interval**. If node x does not respond within the maximum permissible reply delay time, an error message is deposited in the master.

The master then sends the telegram for the next slave node.

8.1.1.2 Bus structure

The data transfer medium and the physical bus interface are essentially determined by what the bus system is used for.

The physical interface of the USS protocol is based on the "Recommended Standard RS-485". For point-to-point links, a sub-quantity of EIA RS-232 (CCITT V.24), TTY (20 mA current loop) or fiber-optic cables can be used as the physical interface.

The interfaces for SIMOVERT MASTERDRIVES are always RS 485 with 2-wire cable.

Exception: Either RS 485 or RS 232 can be connected at the 9-pin SUB D socket connector on the PMU (operator control and parameterizing unit) of the basic units.

NOTICE

This section describes how a USS field bus has to be structured in order to ensure reliable data transfer via the transfer medium in standard applications. Under special conditions of use, additional factors must be taken into account which require further measures or restrictions that are not described in this document.

Topology

The USS bus is based on a linear topology without branches.

Both ends of the line terminate at a node.

The maximum cable length and therefore the maximum distance between the master and the last slave is limited by the characteristics of the cable, the ambient conditions and the data transfer rate. With a data transfer rate of < 100 kbit/s, a maximum length of 1200 m is possible.

The number of nodes is limited to a maximum of 33 (1 master, 32 slaves).

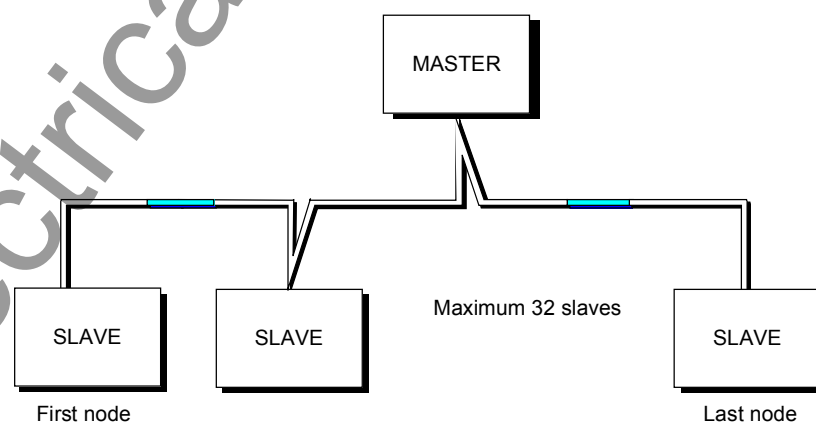


Fig. 8.1-7 USS bus topology

The two ends of a bus line (first node and last node) must be terminated with bus terminating networks.
Point-to-point connections are handled just like bus connections. One node has the master function and the other has the slave function.

Data transfer technology

Data is transferred in accordance with Standard EIA 485. RS 232 can be used for point-to-point links. Data transfer is always half-duplex – i.e. alternating between transmitting and receiving – and it must be controlled by the software. The half-duplex technique allows the same cables to be used for both data-transfer directions. This permits simple and inexpensive bus cabling, operation in environments subject to interference and a high data transfer rate.

Cable characteristics

A shielded, twisted two-wire cable is used as the bus cable.

Conductor diameter Ø	$2 \times \approx 0,5 \text{ mm}^2$
Conductor	$\geq 16 \times \leq 0,2 \text{ mm}$
Lay ratio	$\geq 20 \text{ twists / m}$
Overall shield	Braided, tin-plated copper wire, diameter Ø $\geq 1,1 \text{ mm}^2$ 85 % optical coverage
Overall diameter Ø	$\geq 5 \text{ mm}$
External sheath	Depending on the requirements regarding flame retardation, deposits after burning etc.

Table 8.1-2 Structural data

NOTE

All information should only be considered as a recommendation. Deviations or different measures may be required depending on the particular requirements, the specific application and the conditions on site.

Thermal and electrical characteristics

Cable resistance (20°C)	$\leq 40 \text{ } \Omega/\text{km}$
Insulation resistance (20°C)	$\geq 200 \text{ M}\Omega/\text{km}$
Operating voltage (20°C)	$\geq 300 \text{ V}$
Test voltage (20°C)	$\geq 1500 \text{ V}$
Temperature range	$-40 \text{ }^\circ\text{C} \leq T \leq 80 \text{ }^\circ\text{C}$
Load capability	$\geq 5 \text{ A}$
Capacitance	$\leq 120 \text{ pF/m}$

Table 8.1-3 Thermal and electrical characteristics

Mechanical characteristics

- Single bending: $\leq 5 \times$ outer diameter
 Repeated bending: $\leq 20 \times$ outer diameter

Recommendations

1. Standard, without any special requirements:
 Two-core, flexible, shielded conductor in accordance with VDE 0812, with colored PVC sheath.
 PVC insulation resistant to oil and petroleum products.
 - ◆ Type: LIYCY 2 x 0,5 mm²
 e.g. Metrofunk Kabel-Union GmbH
 Postfach 41 01 09, 12111 Berlin
 Tel 030-831 40 52, Fax: 030-792 53 43
2. Halogen-free cable (no hydrochloric acid is generated when the cable burns):
 Halogen-free, highly flexible, resistant to extreme heat and cold.
 Sheath manufactured from a special ASS silicon-based composite.
 - ◆ Type: ASS 1 x 2 x 0,5 mm²
 e.g. Metrofunk Kabel-Union GmbH
 Postfach 41 01 09, 12111 Berlin
 Tel 030-831 40 52, Fax: 030-792 53 43
3. Recommended if halogen-free and silicon-free cables are required:
 - ◆ Type: BETAflam G-M/G-G-B1 flex. 2 x 0,5 mm²
 e.g. Studer-Kabel-AG, CH 4658 Däniken

Cable lengths

The cable length is dependent on the data transfer rate and the number of connected nodes. The following cable lengths are possible given the specified cable characteristics:

Data transfer rate	Max. number of nodes	Max. cable length
9.6 kbit/s	32	1200 m
19.2 kbit/s	32	1200 m
93.75 kbit/s	32	1200 m
187.5 kbit/s	30	1000 m

Table 8.1-4 Cable lengths

8.1.2 The structure of net data

Information which, for example, a SIMATIC S5 control unit (= master) sends to a drive (= slave) or the drive sends to the control unit is placed in the net-data area of each telegram.

8.1.2.1 General structure of the net-data block

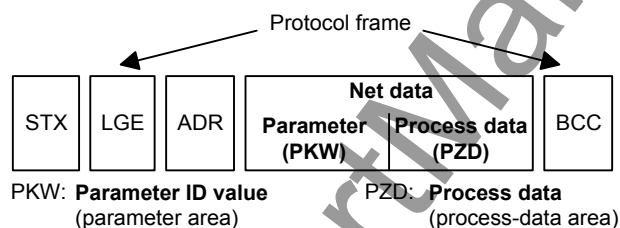
Introduction

The net-data block is divided into two areas:

- ◆ the PKW (parameter ID value) range
- ◆ the PZD (process data) range

Telegram structure

The structure of the net data in the USS-protocol telegram is shown below.



- ◆ The **PKW area** relates to the handling of the parameter ID value (PKW) interface. The PKW interface is not a physical interface but a mechanism which handles parameter transfer between two communication partners (e.g. control unit and drive). This involves, for example, reading and writing parameter values and reading parameter descriptions and associated texts. All tasks which are performed via the PKW interface essentially involve operator control and visualization, service and diagnosis.
- ◆ The **PZD area** contains the signals required for the **automation system**:
 - Control word(s) and setpoint(s) from the master to the slave
 - Status word(s) and actual value(s) from the slave to the master.

Structure of the PKW and PZD areas

PKW area			PZD area		
PKE	IND	PKW elements	PZD1	...	PZD16
Variable length			Variable length		

The two areas together make up the net data block. This structure applies to telegrams from the master to the slave and vice versa.

8.1.2.2 PKW area

With the help of the PKW mechanism, the following tasks can be performed via any serial interface with the USS protocol:

- ◆ Reading and writing parameters in the basic unit and, if available, parameters on a technology board, e.g. T100
- ◆ Reading the description of a parameter (applies to parameters of the basic unit and of technology boards)
- ◆ Reading of texts assigned to the indices of an indexed parameter. (Applies to parameters of the basic unit and of the technology modules.)
- ◆ Reading of texts assigned to the values of a parameter. (Applies to parameters of the basic unit and of the technology modules.)

Settings in the PKW area

The PKW area can be varied. Depending on the particular requirement, **3-word, 4-word or variable word lengths** can be parameterized.

PKW area parameterized for 3 words

The following is an example of a structure when access (write/read) is made to **single-word** (16 bit) parameter values:

1st word	2nd word	3rd word
PKE	IND	PWE1
Parameter ID	Index	Parameter value 1

The PKW area must be permanently set to 3 words at the master and the slave. This setting is made during start-up and should not be altered any more during bus operation.

PKW area parameterized to 4 words

The following is an example of a structure when access (write/read) is made to **double-word** (32 bit) parameter values:

1st word	2nd word	3rd word	4th word
PKE	IND	PWE1	PWE2
Parameter ID	Index	High-Word	Low Word
		Parameter value (double word)	

Parameterization to a fixed length of 4 words applies to telegrams from the master to the slave and from the slave to the master. The setting must be made both at the master and at the slave and can no longer be altered during bus operation.

**PKW area
parameterized with
variable word length**

1 st word	2 nd word	3 rd word	4 th word	...	(m+2) word
PKE	IND	PWE1	PWE2	...	PWE _m

With:

- ◆ 1 word $\leq m \leq 110$ words (maximum) when 16 PZD words (maximum) are contained in the net data block.
- ◆ 1 word $\leq m \leq 126$ words (maximum) when there is no PZD.

Telegram data transfer with variable telegram length means that the slave responds to a telegram from the master with a telegram whose length does not have to be the same length as the telegram from the master to the slave. The length of elements PEW 1 to PWE *m* in the reply telegram and what is contained in them depends on the task issued by the master. Variable length means that only the number of words necessary to pass on the appropriate information is transferred. The minimum length, however, is always 3 words.

If a slave, for example, transfers a parameter value which is a 16-bit quantity (e.g. the output voltage in parameter r003), then only 3 words of the PKW area are sent in the telegram from the slave to the master. With regard to the MASTERDRIVES MC/VC for example, if the current speed (parameter r002) is to be read, the PKW area in the telegram from the slave to the master is 4 words long since the speed is stored as a 32-bit quantity in parameter r002. Variable word-length parameterization is mandatory if, for example, all values are to be read at once from an "indexed" parameter or if the parameter description of a parameter is to be partially or completely read. This setting to variable word-length is made during start-up.

NOTICE

Do not use a variable word length if a SIMATIC S5 or SIMATIC S7 is the master.

Structure of the parameter area (PKW)

	Parameter ID					1st word
Bit No.:	15	12	11	10	0	
		AK	SPM		PNU	

	Parameter index					2nd word
Bit No.:	15			8	7	0
		Index High			Index Low	

	Parameter value					
	Parameter value High (PWE1)					3rd word
	Parameter value Low (PWE2)					4th word

AK: Task or reply ID

SPM: Toggle bit for processing of parameter-change reports

PNU: Parameter number

NOTE

The PKW area is transferred in increasing order, always starting with the 1st word.

**Parameter ID (PKE),
1st word**

The parameter ID (PKE) is always one word (16-bit quantity).

Bits 0 to 10 (PNU), together with bit 15 of the parameter index, make up the number of the desired parameter (see parameter list).

Number	PKE: Bits 0 to 10 (PNU)	Index: Bit 15	
1 - 999	1 - 999	0	Basic unit
2000 - 2999	0 - 999	1	Basic unit
1000 - 1999	1000 - 1999	0	Technology module
3000 - 3999	1000 - 1999	1	Technology module

Bit 11 (SPM) is the toggle for parameter-change reports.

MASTERDRIVES do not support parameter change reports.

Bits 12 to 15 (AK) contain the task or reply ID.

The **task IDs** are sent in the telegram from the master to the slave. The meaning of the IDs is given in Table 8.1-5. Correspondingly, the **reply IDs** are transferred at this position in the telegram from the slave to the master (see Table 8.1-6). Depending on the task ID, only certain reply IDs are possible. If the reply ID is 7 (task cannot be executed), then an error number is entered in parameter value 2 (PWE2). The error numbers are shown in Table 8.1-7.

Task ID	Meaning	Reply ID	
		positive	negative
0	No task	0	7 or 8
1	Request parameter value	1 or 2	↑
2	Change parameter value (word)	1	
3	Change parameter value (double word)	2	
4	Request descriptive element 1	3	
6	Request parameter value (array) ¹	4 or 5	
7	Change parameter value (array, word) ²	4	
8	Change parameter value (array, double word) ²	5	
9	Request the number of array elements	6	
10	Reserved	-	
11	Change parameter value (array, double word) and save in EEPROM ²	5	
12	Change parameter value (array, word) and save in EEPROM ²	4	
13	Change parameter value (double word) and save in EEPROM	2	
14	Change parameter value (word) and save in EEPROM	1	↓
15	Read or change text (only supported via OP or DriveMonitor)	15	7 or 8

¹ The required element of the parameter description is specified in IND (2nd word)

² The required element of the indexed parameter is specified in IND (2nd word)

Table 8.1-5 Task IDs (master -> drive converter)

Reply ID	Meaning
0	No reply
1	Transfer parameter value (word)
2	Transfer parameter value (double word)
3	Transfer descriptive element ¹
4	Transfer parameter value (array, word) ²
5	Transfer parameter value (array, double word) ²
6	Transfer the number of array elements
7	Task cannot be executed (with error number)
8	No control/change rights for the PKW interface
9	Parameter change report (word)
10	Parameter change report (double word)
11	Parameter change report (array, word) ²
12	Parameter change report (array, double word) ²
13	Reserved
14	Reserved
15	Transfer text

* For table footnotes ¹ and ², see Table 8.1-5

Table 8.1-6 Reply IDs (drive converter -> master)

Example

Source for the ON/OFF1 command (control word1, bit 0):
P554 (=22A hex) Change parameter value (array, word) and save in the EEPROM.

Parameter ID (PKE)															1st word						
Bit No.:	15				12		11	10								0					
	AK						SPM				PNU										
	1	1	0	0			0	0	1	0		0	0	1	0		1	0	1	0	
	C						2				2					A					Binary value
																					HEX value

- ◆ Bits 12 to 15: Value = 12 (= "C" hex); change parameter value (array, word) and save in the EEPROM
- ◆ Bits 0 to 11: Value = 554 (= "22A" hex); parameter number with a set change-report bit

**Error numbers for
reply "Task cannot
be executed"**

No.	Meaning
0	Inadmissible legal parameter number (PNU); if PNU is not available
1	Parameter value cannot be changed; if the parameter is a visualization parameter
2	Lower or upper limit exceeded
3	Erroneous sub-index
4	No array
5	Incorrect type of data
6	Setting not permitted (can only be reset)
7	Descriptive element cannot be changed; not possible
11	No operator control rights
12	Key word missing; Drive converter parameter: 'Access Key' and/or 'Parameter Special Access' not correctly set
15	No text array available
17	Task cannot be executed due to operating status; drive converter status does not permit the set task at the moment
101	Parameter number deactivated at the moment; Parameter has no function in the present state of the drive converter (e.g. type of closed-loop control)
102	Channel width too small; only for short channels The parameterized length of the PKW area is too large due to internal limitations of the drive converter. This error message can occur with the USS protocol on the T100 technology board only if access is made to parameters of the basic unit from this interface.
103	Number of PKWs incorrect; only for G-SCom 1/2 and SCB interface (USS); The error number is transferred in the following two cases: • if the task concerns all the indices of an indexed parameter (task index equal to 255) or the whole parameter description is requested and a variable telegram length has not been parameterized. • if the parameterized number of PKWs (process-data items) in the telegram is too small for the set task (e.g. alteration from the double word and the number of PKWs is 3 (words).
104	Parameter value not permissible; This error number is transferred if the parameter value which is to be transferred does not have an assigned function in the drive converter or cannot be accepted at the instant of the change for internal reasons (although it lies within the limits).
105	The parameter has been indexed e.g. task 'PWE change word' for indexed parameter
106	Task not implemented

Table 8.1-7 Error numbers for the reply ID "Task cannot be executed"

**Example
Error message 104**

The parameter 'SCom/SCB PKW #' P702:

- ◆ Minimum value: 0 (0 words)
- ◆ Maximum value: 127 (corresponds to: variable length)
- ◆ Permissible values for USS: 0, 3, 4 and 127.

If a change task with a PWE which is not 0, 3, 4 or 127 is issued to the drive converter, the reply is "Task cannot be executed" with error value 104.

**Parameter index
(IND) 2nd word**

The low-part of the index (bit 0 to 7), depending on the task, describes a definite element:

- ◆ desired array element in the case of indexed parameters,
- ◆ desired element of the parameter description,
- ◆ for indexed parameters with "index text": desired index text,
- ◆ for non-indexed parameters with "selection text": desired selection text.

Bits 8 to 14 must as a general rule all be equal to 0. The only exceptions are those parameters that are indexed and possess "selection texts". In this case bit 9 must be set to 1 to clearly identify the desired text type. The low-part then defines the desired "selection text".

Bit 15, together with bits 0 to 10 in the PKE, serves to constitute the number of a parameters (see Parameter coding).

**Special significance
of index value 255
(low-part)**

With regard to the task "Request (parameter element) descriptive element" (= AK 4) or tasks relating to the reading/writing of indexed parameters (= arrays), index value 255 has a special significance:

Task ID	Meaning
4	The complete (parameter) description is requested
6	Request all values of the indexed parameter This task can generate error message 102.
7, 8, 11 or 12	All values of the indexed parameter are to be changed. These tasks can generate error message 102.

Table 8.1-8 Tasks with index value 255

**Example
Parameter index**

Source for ON/OFF1 command (control word 1, bit 0): P554
(= 22A hex)
Change parameter value of index 1.

Bit No.:	Parameter index				2nd word HEX value
	15	8	7	0	
	0	0	0	1	

Bit 0 to 7: Index or number of the descriptive element
 Bit 8 to 14: 0
 Bit 15: 0

**Parameter value
(PWE) 3rd and 4th
words**

Depending on the word length parameterization of the PKW area, the parameter value (PWE) is transferred as word or double word (32 bit). Only one parameter value can be transferred in a telegram.

If the word length of the PKW area is parameterized with 3 words, then only 16 bit parameters can be transferred. Parameter description elements larger than 16 bit and texts cannot be transferred.

If the word length of the PKW area is parameterized with 4 words, then 16 and 32 bit parameters can be transferred. Parameter description elements larger than 32 bit and texts cannot be transferred.

If the word length of the PKW area is parameterized with "Variable length" (127), then 16 and 32 bit parameters can be transferred. Parameter description elements and texts can also be transferred. Furthermore, all elements of an indexed parameter can be read or changed as a single task and the whole parameter description can be called (index value: low-part = 255).

Transfer of a 16-bit parameter value:

1. PKW area, fixed, 3 words:
PWE1 contains the value
2. PKW area, fixed, 4 words:
PWE2 (least significant word, 4th word) contains the value; PWE1 is set to 0.
3. PKW area, variable:
PWE1 contains the value. There is no PWE2 or higher!

Transfer of one 32-bit parameter value:

1. PKW area, fixed, 3 words:
Task is rejected with error message 103.
2. PKW area, fixed, 4 words:
PWE1 (most significant word; 3rd word) contains the high-word of the double word
PWE2 (least significant word; 4th word) contains the low-word of the double word.
3. PKW area, variable:
As 2.; There is no PWE3 or higher!

**Example
Parameter value**

Source for the ON/OFF1 command (control word 1, bit 0): P554
(= 22A hex)
Change parameter value of index 1 to the value 2100 (hex).

Bit No.:	Parameter value				
	31	24	23	16	
	0	0	0	0	3 rd word, PWE1 (hex)
Bit No.:	15	8	7	0	4 th word, PWE2 (hex)
	2	1	0	0	

Bit 0 to 15: Parameter value for 16-bit parameter or low component for 32-bit parameter

Bit 16 to 31: Value = 0 for 16-bit parameter or high component for 32-bit parameter

8.1.2.3 Process-data area (PZD)

In this area, process data are **continually** exchanged between the master and slaves. The process data to be exchanged with a slave is configured at the start of communications. The setpoint for the current is to be transferred to slave x in the second PZD (= PZD2), for example. This setting is fixed for the whole procedure of data transfer.

PZD1-PZD16 = Process data

(= control / status word(s) and setpoint(s) / actual value(s))

The control/status word(s), setpoint(s) and actual value(s) required for the automation system are transferred in this area.

The length of the PZD area is determined by the number of PZD elements and their size (e.g. word, double word). In contrast to the PKW area, which can be variable, the length of this area (master and slaves) must always be agreed on between the communication partners. The maximum number of PZD words per telegram is limited to 16 words. If only PKW data is to be transferred in the net data block, the number of PZDs may even be 0!

In PZD1, control word 1 or status word 1 is always transferred, depending on the direction of data transfer and, in PZD2, the main setpoint or the main actual value is always transferred, again depending on the direction of data transfer. In the subsequent process data areas PZD3 to PZDn, additional setpoints and actual values are sent. For SIMOVERT MASTERDRIVES, control word 2 or status word 2, if necessary, is transferred in PZD4.

Structure of the PZD area

1 word	1 word	1 word	...	1 word
PZD1	PZD2	PZD3		PZD16

Maximum 16 words

Minimum 0 words, i.e. no PZD area in the net data block

NOTE

PZDn is always transferred before PZDn+1 on the USS bus.

**Task telegram
(master → slave)**

PZD1	PZD2 / PZD3	PZD4	PZD5 ... PZD16
Control word 1	Setpoint (32 Bit) / Setpoints (16 Bit)	Setpoint / Control word 2	Setpoints

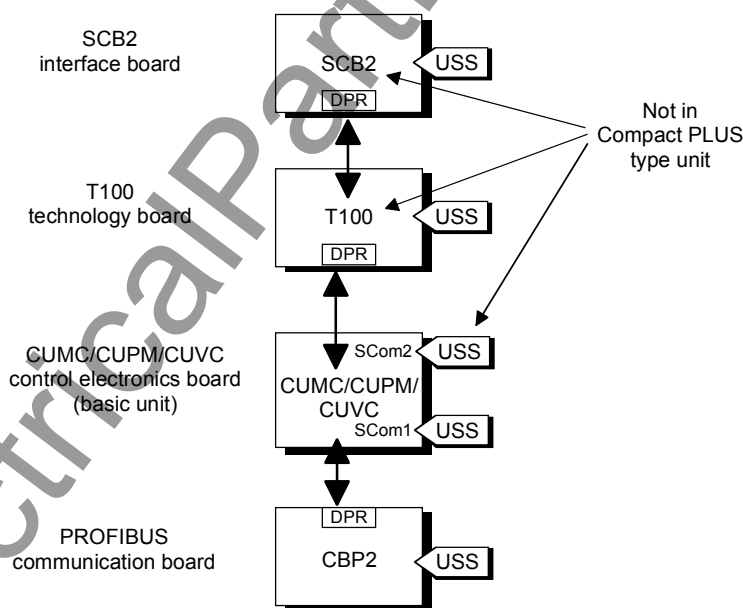
**Reply telegram
(slave → master)**

PZD1	PZD2 / PZD3	PZD4	PZD5 ... PZD16
Status word 1	Main actual value 1)	Actual values 1) / Status word 2	Actual values

- 1) Setpoint/actual value assignments are freely selectable, which means, for example, that the speed setpoint can be given in the task telegram in the PZD2, while the actual speed value can be returned in the reply telegram in the PZD2 (technologically useful). Or another actual value can be returned, such as actual torque value, actual position value or actual current value.

8.1.3 Interface overview

The following section describes all of the presently available SIMOVERT MASTERDRIVES MC/VC interfaces which use the USS protocol.



8.1-8

Interface overview

Fig.

**Basic unit with
CUMC/CUPM/
CUVC/CUVP**

In the SIMOVERT MASTERDRIVES MC series, the control electronics board, CUMC (Control Unit Motion Control), CUPM (Control Unit Motion Control Performance 2), CUMP (Control Unit Motion Control Compact PLUS Performance 2), CUVC (Control Unit Vector Control) or CUVP (Control Unit Vector Control Compact PLUS) is used. Depending on the type of basic unit, it has at least one serial interface with the USS protocol. The following table shows the available interfaces:

Board	Number of interfaces	Physical interface	Baud rate [kBit/s]
CUPM in Compact and chassis type unit	2 interfaces with USS protocol Designation: SCom1 and SCom2	RS485 / 2-wire on terminal strip X103 (SCom1 and SCom2) or RS232 or RS485 / 2-wire at 9-pole SUB-D socket X300 (SCom1)	max. 38.4
CUMP in Compact PLUS unit	1 interface with USS protocol Designation: SCom1	RS485 / 2-wire at terminal strip X100 or RS232 or RS485 / 2-wire at 9-pole SUB-D socket X103	max. 38.4
CUMC in Compact PLUS unit	1 interface with USS protocol Designation: SCom1	RS485 / 2-wire at terminal strip X100 or RS232 or RS485 / 2-wire at 9-pole SUB-D socket X103	max. 38.4
CUMC in Compact and chassis type unit	2 interfaces with USS protocol Designation: SCom1 and SCom2	RS485 / 2-wire on terminal strip X103 (SCom1 and SCom2) or RS232 or RS485 / 2-wire at 9-pole SUB-D socket X300 (SCom1)	max. 38.4
CUVC in Compact and chassis type unit	2 interfaces with USS protocol Designation: SCom1 and SCom2	RS485 / 2-wire on terminal strip X101 (SCom2) and RS232 or RS485 / 2-wire at 9-pole SUB-D socket X300 (SCom1)	max. 38.4
CUVP in Compact PLUS unit	2 interfaces with USS protocol Designation: SCom1 and SCom2	RS485 / 2-wire on terminal strip X100 (SCom2) and RS232 (SCom1) or RS485 / 2-wire (SCom2) at 9-pole SUB-D socket X103	max. 38.4

Table 8.1-9 Interfaces on the CU board

NOTICE

All the interfaces on the CU boards are non floating (not electrically isolated).

**SCB 2
supplementary
board**

The SCB2 (Serial Communications Board) is an expansion board of the SIMOVERT MASTERDRIVES. The board has a floating RS485 interface. Either the peer-to-peer protocol or the USS protocol can be used at this interface.

NOTE

The supplementary SCB2 board cannot be built into the Compact PLUS type of unit.

Board	Number of interfaces	Physical interface
SCB2	1 interface with USS protocol	RS485 / 2-wire at terminal strip X128

Table 8.1-10 Interface on the SCB 2 board

NOTE

For a more detailed description of the SCB 2, refer to the instruction manual, "Serial Communication Board 2" (Order No.: 6SE7087-6CX84-0BD0).

T100 technology board

The T100 technology board is an expansion board of the SIMOVERT MASTERDRIVES. The board has two, non-floating RS485 interfaces. One interface is permanently provided for the peer-to-peer protocol, the other is for the USS protocol.

NOTE

The T100 technology board cannot be built into the Compact PLUS type of unit.

Board	Number of interfaces	Physical interface
T100	1 interface with USS protocol and 1 interface for peer-to-peer linking	RS485 / 2-wire at terminal strip X132

Table 8.1-11 Interfaces on the T100 board

NOTE

For a more detailed description of the T100, refer to the instruction manual "Technology Board T100" [Order No. 6SE7080-0CX87-0BB0, (hardware) and 6SE7080-0CX84-0BB0 (software)].

CBP2 supplementary board

The CBP2 interface board (Communication Board PROFIBUS 2) is an extension board of the SIMOVERT MASTERDRIVES. The board has a floating RS485 interface. For this interface, either the PROFIBUS protocol or the USS protocol can be used.

Board	Number of interfaces	Physical interface
CBP2	1 interface with USS protocol	RS485 / 2-wire at terminal strip X448

Table 8.1-12 Interface on the CBP2 board

NOTE

A more detailed description of the CBP2 can be found in the operating instructions "CBP/CBP2 - Communication Board PROFIBUS" (Order No.: 6SE7087-6NX84-0FF0).

8.1.4 Connecting-up

DANGER



- ◆ The equipment is operated at high voltages. They must be in a no-voltage condition (off load) during all connecting work!
- ◆ When work is being done on the unit, it must be in a no-load condition, i.e. it must be disconnected and locked-out from the line supply.
- ◆ Only appropriately qualified personnel may work on or with the equipment.
- ◆ Death, severe bodily injury or considerable material damage may result if this warning is not complied with.
- ◆ Due to the DC link capacitors, there are still hazardous voltage levels in the equipment for at least 5 minutes after it has been disconnected from supply. There must therefore be a delay of at least 5 minutes before the unit is opened.
- ◆ The power terminals and the control terminals can still carry hazardous voltage even when the motor has been shut down.

8.1.4.1 Bus cable connection

MC, VC, "Compact PLUS" type

On SIMOVERT MASTERDRIVES, connection of the USS bus cable depends on the control version and, in the case of MC units, it is dependent on the respective type of construction.

With the "Compact PLUS" type of unit, either terminal strip X100 or connector X103 can be used to connect up the USS bus cable. The exact pin assignment is given in the relevant operating instructions for the basic unit.

MC, "Compact type" and "chassis type"

With "Compact type" and "chassis type" units, the SCom1 and SCom2 interfaces can be operated at the same time on terminal strip X103 with the USS protocol. Alternatively, connector X300 can be used as SCom1. The exact pin assignment of terminal strip X103 or connector X300 is given in the relevant operating instructions of the basic unit.

VC, "Compact type" and "chassis type"

In the case of the "Compact type" and "chassis type" units, either the connection of terminal strip X101 (SCom2) or X300 (SCom1) can be used to connect up the USS bus cable. The exact pin assignment of terminal strip X101 or connector X300 is given in the relevant operating instructions of the basic unit.

SCB 2 board

In the case of the SCB2 board, the bus cable is terminated at terminal strip X128. The exact pin assignment and other notes on termination are given in the operating instructions for the SCB2.

Technology board T100

In the case of the T100 technology board, the USS protocol is implemented at interface 1. The bus cable is terminated at terminal strip X132. The exact pin assignment and other notes on termination are given in the hardware operating instructions for the T100.

8.1.4.2 Fitting the bus cable

At all interfaces to the CUPM, CUMC, CUVC control electronics, the SCB2 board and the T100, except for connectors X103 and X300 or X448 (9-pin SUB-D connectors), the USS bus cable is connected by means of screw/plug-in terminals. The correct method of connecting the bus cable at the connector is shown in the following diagram.

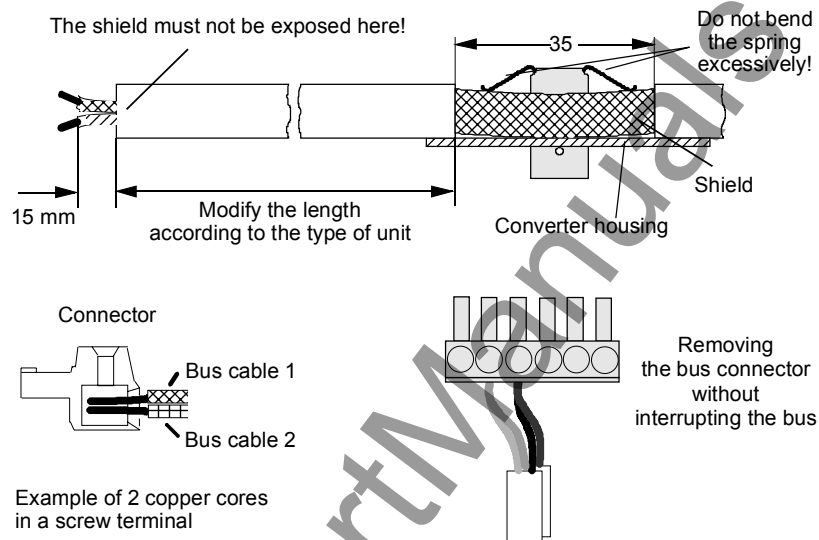


Fig. 8.1-9 Connecting up the bus cables

NOTE

It must be ensured that both copper cores are securely held inside the screw terminal.

8.1.4.3 EMC measures

For interference-free operation of the USS, it is absolutely necessary that the following measures are carried out:

Shielding

Shielding is necessary for damping magnetic, electrical and electromagnetic interference fields. Interference currents are discharged to earth by the shield braiding via the housing earth.

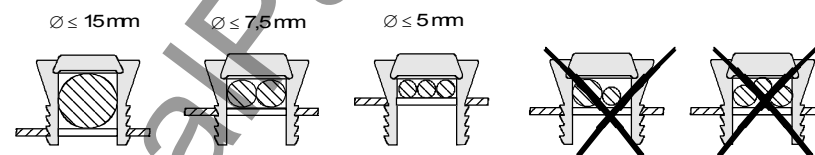
NOTE

The bus cables must be twisted and shielded and are to be routed separately from power cables, the minimum clearance being 20 cm. The shield must be connected through the largest possible surface area on both sides, i.e. the shield of the bus cable between 2 converters must be connected to the converter housing at **both** ends. The same applies to the shield of the bus cable between master and converter.

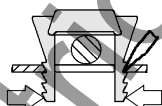
If bus and power cables intersect, they must do so at an angle of 90 °.

- ◆ With regard to the bus cable, the shield must not be exposed in the bus connector. Shielding is provided by the shield clamps (Compact type units) or shield clamps and cable ties (chassis type units) at the converter housing. How to use the shield clamps is shown in the following illustration. It must be ensured that the solid copper core is not damaged when the insulation is removed from the ends of the conductors.
- ◆ It must also be ensured that the shield of every bus cable is connected where the cable enters the cabinet as well as at the converter housing!

Snap in the shield clamp



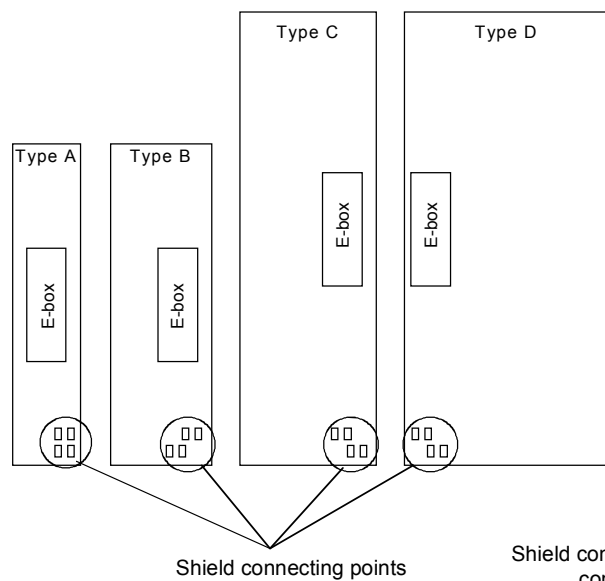
Release the shield clamp



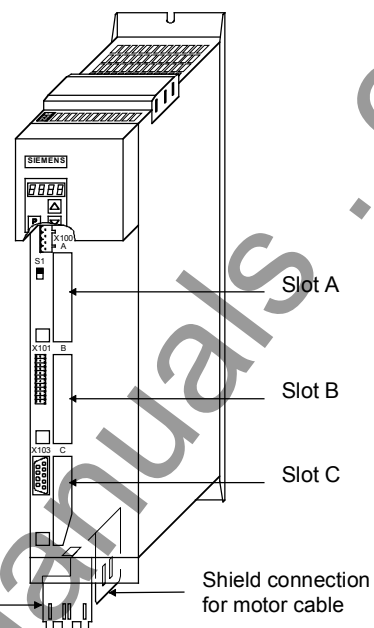
Squeeze the shield clamp together with your hand or a screwdriver and pull upwards.

Fig. 8.1-10 Using the shield clamps

Compact type and chassis type units



Compact PLUS MC:



Compact PLUS VC:

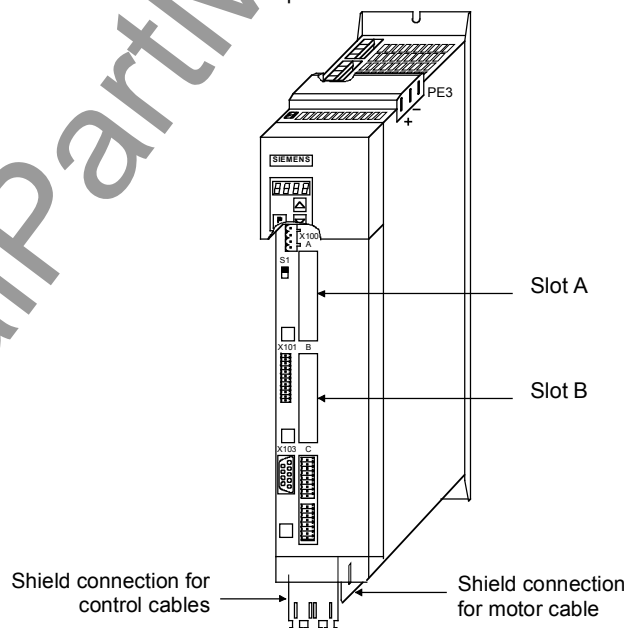


Fig. 8.1-11 Position of the shield connecting points

Equipotential bonding

Equipotential bonding is necessary in order to prevent differences in potential (e.g. due to different supply voltages) between the individual bus nodes (converters and master system).

- ◆ This is achieved with the help of equipotential-bonding conductors:
 - 16 mm² Cu for equipotential-bonding conductors up to 200 m in length
 - 25 mm² Cu for equipotential-bonding conductors more than 200 m in length
- ◆ The equipotential-bonding conductors are to be laid so that there is the smallest possible surface area between a conductor and any signal cables.
- ◆ The equipotential-bonding conductor must be connected to the earth electrode/protective conductor through the largest possible surface area.

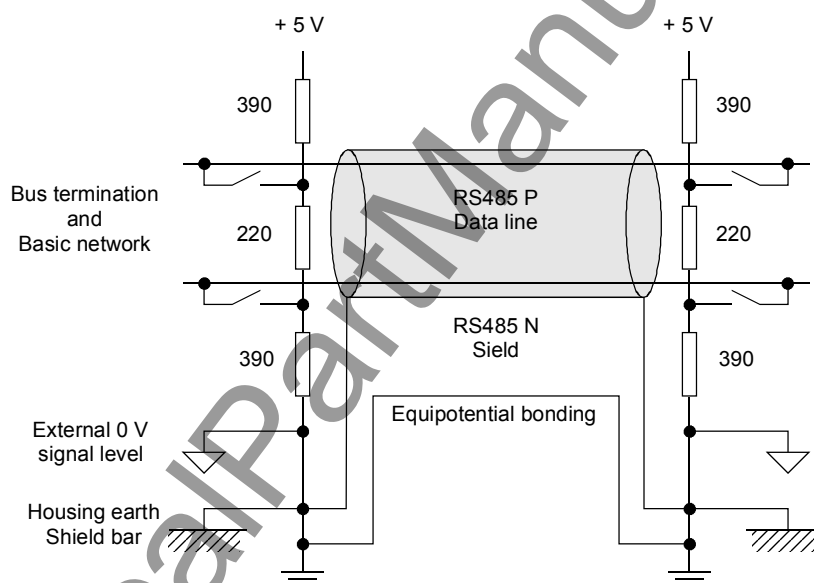


Fig. 8.1-12 Shielding and equipotential bonding

Laying cables

Instructions for laying cables:

- ◆ Bus cables (signal cables) must not be laid close to and parallel to power cables.
- ◆ Signal cables and the associated equipotential-bonding cables must be laid as closely together as possible and kept as short as possible.
- ◆ Power cables and signal cables must be laid in separate cable ducts.
- ◆ Shields must be connected through the largest possible surface area.

For more information on electromagnetically compatible installation of systems, see for example Chapter 3 of the Compendium or the description "Instructions for Design of Drives in Conformance with EMC Regulations" (Order No. 6SE7087-6CX87-8CE0).

8.1.4.4 Bus termination, USS protocol

In order to ensure interference-free USS operation, the bus cable must be terminated with bus terminating resistors at both ends. The bus cable from the first USS node to the last USS node is to be regarded as **one** bus cable. The USS bus therefore must be terminated twice. The bus terminating resistors must be switched in at the **first** bus node (e.g. master) and **last** bus node (e.g. converter).

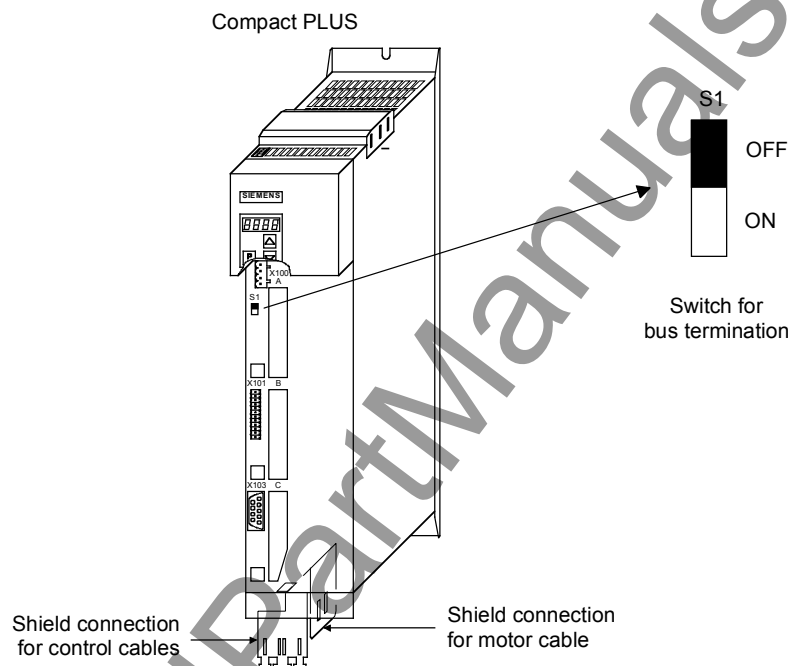


Fig. 8.1-13 S1 bus-terminating switches in the Compact PLUS type of unit

NOTE

In the Compact and chassis type units, two mutually independent USS interfaces (SCom1 and SCom2) are available. Switch S1 or S2 is provided for switching in the terminating resistor.

If the bus-terminating node is a T100 board, the bus terminating resistors are switched in through the two plug-in jumpers, X8 and X9.

NOTE

- ◆ When the unit is supplied, the terminating resistors are not switched in!
- ◆ Please note that the bus termination is switched in only at the first bus node (e.g. SIMATIC S 5/CP524) and last bus node (e.g. CUPM)! When the matching resistors are being set, the electronics box must be **isolated from supply**!
- ◆ **Data transmission faults possible on the bus!**
During active bus operation, the units with a **switched-in** terminating resistor must not be disconnected from supply. The matching resistor when disconnected from supply (off-load) is no longer effective because the terminating resistor obtains its voltage from the connected unit.

Bus connection via terminal strip

The following illustration shows an example of the bus connection at terminal strip X100 (Compact PLUS). If the connector at terminal strip X100 of one node is removed, data transfer via bus is **not** interrupted. The other nodes on the bus continue to be supplied with data via the bus.

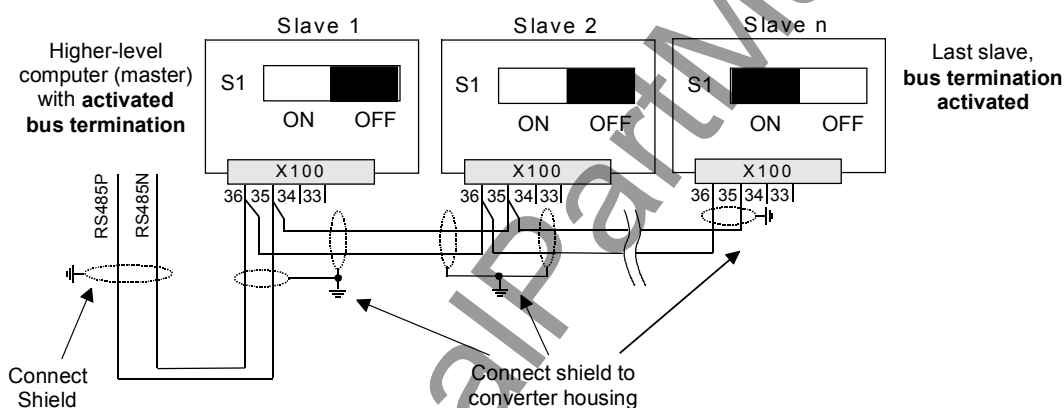


Fig. 8.1-14 Connection of the 2-wire bus cable at terminal strip X100 (Compact PLUS)

Bus connection via connector X103

The following illustration shows the structure of a bus connection via the 9-pin connector, X103 (Compact PLUS).

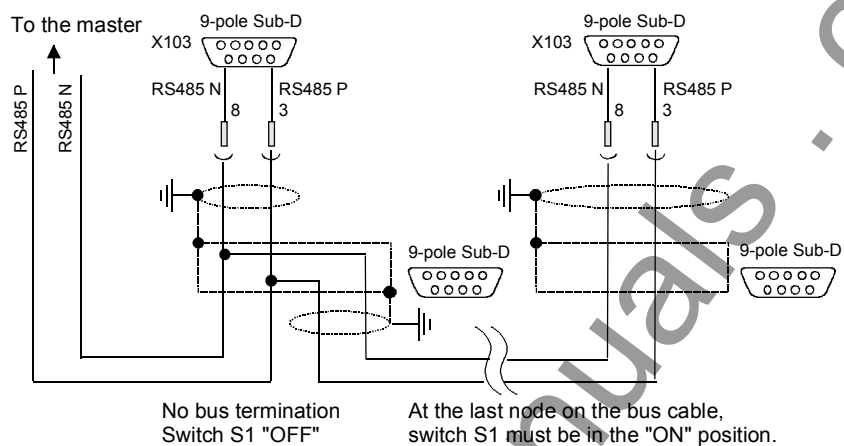


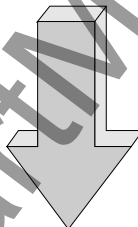
Fig. 8.1-15 Connection of the 2-wire bus cable at terminal strip X103 (Compact PLUS)

8.1.5 Start-up

The USS protocol can be started up in two steps:

1. Parameterization of the USS protocol at the "selected" interface
2. Parameterization of process-data interconnections and the "parameterizing enable" for the selected interface.

Parameterizing the USS protocol
Create the right conditions: Set P060 = 1 (menu selection)
Parameterize the interface: Settings to be made: - P682 (SCB protocol) only applies to the SCB2, - P700 (SCom/SCB BusAddr), P701 (SCom/SCB baud rate), - P702 (SCom/SCB PKW #), P703 (SCom/SCB PcD # and P704 (SCom/SCB TIgOFF)



Parameterizing the parameterizing enable and process-data interconnections
Set the parameterizing enable via USS at the selected interface: Set P053 (parameter access)
Set process-data interconnections: - For status words and actual values: - P707 (Src SCom 1 TrnsDat) and P708 (Src SCom 2 TrnsDat) for CUPM - P690 (SCB actual value) for SCB 2 board - For control words and setpoints: - e.g. P554 (control word, bit 0) to P591 (control word, bit 32), - P443 (Src Main Setp), P433 (Src Add Setp1), etc.

8.1.5.1 Parameterization of the USS protocol (1st step)

The USS protocol is parameterized at serial interfaces SCom 1 and SCom 2 on the CU board of the basic units or at the serial interface on the SCB 2 board by means of the following parameters: **P682, P700, P701, P702, P703 and P704.**

NOTE

The USS protocol is parameterized at the serial interface of the T100 technology board by means of the "technology parameters" H290, H291, H292, H293, H294 and H295. These parameters are part of the T100 (see software instruction manual of the T100).

Example 1

USS protocol at the SCom1 on MASTERDRIVES MC

As already described in Section 8.1.3, the bus cable for the SIMOVERT MASTERDRIVES MC can be connected either at terminal strip X100/X103 ("Compact PLUS" type) or at connector X103/X300 ("Compact" and "chassis" types).

- ◆ Settings:
 - USS protocol with 19.2 kbit/s and 3-word PKW area and 2-word PZD area
 - 3-word PKW area:
 - With this setting, all parameters whose values are 16-bit quantities (1 word) can be read and written via the USS protocol.
 - 2-word PZD area:
 - Transfers control word 1 and a setpoint (each of them 16 bit) from the master to the converter and status word 1 and an actual value (each of them 16 bit) from the converter to the master.
- ◆ Preconditions:
 - P060 = 1 or 7 (default setting)
- ◆ Parameterizing the SCom 1 interface (applies to X100 or X103 ("Compact PLUS" type) and X103 or X300 ("Compact" and "chassis" types) at the same time):

Parameter number	Parameter	Index and value (index i001 for SCom 1)	Comments
P700	SCom/SCB BusAddr	i001 = 0	Bus address SCom1 = 0
P701	SCom/SCB Baud	i001 = 7	19.2 kbit/s
P702	SCom/SCB PKW #	i001 = 3	3-word PKW (SCom 1)
P703	SCom/SCB PcD #	i001 = 2	2-word PZD (SCom 1)
P704	SCom/SCB TlgOFF	i001 = 0 to 6500	0: No monitoring >0: Monitoring time in ms

Example 2**USS protocol at the SCom2 (only in Compact type and chassis type units)**

- ◆ Setting:
USS protocol with 38.4 kbit/s and 4-word PKW area and 6-word PZD area
 - 4-word PKW area:
With this setting, all parameters whose values are 16-bit (= 1 word) or 32-bit (double word) quantities can be read or written via the USS protocol.
 - 6-word PZD area:
Transfers control words 1 and 2 and a maximum of four setpoints (each of them 16 bits) from the master to the converter or control words 1 and 2 (each one of them 16 bits) and a maximum of four actual values (each one of them 16 bits) from the converter to the master.
- ◆ Preconditions:
P060 = 1 or 7
- ◆ Parameterizing the SCom2 interface (CUPM, CUMC: X103, CUVC: X101):

Parameter number	Parameter	Index and value (index i002 for SCom 2)	Comments
P700	SCom/SCB BusAddr	i002 = 15	Bus address, SCom 2 = 15
P701	SCom/SCB Baud	i002 = 8	38.4 kbit/s
P702	SCom/SCB PKW #	i002 = 4	4-word PKW (SCom 2)
P703	SCom/SCB PcD #	i002 = 6	6-word PZD (SCom 2)
P704	SCom/SCB TIgOFF	i002 = 0 to 6500	0: No monitoring >0: Monitoring time in ms

Example 3**USS protocol at the SCB2 board**

- ◆ Settings:
USS protocol with 19.2 kbit/s and 4-word PKW area and 2-word PZD area
 - 4-word PKW area:
With this setting, all parameters whose values are 16-bit (= 1 word) or 32-bit (double word) quantities can be read or written via the USS protocol.
 - 2-word PZD area:
Transfers control word 1 and a setpoint (each of them 16 bit) from the master to the converter and control word 1 and an actual value (each of them 16 bit) from the converter to the master.
- ◆ Preconditions:
P060 = 1 or 7
- ◆ Parameterization of the interface on the SCB2 board:

Parameter number	Parameter	Value	Comments
P682	SCB protocol	2	Physical bus cable, 2-wire USS protocol (according to /1/, only USS operation with 2 wires is defined).

Parameter number	Parameter	Index and value (index i003 for SCB2)	Comments
P700	SCom/SCB BusAddr	i003 = 21	Bus address SCom2 = 21
P701	SCom/SCB Baud rate	i003 = 7	19.2 kbit/s
P702	SCom/SCB PKW #	i003 = 4	4-word PKW
P703	SCom/SCB PcD #	i003 = 2	2-word PZD
P704	SCom/SCB TlgOFF	i003 = 0 to 6500	0: No monitoring >0: Monitoring time in ms

Example 4**USS protocol on the CBP2 board**

- ◆ Settings:
USS protocol with 19.2 kbit/s and 4-word PKW area and 2-word PZD area
 - 4-word PKW area:
With this setting, all parameters whose values are 16 bit- (= 1 word) or 32-bit variables (double word) can be read or written by means of the USS protocol.
 - 2-word PZD area:
Transmission of control word 1 and a setpoint (each 16 bits) from the master to the converter and of status word 1 and an actual value (each 16 bits) from the converter to the master.
- ◆ Requirements:
P060 = 1 or 7
- ◆ Parameterization of the interface on the CBP2 board:

Parameter number	Parameter	Value	Comments
P713.x	CBP2 protocol	2	A change from PROFIBUS to USS protocol and vice versa only comes into effect when the voltage of the drive is turned off and then on again.

Parameter number	Parameter	Value	Comments
P918.x	CBP2 BusAddr	21	Bus address CBP2 = 21
P718.x	CBP2 Baud	7	19.2 kbit/s
P719.x	CBP2 PKW #.	4	4-word PKW
P720.x	CBP2 PcD #.	2	2-word PZD
P722.x	CBP2 TlgOFF.	0...6500	0: No monitoring >0: Monitoring time in ms

8.1.5.2 Parameterizing the parameterizing enable and process-data interconnections (2nd step)

Parameterization of the parameterizing enable

During start-up, an interface with the USS protocol must be explicitly enabled for parameterization in order to be able to change (= write) the parameters of a SIMOVERT MASTERDRIVES via this interface – this applies to the parameters of the basic unit (P/U parameters) and to the technology-board parameters (H/L parameters).

NOTE

Access to the SIMOVERT MASTERDRIVES via USS protocol is only possible if, during start-up, the PKW area is appropriately defined to contain 3, 4 words (fixed length) or a variable PKW length (= value 127) in the useful (net) data area.

The following rules apply to this:

- ◆ All parameters (P, r, U and n parameters of the basic units, or H, d, L and c parameters of the technology board) can be read out via any interface. **For reading purposes**, it is **not** necessary that the interface has been enabled for parameterization.

P, U, H and L parameters:	Can be read and written
r, n, d and c parameters:	Can only be read
- ◆ Parameterizing enable is specified in **parameter P053** (parameter access). This parameter **can always be written** from any interface.
- ◆ Several interfaces can be in possession of a parameterizing enable simultaneously.

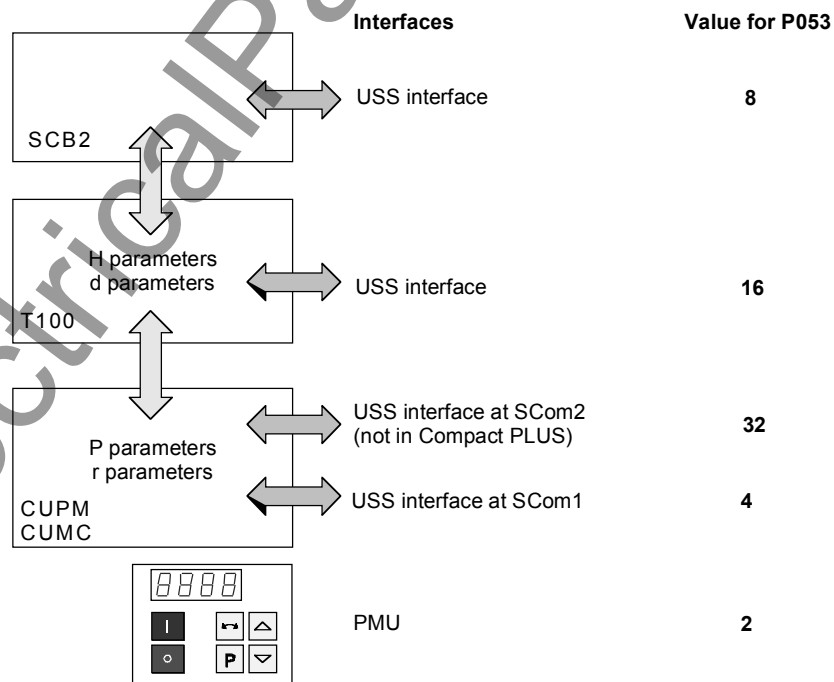


Fig. 8.1-16 Parameterizing enable for the USS interfaces

The rules for generating the value which is entered in parameter P053 for specifying parameter access is explained with the following example.

Example

Setting the parameterizing enable for SIMOVERT MASTERDRIVES with SCB2

Setting:

Write access to the parameters of the basic units (P parameters) via the PMU as well as via the USS protocol at both SCom1 interfaces and on SCB2

Parameter number	Value	Comments
P053	14	2 = PMU, 4 = SCom1, 8 = SCB2 → value = 2 + 4 + 8 = 14

Parameterizing process-data interconnections

As already described in Section 8.1.2.3 (PZD area), the PZD area consists of a maximum of 16 words. During start-up, the length of this area is defined in words using parameter P703 (SST/SCB PZD #). This definition applies to the telegram from the master to the converter and, vice versa, to the telegram from the converter back to the master. In the telegram from the master to the converter, the PZD area contains control word 1 or control word 2 and the setpoints. In the telegram from the converter to the master, status word 1 or status word 2 and the actual values are transferred.

1 word	1 word	1 word	...	1 word
PZD1	PZD2	PZD3		PZD16

Maximum 16 words

Minimum 0 words, i.e. no PZD area in the net data block

NOTE

Here, process-data interconnection is only described for the basic units. Process-data interconnection for the technology boards is described in their instruction manual.

"Interconnecting" control word 1 and control word 2

The two control words (bits 0 to 15) and 2 (bits 16 to 31) give commands and external information to the converter. A select parameter is assigned to each control-word bit, e.g. bit 0 of parameter P554. The select parameter specifies from which source(s) this control bit can be influenced (= changed).

USS interface, from which control word bits 0 to 15 (= control word 1) are to be changed (source)	Values to which select parameters P554 to P575 are to be set
SCom1	21xy
SCom2	61xy
SCB2	45xy

Note:

- ◆ e.g. 21xy:
The first digit (here 2) identifies the interface SST1 as source.
The second digit (here 1) indicates that it is the 1st word in the PZD area of the telegram. "xy" (= 00 to 15) identifies the bit position.

NOTE

Control word 1 is always transferred in the 1st word of the PZD area in the USS protocol.

Example 1

- ◆ The control word command "ON/OFF1" should be taken from bit 0 in the 1st PZD word of SST1.
- ◆ The control word command "OFF2" should be taken from bit 1 in the 1st PZD word of SST1.
- ◆ The control word command "ACK" should be taken from bit 7 in the 1st PZD word of SST1.

Parameter number	Parameter	Index and value (index i001 for BICO data set 1) (index i002 for BICO data set 2)	Comments
P554	Source ON/OFF1	i001 = 2001	ON/OFF from SCom1
P555	Source 1 OFF2	i001 = 2001	Operating condition/OFF2 from SCom1
P565	Source 1 ACK	i001 = 2107	Edge 0 → 1

etc.

Values of select parameters P576 to P591

The following values of select parameters P576 to P591 are to be set for the USS interfaces:

USS interface from which control-word bits 16 to 31 (= control word 2) are to be changed (source)	Values to which select parameters P576 to P591 are to be set
SCom1	24xy
SCom2 (not with the Compact PLUS)	64xy
SCB2	48xy

Note:

- ◆ e.g. 48xy:
The first position (in this case, 4) identifies the interface on SCB 2 as the source.
The second digit (here 8) indicates that it is the 4th word in the PZD area of the telegram (5 signifies the 1st word). "xy" (= 00 to 15) identifies the bit position.

NOTE

If necessary, control word 2 is always transferred in the 4th word of the PZD area in the USS protocol.
→ Set PZD area to a length of at least 4 words (P703).

Example 2

- ◆ Bit 0 for switching over the function data set should be taken from bit 0 in the 4th PZD word of SCB2.
- ◆ Bit 1 for switching over the function data set should be taken from bit 1 in the 4th PZD word of SCB2.

Parameter number	Parameter	Index and value (index i001 for BICO data set 1) (index i002 for BICO data set 2)
P576	Source FDS Bit 0	i001 = 4800
P577	Source FDS Bit 1	i001 = 4801

etc.

"Interconnection" of setpoints

The user can select the source from which the setpoints for the converter are to be taken. This is done in the same way in which control-word bits are "interconnected". This is now illustrated with two examples.

Example 1

The "wiring" of the setpoints is done via parameters P443 (source main setpoint) and P433 (source supplementary setpoint 1).

Source for setpoints	Value for parameters P443 and P428
Interface allocation: SCom1 SCB2	20xx 45xx
Position of the setpoints (16 bit quantify) in the PZD area: In the 2nd word → 02 In the 3rd word → 03 etc.	xx = 02, 03, 04 (only if control word 2 is not transferred), 05, up to 16

The main setpoint comes from SCom 1 and is located in the 2nd word of the PZD area. The supplementary setpoint comes from the USS interface on SCB 2 and is also located in the 2nd word of the PZD area (for BICO data set 1).

Parameter number	Parameter	Index and value (index i001 for BICO data set 1) (index i002 for BICO data set 2)
P443	Source of main setpoint	i001 = 2002
P433	Source of supplementary setpoint 1	i001 = 4502

Example 2

The "wiring" of the setpoints is done via parameters **P443** (source main setpoint), **P433** (source supplementary setpoint 1), **P438** (source supplementary setpoint 2), and so on. For a detailed description, see the instruction manual.

Source for the setpoints	Values for parameters P443, P433, P438 and so on
Interface allocation: SCom1 SCom2 SCB2	20xx 60xx 45xx
Position of the setpoints (16-bit quantity) in the PZD area: In the 2nd word → 02 In the 3rd word → 03 and so on	xx = 02,03, 04 (only if control word 2 is not transferred), 05, up to 16
Position of the setpoints (32-bit quantity) in the PZD area: In the 2nd word + 3rd word → 32 Rules for generating: xx = 30 (indicates 32-bits) + position in the PZD area at which the 32-bit setpoint begins. In the 3rd word and 4th word → 33 and so on	xx = 32,33 (only if control word 2 is not transferred), 34 (only if control word 2 is not transferred), 35, up to 45

NOTE

When 32-bit quantities are being transferred, the high word is located in PZD n and the low word in PZD n+1
→ For example, 32-bit setpoint in PZD2 and PZD3; the high-word is then transferred in PZD2 and the low word in PZD3 via the USS bus.

The main setpoint (32-bit quantity) comes from SCom1 and is located in the 2nd word and 3rd word of the PZD area. Control word 2 is in the 4th word. In the 5th and 6th words, supplementary setpoint 1 (32-bit quantity) is transmitted (for BICO data set 1).

Parameter number	Parameter	Index and value (index i001 for BICO data set 1) (index i002 for BICO data set 2)
P443	Source of main setpoint	i001 = 2032
P433	Source of supplementary setpoint 1	i001 = 2035

"Interconnection" of status words 1 and 2 and the actual values

The two status words 1 (bits 0 to 15) and 2 (bits 16 to 31) send messages from the converter to a higher-level converter system.

An indexed parameter is assigned to each interface. Each index is assigned to a net-data word in the PZD area. For example, index i001 to the 1st word, index i002 to the 2nd word and so on up to i016.

Parameter number	Parameter	Index and value ♦ (index i001 for BICO data set 1) (index i002 for BICO data set 2)
SCom1	707 (SCom1 actual values)	i001 to 016
SCom2 (not with the Compact PLUS)	708 (SCom2 actual values)	i001 to 016
SCB2	706 (SCB actual values)	i001 to 016

NOTE

Status word 1 is always transferred in the 1st word of the PZD area in the USS protocol.

Example 1

"Interconnection" of status word 1 and the actual speed/frequency (KK0091) at interface SCom1.

- ♦ Precondition:
PZD area at least 2 words in length; P703, i001 ≥ 2 is set.

Parameter No.	Parameter	Index and value	Comments
P707	SCom1 actual values	i001 = 0032	1st word in the PZD area: status word (K0032)
		i002 = 0091	2nd word in the PZD area: actual speed/frequency (KK0091, only H-Word)
		i003 to i016 = 0	3rd to 16th word in the PZD area (if parameterized): "Not interconnected"

Example 2

"Interconnection" of status word 1, status word 2, actual speed (KK0091) and the actual DC link voltage (K0240) at the interface on SCB2.

- ◆ Precondition:
PZD area at least 5 words in length; P703, i003 ≥ 5 is set.

Parameter number	Parameter	Index and value	Comments
P706	SCB actual values	i001 = 0032	1st word in the PZD area: status word (K0032)
		i002 = 0091	2nd word in the PZD area: high word of the actual speed (KK0091)
		i003 = 0091	3rd word in the PZD area: low word of the actual speed (KK0091)
		i004 = 0033	4th word in the PZD area: status word 2 (K0033)
		i005 = 0240	5th word in the PZD area: Vd(act) (K0240)

NOTE

When 32-bit quantities are being transferred, the high word is located in PZD n, the low word in PZD n+1.

→ For example, 32-bit actual value of KK0091 in PZD2 and PZD3.

8.2 PROFIBUS

In addition to the CBP communications board, there is the CBP2 with extended functionality. It replaces but remains fully compatible with the CBP.

In the following, "CBP" refers to both boards. Any individual features which a board possesses are specially indicated.

8.2.1 Product description of the CBP communications board

The CBP communications board (Communications board PROFIBUS) is for linking SIMOVERT MASTERDRIVES® to higher-level automation systems via PROFIBUS-DP.

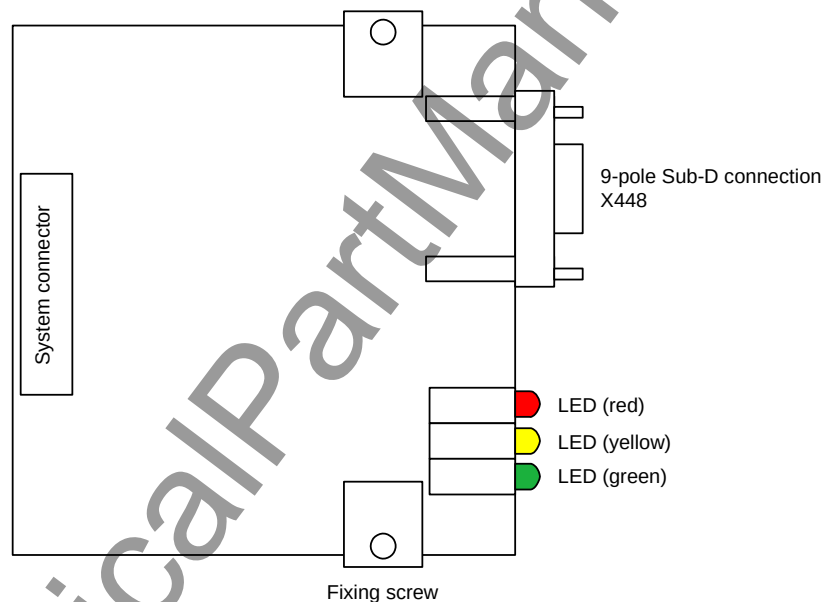


Fig. 8.2-1 View of the communications board

Technical data

The communications board has three LEDs (green, yellow, red) for providing information on the current operating status.

Voltage is supplied from the basic unit through the system's plug-in connector.

The CBP has a 9-pole SUB D socket (X448) which is provided for connecting it up to the PROFIBUS system in accordance with the PROFIBUS standard. All connections of this RS485 interface are short-circuit-proof and floating.

The CBP supports baud rates of 9.6 kbaud to 12 Mbaud and is also suitable for connecting fiber-optic cable by means of optical link plugs (OLPs).

NOTE

For reasons of space, optical link plugs cannot be used for Compact units, types 1 and 2!

Functionality

- ◆ Useful data is exchanged with the master according to the "PROFIBUS profile for variable-speed drives", PROFIdrive.
- ◆ Acyclical communications channel for transferring parameter values up to a length of 101 words with a SIMATIC S7-CPU.
- ◆ Acyclical communications channel for linking the PC-based Drive ES start-up and service tool.
- ◆ Automatic adoption of the useful data structure defined in the master.
- ◆ Monitoring of the bus interface.
- ◆ Supporting of SYNC-type PROFIBUS control commands for synchronized data transfer from the master to several slaves.
- ◆ Supporting of FREEZE-type PROFIBUS control commands for synchronized data transfer from several slaves to the master.
- ◆ Extremely simple parameterization of the CBP via the PMU of the basic unit.

Extended functionality of the CBP2

- ◆ Flexible configuration of the setpoints/actual values up to a maximum of 16 process data words
- ◆ Clock synchronization at the isochronous PROFIBUS for synchronization of processing by the master and slaves (MASTERDRIVES MC only)
- ◆ Cross traffic for direct data exchange between slaves
- ◆ Direct access to a drive by a SIMATIC OP
- ◆ USS protocol

Extension by PROFIdrive V3 functions in conjunction with CBP2 from V2.21

- ◆ Acyclical parameter channel in accordance with PROFIdrive profile, version 3, with data block 47
- ◆ Standard telegrams 1 to 6

For MASTERDRIVES MC and during use of T100, T300 or T400, please pay attention to the note in Section 2.3.2 "TB Blocks".

8.2.2 Description of the CBP's functions on the PROFIBUS-DP

Definition

PROFIBUS is an international, open field bus standard with a wide scope of application in production and process automation. Neutrality and openness are guaranteed by international standards EN 50170 and IEC 61158.

The PROFIBUS-DP enables very fast, time-critical transfer of data on the field level.

With the PROFIBUS, a distinction is made between masters and slaves.

- ◆ **Masters** determine data traffic on the bus and are also designated in the literature as active nodes.
There are two classes of master:
 - **DP-Master Class 1 (DPM1):**
These are central stations (e.g. SIMATIC S5, S7 and SIMADYN D) which exchange information with the slaves in defined communications cycles.
 - **DP-Master Class 2 (DPM2):**
Units of this type are programming units, planning units or control and monitoring units which are used for configuring, starting up or monitoring systems in operation.
- ◆ **Slaves** (e.g. CBP, CB15 etc.) can only acknowledge the messages they receive or transfer messages to a master when the latter requests a slave to do so. Slaves are also designated as passive nodes.

Protocol architecture

The protocol architecture of the PROFIBUS-DP is oriented to the OSI (Open System Interconnection) reference model in accordance with the international standard, ISO 7498, and uses layers 1 and 2 as well as the user interface.

Transmission equipment

When transmission equipment is being selected, criteria such as high transmission speed and simple, inexpensive wiring and cabling is of primary importance. PROFIBUS supports transmission according to RS485 and also transmission by means of fiber-optic cable.

The transmission speed can be selected between 9.6 kbaud and 12 Mbaud. **The same speed is specified for all units** on the bus when the system is started up for the first time.

Bus-access procedure

The PROFIBUS works according to the token-passing procedure, i.e. the masters become token holders for a defined time window in a logical ring. Within this time window, the master can communicate with other masters. Alternatively, it can communicate with slaves by using a lower-level master-slave procedure.

The PROFIBUS-DP mainly uses the master-slave method and data is usually exchanged with the drives cyclically.

Data exchange via PROFIBUS

This enables very rapid data exchange between the higher-level systems (e.g. SIMATIC, SIMADYN D, PC/PGs) and the drives. Access to the drives is always made according to the master-slaves method. The drives are always the slaves and each slave is clearly defined by its address.

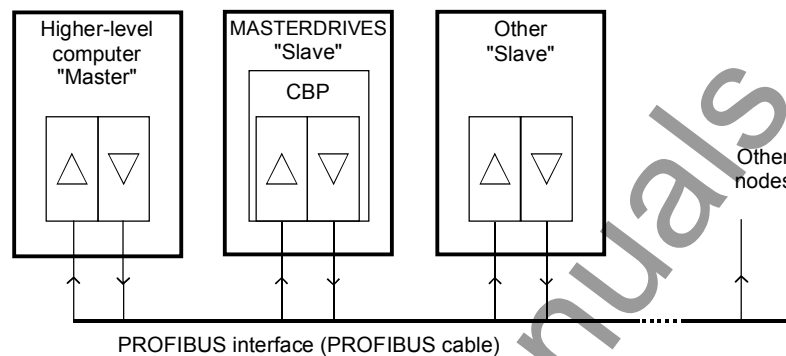


Fig. 8.2-2 PROFIBUS interfaces

The cyclical communications functions are determined by the PROFIBUS-DP basic functions in accordance with EN 50170.

For purposes of parameterization during cyclical data exchange with intelligent drives, acyclical extended communications functions are also used which are defined in PROFIBUS Guideline No. 2.081 (German) or 2.082 (English).

The following illustration contains an overview of the communications functions which are enabled with the CBP.

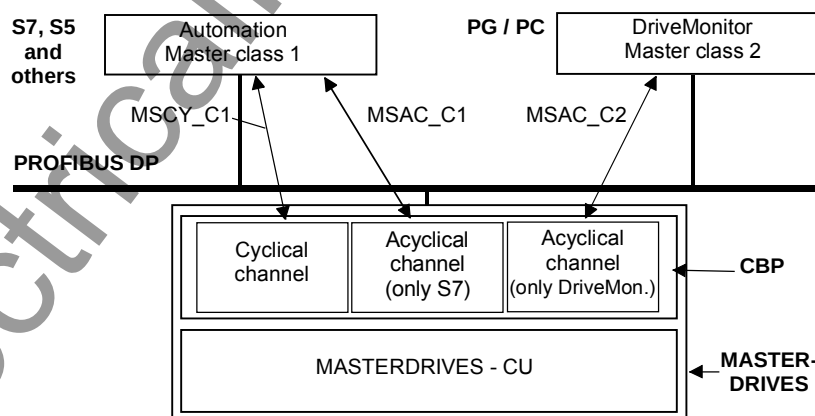


Fig. 8.2-3 Data-traffic channels of the CBP

The following illustration contains an overview of the communications functions which are enabled with the CBP2:

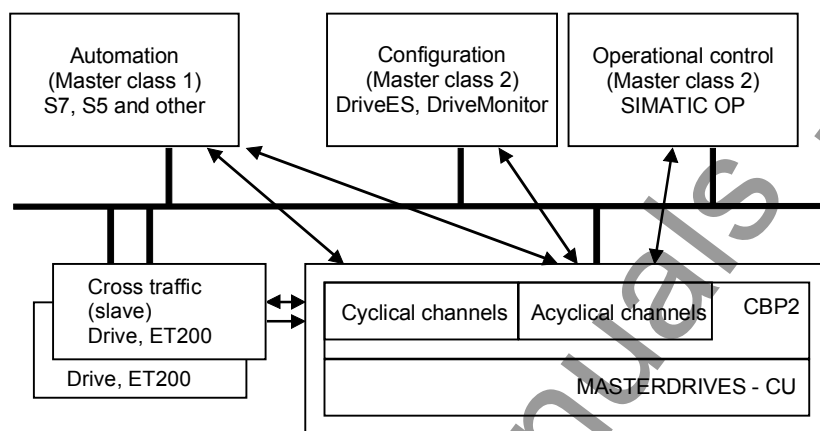


Fig. 8.2-4 Data-traffic channels of the CBP2

8.2.2.1 Cyclical data transmission

DANGER



When interconnecting connectors, binectors, and double word connectors, please note that simultaneous interconnection of a connector, and a double word connector with the same name is not permitted, because when a double word connector (e. g. KK3032) is connected, the meanings of the connectors K3002 and K3003 are swapped round (high-word and low-word exchanged).

On MASTERDRIVES MC and Compact Plus on software version V1.50 and higher and on MASTERDRIVES CUVC on software version V3.23 and higher, simultaneous use of connectors and double word connectors with the same name is mutually interlocked (see also function diagrams [121] and [131]).

Because the binectors are not included in the interlocking (to ensure compatibility for older configurations), their significance changes according to whether the pertinent word or double word is wired.

The structure of useful data as PPOs

Useful data for the **cyclical MSCY_C1 channel** (see Figs. 8.2-3 and 8.2-4) is structurally defined in the PROFIBUS profile for variable-speed drives version 2 as a parameter process data object (PPO).

Frequently, the **cyclical MSCY_C1 channel** is simply called the **STANDARD channel** as well.

NOTES

Data is exchanged with the MASTERDRIVES in accordance with the specifications of the PNO guideline "PROFIBUS profile for variable-speed drives". PROFIdrive CBP and CBP2 V2.10 implement PROFIdrive version 2 (PNO: Order No. 3071).
CBP2, V2.21 and later, implements PROFIdrive Version 3 (PNO: Order No. 3172) as a compatible expansion. The useful data structure described below is still supported.

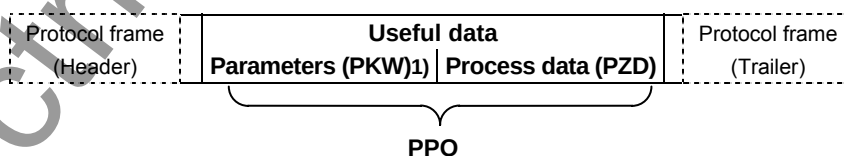
For the drives, the guideline specifies the useful-data structure with which a master can access the drive slaves by means of cyclical MSCY_C1 data transfer. With MSCY_C1 data transfer, useful data is divided up into two areas which can be transmitted in each telegram:

- ◆ The process data area (PZD), i.e. control words and setpoints or status information and actual values
- ◆ The parameter area (PKW) for reading/writing parameters – e.g. reading out faults – and for reading out information on the characteristics of a parameter such as reading out the min./max. limits etc.

The type of PPO (see next page) used by the PROFIBUS-DP master to communicate with the converter can be configured from the master when the bus system is started up. Which type of PPO is selected depends on the task of the drive in the automation network. The process data are always transmitted. In the drive, they are processed with the highest priority and in the shortest time slots. The process data are used to coordinate the drive with the other units in the automation network, e.g. for power on/off, entering setpoints etc.

With the help of the parameter area, the user can access all the parameters in the converter via the bus system as required. For example, detailed diagnostic information, alarms and so on can be read out. In this way, a higher-level system, (e.g. a PC), can be used to call additional information for visualization of the drive without affecting process data transmission.

The telegrams of cyclical data transfer therefore have the following basic structure:



1) PKW: Parameter identifier value

There are five types of PPO:

- ◆ Useful data **without** a parameter area with two words or six words of process data
- ◆ or useful data **with** a parameter area and two, six or ten words of process data.

PKW				PZD									
PKE	IND	PWE		PZD1 STW1 ZSW1	PZD2 HSW HIW	PZD3	PZD4	PZD5	PZD6	PZD7	PZD8	PZD9	PZD10
1st Word	2nd Word	3rd Word	4th Word	1st Word	2nd Word	3rd Word	4th Word	5th Word	6th Word	7th Word	8th Word	9th Word	10th Word
PPO1													
PPO2													
PPO3													
PPO4													
PPO5													

PKW: Parameter ID value

STW: Control word 1

PZD: Process data

ZSW: Status word 1

PKE: Parameter ID

HSW: Main setpoint

IND: Index

HIW: Main actual value

PWE: Parameter value

Table 8.2-1 Parameter process data object (PPO types)

Dividing the useful data into parameter identifier values and process data enables different tasks to be carried out.

Parameter data area (PKW) With the PKW (parameter identifier value) part of the telegram, any parameter in the converter can be observed and/or altered. The mechanisms of task/reply IDs necessary for this are described later in the chapter "Mechanisms of PKW processing".

Process data area (PZD) With the process data part, control words and setpoints (tasks: master → converter) or status words and actual values (replies: converter → master) are transferred.

The transferred process data only have an effect if the control-word bits, the setpoints, the status words and the actual values are routed in the basic unit in accordance with the chapter "Process data wiring".

The following page gives an overview of typical ways of routing process data to the basic unit. For this routing of the data, the term "process data wiring" is often used.

NOTE

The following process data wiring only applies if a technology board has not been mounted.

If a technology board is used (e.g. T400, T300, T100), the process data wiring in the manual for the technology board is to be used.

Telegram:
Master → Converter

(Setpoint channel)

Combination values for:

16-bit process data

16-/32-bit process data (example)

Alternatives

Process data quantity for:

PPO types 1 and 3

PPO types 2 and 4

PPO type 5

Telegram:
Converter → Master

(Actual-value channel)

Assignment of actual-value
parameters for

16-bit process data

16-/32-bit process data (example)

PZD									
PZD 1	PZD 2	PZD 3	PZD 4	PZD 5	PZD 6	PZD 7	PZD 8	PZD 9	PZD 10
STW1	HSW								
1st word	2nd word	3rd word	4th word	5th word	6th word	7th word	8th word	9th word	10th word

3001	3002	3003	3004	3005	3006	3007	3008	3009	3010
3001	3032		3004	3035		3037		3039	

3001	3032		3004	3005	3036		3038		3010
3001	3002	3003	3004	3035		3007	3038		3010

PZD2										
PZD6										
PZD10										

PZD									
PZD 1	PZD 2	PZD 3	PZD 4	PZD 5	PZD 6	PZD 7	PZD 8	PZD 9	PZD 10
ZSW1	HIW								

P734	P734	P734	P734	P734	P734	P734	P734	P734	P734
P694	P694	P694	P694	P694	P694	P694	P694	P694	P694
i001	i002	i003	i004	i005	i006	i007	i008	i009	i010

P734	P734		P734	P734		P734	P734		P734
P694	P694		P694	P694		P694	P694		P694
i001	i002 = i003		i004	i005 = i006		i007	i008 = i009		i010

Parameters for FC (CU1), VC (CU2) and SC (CU3)

PZD:	Process data	HSW:	Main setpoint
STW:	Control word	HIW:	Main actual value
ZSW:	Status word		

Table 8.2-2 Fixed assignment and combination values

NOTE

If a second CBP is being operated in the converter, then the "8000" connectors will be applicable for the second CBP instead of the "3000" connectors, and parameter P736 will be applicable instead of parameter P734 (see function diagrams for CB/TB boards in Chapter 12).

CBP2 - Free configuration

Extended functionality of the CBP2 in a SIMATIC STEP7 environment with DriveES:

In addition to the five types of PPO, free configuration of the cyclical data is possible.

Up to 16 process data words can be configured, even with a different number of setpoints and actual values. The consistency ranges can be flexibly adjusted.

A parameter area (PKW) can be configured irrespective of the number of process data items.

CBP2, V2.21 and later, standard telegrams

On version V2.21 and later of the CBP2, cyclic data transmission is implemented via standard telegrams in accordance with PROFIdrive profile, version 3.

The CBP2 supports standard telegrams 1 to 6 (cf. Section 8.2.7.3 "Process data interconnection via standard tele").

8.2.2.2 Acyclical data transfer**Extended DP functions**

The PROFIBUS-DP has now been improved to include other methods of data transfer. In addition to cyclical data transfer, the extended PROFIBUS-DP enables the following forms of data transfer as defined in PROFIBUS guidelines No. 2.081 (German) or 2.082 (English):

- ◆ Acyclical data transfer at the same time as cyclical data transfer
- ◆ Alarm processing

Acyclical data transfer enables:

- ◆ the exchange of larger amounts of useful data up to 206 bytes
- ◆ a reduction in the number of peripheral addresses in the SIMATIC by means of relocating the PKW area from cyclical to first acyclical data transfer
- ◆ as a result, also reduction of the bus cycle time due to shorter telegrams in cyclical data transfer
- ◆ simultaneous access by Drive ES (PG/PC) for diagnosis and parameterization by means of the second data transfer

Realization of the extended DP functions

The different masters or the different methods of data transfer are represented in the CBP by corresponding channels (see Fig. 8.2-4):

- ◆ **Cyclical data transfer with a Class 1 master (MSCY_C1)**
Use of DATA-EXCHANGE and the PPO types in accordance with the PROFIdrive profile
- ◆ **Acyclical data transfer with the same Class 1 master (MSAC_C1)**
Use of the PROFIBUS functions, DDLM_READ and DDLM_WRITE
The contents of the transferred data block corresponds to the structure of the parameter area (PKW) in accordance with the USS specification (with data block 100)
or (for CBP2 V2.21 and later only)
the structure of the acyclic parameter channel according to PROFIdrive profile, version 3 (with data block 47).
- ◆ **Acyclical data transfer with start-up tool DriveES (Class 2 master; MSAC_C2)**
DriveMonitor of DriveES-Basic can access parameters and process data in the basic units acyclically.
- ◆ **CBP2: acyclical data traffic with SIMATIC OP (second Class 2 master; MSAC_C2) only**
SIMATIC OP can access parameters in the basic units acyclically.
- ◆ **CBP2 V2.21 and later only:** Instead of DriveES or SIMATIC OP an external master (Class 2 Master) compliant with acyclic parameter channel according to PROFIdrive profile version 3 with data block 47 can also access converter data.

8.2.2.3 Acyclical master class 1, automation (PLC)

MSAC_C1 channel Acyclical communication between the DP master Class 1 (DPM1) and the DP slaves takes place via supplementary service access point 51. In a service sequence, the DPM1 establishes a link to the slave, this link being designated MSAC_C1. Establishment of this link is closely related to the link for cyclical data transfer between the DPM1 and the slaves. Once a link has been established, the DPM1 can conduct cyclical data transfer via the MSCY_C1 link and, at the same time, acyclical data transfer via the MSAC_C1 link.

The MSAC_C1 channel enables READING and WRITING of any of the data blocks in the slave. These data blocks are accessed with the PROFIBUS functions, DDLM_Read and DDLM_Write.

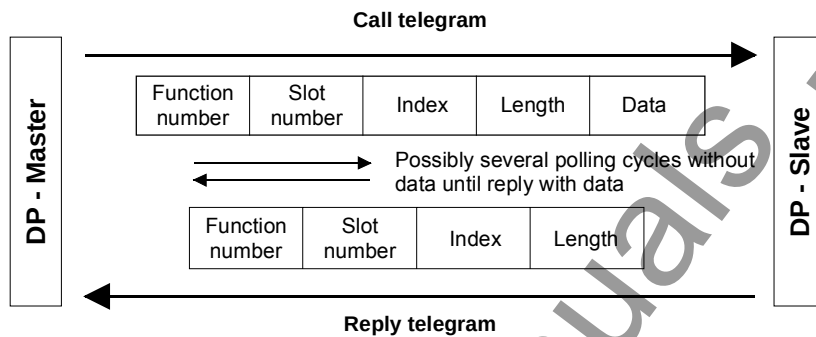
For processing parameters, the CBP supports a data block with the index 100 in slot 2. Because the parameters can only be altered infrequently in comparison to the process data, the parameter area of the telegram can be removed from the fast cyclical channel in order to save bus resources.

NOTE

With the CBP2, version V2.21 and later, a class 1 master automation (PLC) can also utilize acyclic parameter access according to PROFIdrive V3, cf. Section 8.2.4 "PROFIdrive V3: Acyclic parameter accessing with data block 47".

Telegram structure The following illustration shows the telegram structure for data transfer via the acyclical MSAC_C1 channel.

Write function



Read function

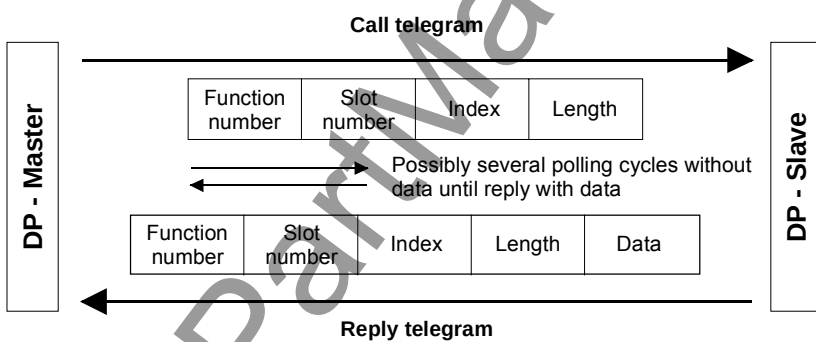


Fig. 8.2-5 Sequence of a Read and Write function

Sequence of a PKW task

The following sequence is necessary for handling a PKW task:

1. With the function `DDLM_Write`, a PKW task is transferred in the data block with the index 100 to the CBP.
2. A positive acknowledgement of `DDLM_Write` is awaited.
3. With the function `DDLM_Read`, the PKW reply is requested by the CBP in the data block with the index 100.
4. The PKW reply to the task is contained in the positive acknowledgement of `DDLM_Read`.

The contents of the data block with the index 100 corresponds to the structure of the PKW area of the telegram in accordance with the USS specification.

With the PKW (parameter identifier value) area, any parameter in the converter can be visualized and/or altered. The mechanisms of task/reply IDs necessary for this are described later in the chapter 8.2.3 "Mechanisms of PKW processing".

In the MSAC_C1 channel, larger amounts of data can be transferred at the same time than by means of PPOs in the cyclical channel. The whole data unit is used exclusively for transmitting parameters.

It offers the same possibilities, however, as in the USS specification, i.e. complete arrays can also be processed with one task (IND = 255). All values of the array are directly transmitted one after the other in a data block. The maximum length of a data block is 206 bytes.

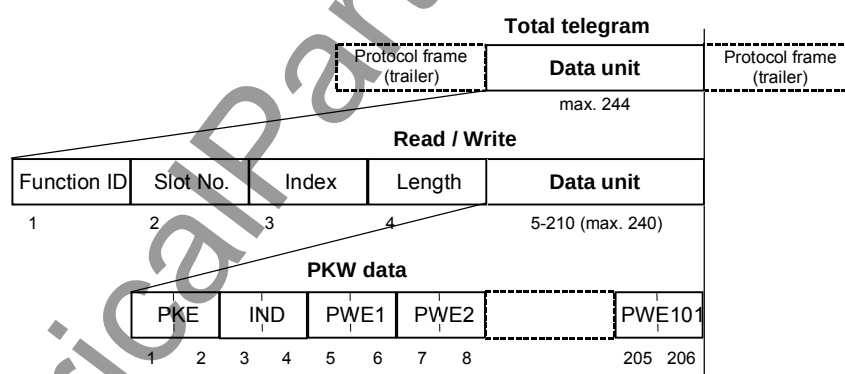


Fig. 8.2-6 *Structure of PKW data in cyclical data transfer*

NOTE

Process data (PZDs) cannot be stipulated via this acyclical MSAC_C1 channel.

**Example for the
SIMATIC S7**

In the SIMATIC S7, the data block with the index 100 corresponds to the data record DS100.

From the SIMATIC S7 side, data can be exchanged via the MSAC_C1 channel with the system functions SFC 58 "WR_REC" (Write Data Record) and SFC 59 "RD_REC" (Read Data Record).

When the system functions are called, the parameter **RECNUM** is to be set to 100.

If the logical address of the CBP is determined by means of SFC 5 "GADR_LGC" (convert geographical address to logical address), the parameters are to be provided with the following when SFC 5 is called:

SUBNETID	=	ID of the planned DP master system in accordance with the hardware configuration	
RACK	=	Node / bus address of the CBP	According to S7 HW configuration
SLOT	=	2	
SUBSLOT	=	0	
SUBADDR	=	0	

The function-block package, DVA_S7 (see also section 8.2.8.2), is a standard method of data exchange between the SIMATIC S7 and the CBP via the acyclical MSAC_C1 channel. The user is provided with a data block as the data interface. This data block has a TRANSMIT MAILBOX and a RECEIVE MAILBOX, thus considerably reducing the expenditure on the application for the user.

8.2.2.4 Acyclical master class 2 - Configuration (DriveES)

MSAC_C2 channel for the Drive ES

The MSAC_C2 channel on the CBP must be reserved for the start-up and service tool Drive ES.

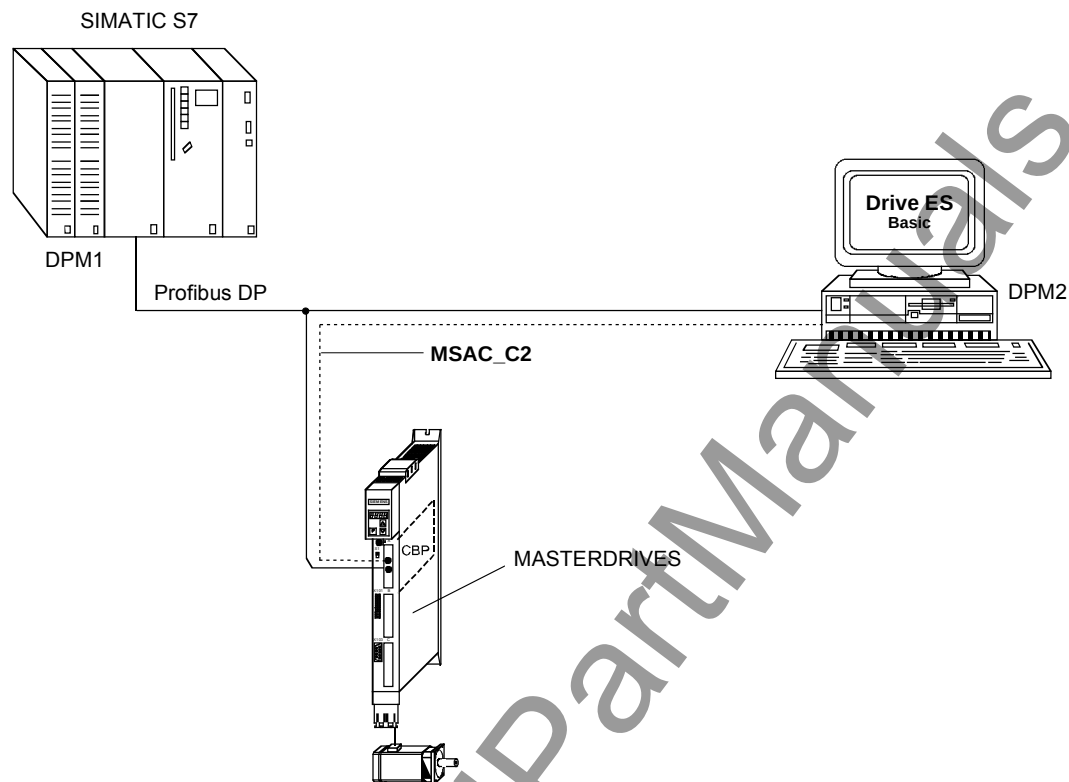


Fig. 8.2-7 Drive ES with Profibus

8.2.2.5 Acyclical master class 2 - Operator control (SIMATIC OP)

Functionality only with CBP2.

With a SIMATIC OP as the PROFIBUS DP master, you can achieve direct access to a drive.

A drive with a CBP2 behaves like a SIMATIC S7 towards a SIMATIC OP. For access to the drive parameters, the following simple illustration applies:

Parameter number = Data block number

Parameter subindex = Data block offset

All SIMATIC OPs and TDs with the final digit 7 are suitable.

ProTool

You can configure SIMATIC OP with "ProTool". The following specific settings for drives are to be entered during configuration with Pro Tool.

Open-loop control

Control units: Protocol always **"SIMATIC S7 - 300/400"**, additional parameters:

Field	Value
Network parameter - Profile	DP
Network parameter - Baud rate	(as selected)
Communications partner - Address	(the PROFIBUS address of the drive)
Communications partner - Slot/rack	Don't care, 0

Variable

Variables: "General" register:

Field	Value
Name	(as selected)
Control unit	(as selected)
Type	Depending on parameter value addresses, e.g.: INT: for I2, O2 DINT: for I4, O4 WORD: for V2, L2
Range	DB
DB (data block number)	Parameter number 1 to 3999
DBB, DBW, DBD (data block offset)	Subindex 0: for non-indexed parameters 1 to 101: for indexed parameters
Length	(not activated)
Acquisition cycle	(as selected)
Number of elements	1
Places after the decimal point	(as selected)

NOTES

- ◆ You can operate a SIMATIC OP together with a drive, irrespective of any automation system which may be present. A simple "point-to-point" connection with only two nodes is possible.
- ◆ The "Variable" OP functions can be used for drives. Other functions cannot be used (e.g. "Messages" or "Recipes").
- ◆ Access is possible to individual parameter values. Access is not possible to whole arrays, descriptions or texts.
- ◆ The parameter values transferred to the OP are the non-standardized internal values of the drive. You can influence the value displayed on the OP with "Functions" in Pro Tool (e.g. "Linear conversion").
- ◆ The diagnostic output on the SIMATIC OP is limited. In the case of unsuccessful attempts at access, the CB diagnostic parameter, r732.22, and the following can help you further. See Section "Diagnosis and Troubleshooting".

8.2.3 Mechanisms for processing parameters via the PROFIBUS

Parameter area (PKW)

With the PKW mechanism (for PPO types 1, 2 and 5 and when the acyclical channels, MSAC_C1 and MSAC_C2, are used), you can perform the following tasks:

- ◆ Handling and visualizing parameters (read/write)
- ◆ Transferring and acknowledging parameter change reports (not realized)

The parameter area always contains at least 4 words.

		Parameter ID (PKE)				1st word
Bit No.:	15	12	11	10	0	
	AK		SPM	PNU		

		Parameter index (IND)				2nd word
Bit No.:	15	8		7	0	
The structure and significance depend on the type of data transfer (see following pages)						

		Parameter value (PWE)				
	Parameter value High				(PWE1)	3rd word
	Parameter value Low				(PWE2)	4th word

AK:	Task ID or reply ID
SPM:	Toggle bit for processing the parameter change report
PNU:	Parameter number

Table 8.2-3 Structure of the parameter area (PKW)

**Parameter ID (PKE),
1st word**

The parameter ID (PKE) is always a 16-bit value.

Bits 0 to 10 (PNU) contain the number of the required parameter.

Bit 11 (SPM) is the toggle bit for parameter change reports.

Bits 12 to 15 (AK) contain the task ID or the reply ID.

With regard to the task telegram (master → converter), the significance of the task ID is given in Table 8.2-4. Task IDs 10 to 15 are specifically for MASTERDRIVES and are not specified in the PROFIBUS-DP profile.

With regard to the reply telegram (converter → master), the significance of the reply ID is given in Table 8.2-5. Reply IDs 11 to 15 are specifically for MASTERDRIVES and are not specified in the PROFIBUS-DP profile. Only certain reply IDs are possible, depending on the task ID. If the reply ID has the value 7 (task cannot be executed), an error number is deposited in parameter value 2 (PWE2) in accordance with Table 8.2-6.

Task ID	Significance	Reply ID	
		positive	negative
0	No task	0	7 or 8
1	Request parameter value	1 or 2	↑
2	Change parameter value (word)	1	
3	Change parameter value (double word)	2	
4	Request description element ¹	3	
5	Change description element (not with CBP)	3	
6	Request parameter value (array) ¹	4 or 5	
7	Change parameter value (array, word) ²	4	
8	Change parameter value (array, double word) ²	5	
9	Request the number of array elements	6	
10	Reserved	-	
11	Change parameter value (array, double word) and store in the EEPROM ²	5	
12	Change parameter value (array, word) and store in the EEPROM ²	4	
13	Change parameter value (double word) and store in the EEPROM	2	
14	Change parameter value (word) and store in the EEPROM	1	↓
15	Read or change text (not with CBP)	15	7 or 8

Table 8.2-4 Task IDs (master → converter)

Reply ID	Significance
0	No reply
1	Transfer parameter value (word)
2	Transfer parameter value (double word)
3	Transfer description element ¹
4	Transfer parameter value (array, word) ²
5	Transfer parameter value (array, double word) ²
6	Transfer the number of array elements
7	Task cannot be executed (with error number)
8	No operator change rights for the PKW interface
9	Parameter change report (word)
10	Parameter change report (double word)
11	Parameter change report (array, word) ²
12	Parameter change report (array, double word) ²
13	Reserved
14	Reserved
15	Transfer text (not with CBP)

¹ The required element of the parameter description is specified in IND (2nd word)

² The required element of the indexed parameter is specified in IND (2nd word)

Table 8.2-5 Reply IDs (converter -> master)

Example

Source for the ON/OFF1 command (control word 1, bit 0):
P554 (=22A Hex)

Change parameter value (array, word) and store in the EEPROM

Parameter ID (PKE)																1st word	
Bit No.:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	AK				SPM		PNU										
	1	1	0	0	0	0	1	0	0	0	1	0	1	0	1	0	Binary value
	C				2		A										HEX value

- ◆ Bits 12 to 15: Value = 12 (= "C" Hex); change parameter value (array, word) and store in the EEPROM
- ◆ Bits 0 to 11: Value = 554 (= "22A" Hex); parameter number without set bit for the parameter change report

No.	Significance	
0	Non-admissible parameter No. (PNU)	If the PNU does not exist
1	Parameter value cannot be changed	If the parameter is a visualization parameter
2	Upper or lower limit exceeded	–
3	Erroneous subindex	–
4	No array	–
5	Incorrect data type	–
6	Setting not allowed (can only be reset)	–
7	Description element cannot be changed	Generally not possible for MASTERDRIVES
11	No operator control rights	–
12	Key word missing	Drive converter parameter "access key" and/or "parameter special access" not correctly set
15	No text array available	–
17	Task cannot be executed due to operating status	Drive converter status does not permit the present task
20	Non-admissible value	Modification access using a value which is within tolerance, but non-admissible for other permanently valid reasons
21	Response too long	The current response is longer than the maximum transmittable length
22	Non-admissible parameter address	Non-admissible or unsupported value for attribute, number of elements, parameter number, subindex or a combination of the above
23	Non-admissible format	Write task: Parameter data in a non-admissible or unsupported format
24	Inconsistent number of values	Write task: Number of values of parameter data does not tally with the number of elements in the parameter address
101	Parameter number deactivated at present	Specific to MASTERDRIVES
102	Channel width too small	Specific to MASTERDRIVES: only for short channels
103	Incorrect number of PKWs	Specific to MASTERDRIVES: only for G-SST1/2 and SCB interface (USS)
104	Parameter value not admissible	Specific to MASTERDRIVES
105	The parameter is indexed	e.g. task: "PWE, change word" for indexed parameters
106	Task not implemented	

Table 8.2-6 Error numbers for the reply "Task cannot be executed" (drive converter parameters)

Comment on error number 103

Error number 103 is only relevant to the G-SST1, 2 interface and the SCB interface. It is transferred in the following two cases:

- ◆ If the task involves indices of an indexed parameter (task index equal to 255) or the complete parameter description is requested and a variable telegram length has not been parameterized.
- ◆ If the set task is too small for the parameterized number of PKW data in the telegram (e.g. the double word and the PKW number is changed to 3 (words)).

Comment on error number 104

This error number is transferred if the parameter value which is to be adopted has not been assigned a function in the drive converter or cannot be adopted at the time of the change for internal reasons (although it lies within the limits).

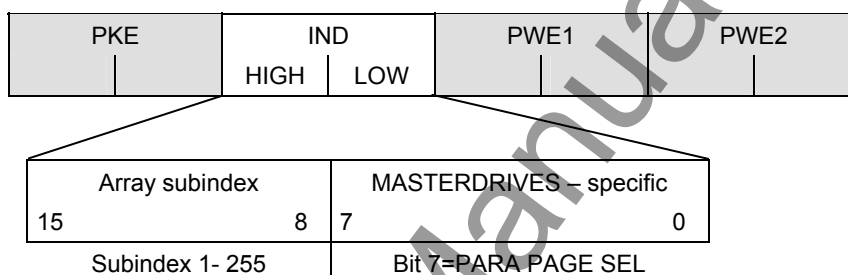
This error number always occurs, for example, when only values explicitly entered in a table are valid for a parameter value and are not transferred exactly (e.g. the number of PKW data for the USS interfaces for which only the explicit values 0, 3, 4 and 127 are allowed).

Parameter index (IND) 2nd word

The assignment of the index (**IND**) is to be regarded as a special feature or difference between what is specified in the PPOs and what is specified for the acyclical channels MSAC_C1 and MSAC_C2.

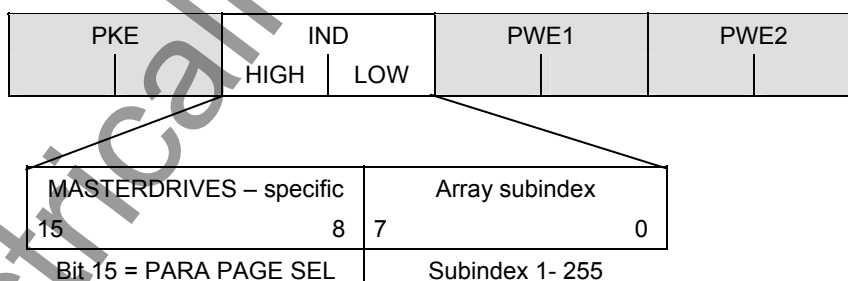
The array sub-index (also designated in shorter form as the sub-index in the PROFIBUS profile) is an 8-bit value and, during cyclical data transfer, is transferred in the most significant byte (bits 8 to 15) of the parameter index (IND). The least significant byte (bits 0 to 7) is not defined in the profile DVA. In the PPO of the CBP, the least significant byte of the parameter index is used in order to be able to address additional technology parameters or parameters of free components in the MASTERDRIVES by means of parameter page selection.

Structure of IND with cyclical communication by means of PPOs



The array subindex is an 8-bit value and, with acyclical data transfer (MSAC_C1), is always transferred in the least significant byte (bits 0 to 7) of the parameter index (IND). The function of parameter-page selection for additional technology parameters or parameters of free components in the MASTERDRIVES is assumed here by the most significant byte (bits 8 to 15) of the parameter index. This structure corresponds to the stipulations of the USS specification.

Structure of IND with acyclical communication via MSAC_C1



The function of the IND

For an indexed parameter, if the subindex in a task is transferred with the values between 1 and 254, the required index of the parameter is transferred. The significance of the individual indices of the parameter can be found in the "Parameter List" of the operating instructions for the converter.

When a description element is being processed, the number of the required element is transferred. The significance of the description elements is given in the PROFIBUS profile "Variable-speed drives", PROFIdrive version V2 (PNO: Order No. 3071).

The value 255 for the array subindex is of special importance. If the array subindex is transferred with 255, all indices of an indexed parameter are transferred simultaneously in one data block.

This function is useful only for acyclical data transfer via MSAC_C1. The transferred data block has the same structure as in the USS specification (see Fig. 8.2-6). The maximum size of a data block is 206 bytes.

The bit for parameter page selection has the following effect:

If this bit is equal to 1, the parameter number (PNU) transferred in the PKW task is provided with an offset of 2000 in the CBP and then passed on.

Parameter designation (acc. to parameter list)	Serial parameter number	Required addressing of the parameter via PROFIBUS		
		PNU [decimal]	PNU [hex.]	Bit (*)
P000 - P999 (r000 - r999)	0 - 999	0 - 999	0 - 3E7	= 0
H000 - H999 (d000 - d999)	1000 - 1999	1000 - 1999	3E8 - 7CF	= 0
U000 - U999 (n000 - n999)	2000 - 2999	0 - 999	0 - 3E7	= 1
L000 - L999 (c000 - c999)	3000 - 3999	1000 - 1999	3E8 - 7CF	= 1

*) Parameter page selection

Example

Source for the ON/OFF command (control word 1, bit 0):
P554 (=22A Hex)

Change parameter value of index 1 (structure of the IND according to PPO)

Bit No.:	Parameter index (IND)																2nd word
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	Binary value
								1									HEX value

- ◆ Bits 8 to 15: Index of parameter P554
- ◆ Bits 0 to 7: Value = 0

Parameter value (PWE) 3rd and 4th words

The parameter value (PWE) is always transferred as a double word (32 bits). In a PPO telegram, only one parameter value can be transferred.

A 32-bit parameter value is composed of PWE1 (most significant word, 3rd word) and PWE2 (least significant word, 4th word).

A 16-bit parameter value is transferred in PWE2 (least significant word, 4th word). In this case, you must set PWE1 (most significant word, 3rd word) to 0 in the PROFIBUS-DP master.

**Example for
CUPM/CUMC/CUVC**

Source for the ON/OFF command (control word 1, bit 0): P554 (= 22A Hex)

Change parameter value of index 1 to the value 3100

Parameter value (PWE)					
Bit No.:	31	24	23	16	
	0	0	0	0	3rd word (PWE1) (hex)
Bit No.:	15	8	7	0	4th word (PWE2) (hex)
	3	1	0	0	

- ◆ Bits 0 to 15: Parameter value for 16-bit parameter or low component for 32-bit parameter
- ◆ Bits 16 to 31: Value = 0 for 1-bit parameter or high component for 32-bit parameter

Rules for task/reply processing

- ◆ A task or a reply can only relate to one parameter value.
- ◆ The master must repeat a task until it receives the appropriate reply.
- ◆ The master identifies the reply to a task which has been set:
 - By evaluating the reply ID
 - By evaluating the parameter number, PNU
 - If necessary, by evaluating the parameter index, IND
 - If necessary, by evaluating the parameter value, PWE.
- ◆ The task must be sent complete in one telegram; telegrams with split tasks are not permissible. The same applies to the reply.
- ◆ With regard to reply telegrams which contain parameter values (actual values), the slave (CBP) always replies with the latest current values if the telegram is repeated.
- ◆ If the PKW interface requires no information during cyclical operation (only PZD data are important), the "No task" task must be sent.

WARNING

When you change the initialization function of software version V1.3x to V1.40 and higher, or VC firmware from 3.22 to 3.23 and higher, the behavior of the converter also changes (reverting to the behavior of software versions V1.2x and lower again) as follows:

If the electronics supply is switched off on a converter that is in state "READY" and is connected to an automation system via a field bus (PROFIBUS, CAN, DEVICE-NET, or CC-Link), this causes a fault message for this converter in the automation system.

If the automation system nevertheless sends a control word STW1 with valid authorization (bit 10 = 1) and a pending ON command (bit 0 = 1) to this converter, this can cause the converter to switch on and go straight into "OPERATION" state when the electronics supply is connected at the converter.

8.2.4 PROFIdrive V3: Acyclic parameter accessing with data block 47

NOTE

Acyclic parameter accessing with data block 47 is supported by the CBP2 with firmware version V2.23 and later.

A detailed description of acyclic parameter accessing with data block 47 can be found in PROFIBUS Profile, PROFIdrive (PNO: Order No. 3172).

General properties

- ◆ Compatibility with PKW tasks in accordance with PROFIdrive profile version 2
- ◆ 16-bit wide address for each parameter number and subindex
- ◆ Transfer of complete arrays or areas thereof, or the entire parameter description
- ◆ Transfer of different parameters in one access operation (multi-parameter tasks)
- ◆ Only **one** parameter task is processed at a time (no pipelining)
- ◆ A parameter task/response must fit into one data block (max. 240 bytes). Tasks/responses are **not split** over several data blocks. The maximum length of data blocks can be less than 240 bytes as a result of slave property or bus configuration.
- ◆ **"Multi-parameter"** tasks are defined for optimized, simultaneous access to different parameters (e.g. HMI screen contents).
- ◆ Data block 47 can be processed by acyclical channels MSAC_C1 and MSAC_C2.

Subindex 0

The definition of an array has been changed in IEC 61158 as compared to the definition in EN 50170.

The PROFIdrive profile version 2 is compliant with EN 50 170, according to which the subindex of an indexed parameter or array begins with index 1. In the current IEC standard 61158, access to an indexed parameter or array begins with the index 0.

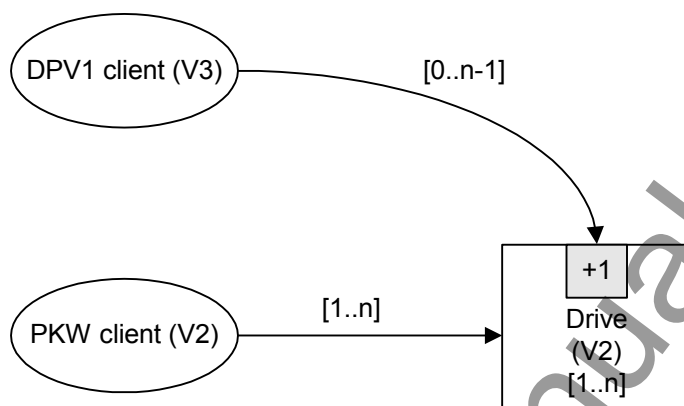
As a consequence, the parameter model and the DPV1 parameter channel had to be adapted in PROFIdrive profile version 3 so as to ensure compliance with the IEC standard.

Compatibility with the PKW mechanism in PROFIdrive profile version 2

MASTERDRIVES continues to use the parameter model according to PROFIdrive V2 at the internal interface. MASTERDRIVES can be accessed via data block 47 as a DPV1 client with the CBP2. For tasks using DB47, the CBP2 thus adds an offset of 1 to the parameter subindex.

The cyclical parameter access via PKW and the acyclical parameter access with data block 100 can continue to be used without any changes.

MASTERDRIVES MC with parameter model to PROFIdrive profile version 2. In combination with the CBP2, DPV1 can be utilized in accordance with PROFIdrive profile version 3.



Special features / restrictions

- ◆ Access operations to simple parameters (i.e. parameters without indices) must be identified by "No. of elements" = 0.
- ◆ Changing the sub-areas of an array is not supported by the CBP2, i.e. it is possible to transfer a write task either for **one** index or for **all** indices. To alter a complete parameter array, the number of values has to be equal to or greater than the size of the array.
- ◆ The editing of texts or descriptions is not supported.
- ◆ Reading of several or all texts from a text array via a parameter task is not supported, i.e. only one text from one text array (subindex) can be read with one parameter task.

8.2.4.1 Comparison between parameter tasks to PROFIdrive version 2 and 3

	PKW to PROFIdrive profile V2	DPV1 parameter tasks to PROFIdrive profile V3	Remarks
Task reference	-	New! 8-bit	Task/response identification
Task identifier	Request/change value/des./texts 4-bit	Request/change 8-bit	Distinction value/description/text as additional attribute
No. of parameters	-	New! 8-bit	Multi-parameter tasks
Parameter number	0..1999 (11 bits)	Content as for PKW 16-bit	Parameter number = 0 not allowed
Subindex	1..255 (8 bits)	Content as for PKW - 1 16-bit	Offset in subindex due to modified array definition: DPV1 subindex = PKW subindex - 1
No. of elements	- (always "1")	New 8-bit	Access to simple parameters (nonindexed parameters) is defined in DB47 with "No. of elements" = 0.
Attribute	-	New 8-bit	Distinction value/description/text
Total length	2 words	5 words	

8.2.4.2 Example of "Request parameter value", simple

Parameter task:

Task header	Task reference	Task identifier = Request parameter	Offset 0
	Axis = 0	No. of parameters = 1	2
Parameter address	Attribute = value	No. of elements = 0 (!)	4
	Parameter number		
	Subindex = 0		10

Positive parameter response with word:

Response header	Task ref. mirrored	Response identifier = Request parameter (+)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = word	No. of values = 1	4
	Value		6
			8

Positive parameter response with double word:

Response header	Task ref. mirrored	Response identifier = Request parameter (+)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = double word	No. of values = 1	4
	Value		6
			10

Negative parameter response:

Response header	Task ref. mirrored	Response identifier = Request parameter (-)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = error	No. of values = 1	4
	Error value		6
			8

NOTE

The coding of the fields in the parameter task/response task is described in Section 8.2.4.11 "Coding in the parameter task according to PROFIdrive Version 3" .

8.2.4.3 Example of "Change parameter value", simple

Parameter task:

Task header	Task reference	Task identifier = Change parameter	Offset 0
	Axis = 0	No. of parameters = 1	2
Parameter address	Attribute = value	No. of elements = 0 (!)	4
	Parameter number		
	Subindex = 0		
Parameter value	Format = word	No. of values = 1	10
	Value		12
			14

Positive parameter response:

Response header	Task ref. mirrored	Response identifier = Change parameter (+)	0
	Axis mirrored	No. of parameters = 1	2
			4

Negative parameter response:

Response header	Task ref. mirrored	Response identifier = Change parameter (-)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = error	No. of values = 1	4
	Error value		6
			8

8.2.4.4 Example of "Request parameter value", more than one array element

Parameter task:

Task header	Task reference	Task identifier = Request parameter	Offset 0
	Axis = 0	No. of parameters = 1	2
Parameter address	Attribute = value	No. of elements = 5	4
	Parameter number		
	Subindex = 0		10

Positive parameter response:

Response header	Task ref. mirrored	Response identifier = Request parameter (+)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = word	No. of values = 5	4
	Value 1		6
	Value 2		
	Value 3		
	Value 4		
	Value 5		16

Negative parameter response:

Response header	Task ref. mirrored	Response identifier = Request parameter (-)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = error	No. of values = 1	4
	Error value		6
			8

8.2.4.5 Example of "Change parameter value", more than one array element

NOTE

Changing the sub-areas of an array is not supported by the CBP2, in other words, it is possible to transfer a write task either for one index or for all indices. To alter a complete parameter array the number of values must be equal to or greater than the array size.

The following example shows a write operation to one parameter with 5 subindices.

Parameter task:

			Offset
Task header	Task reference	Task identifier = Change parameter	0
	Axis = 0	No. of parameters = 1	2
Parameter address	Attribute = value	No. of elements = 5	4
	Parameter number		
	Subindex = 0		
Parameter value	Format = word	No. of values = 5	10
	Value 1		12
	Value 2		
	Value 3		
	Value 4		
	Value 5		
			22

Positive parameter response:

Response header	Task ref. mirrored	Response identifier = Change parameter (+)	0
	Axis mirrored	No. of parameters = 1	2
			4

Negative parameter response:

Response header	Task ref. mirrored	Response identifier = Change parameter (-)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = error	No. of values = 1	4
	Error value		6
			8

8.2.4.6 Example of "Request parameter value", multi-parameter

Parameter task:

Task header	Task reference	Task identifier = Request parameter	0
	Axis = 0	No. of parameters = 3	2
1 st parameter address	Attribute = value	No. of elements = 1	4
	Parameter number		
	Subindex = 7		
2 nd parameter address	Attribute = value	No. of elements = 100	10
	Parameter number		
	Subindex = 0		
3 rd parameter address	Attribute = value	No. of elements = 2	16
	Parameter number		
	Subindex = 13		

Parameter response (+): All part accesses o.k.

Response header	Task ref. mirrored	Response identifier = Request parameter (+)	0
	Axis mirrored	No. of parameters = 3	2
1 st parameter value(s)	Format = word	No. of values = 1	4
	Value		6
2 nd parameter value(s)	Format = word	No. of values = 100	8
	Value 1		10
	Value 2		
	...		
	Value 100		
3 rd parameter value(s)	Format = double word	No. of values = 2	210
	Value1		212

	Value2		

Parameter response (-): First and third part access o.k., second part access errored

Response header	Task ref. mirrored	Response identifier = Request parameter (-)	0
	Axis mirrored	No. of parameters = 3	2
1 st parameter value(s)	Format = word	No. of values = 1	4
	Value		6
2 nd parameter value(s)	Format = error	No. of values = 1	8
	Error value		10
3 rd parameter value(s)	Format = double word	No. of values = 2	12
	Value1		14

	Value2		

			22

8.2.4.7 Example of "Change parameter value", multi-parameter

Parameter task:

Task header	Task reference	Task identifier = Change parameter	Offset 0
	Axis = 0	No. of parameters = 3	2
1 st parameter address	Attribute = value	No. of elements = 1	4
	Parameter number		
	Subindex = 7		
2 nd parameter address	Attribute = value	No. of elements = 100	10
	Parameter number		
	Subindex = 0		
3 rd parameter address	Attribute = value	No. of elements = 2	16
	Parameter number		
	Subindex = 0		
1 st parameter value(s)	Format = word	No. of values = 1	22
	Value		24
2 nd parameter value(s)	Format = word	No. of values = 100	26
	Value 1		28
	Value 2		
	...		
	Value 100		
3 rd parameter value(s)	Format = double word	No. of values = 2	228
	Value1		230
	Value2		
			238

Parameter response (+): All part access o.k.

Response header	Task ref. mirrored	Response identifier = Change parameter (+)	0
	Axis mirrored	No. of parameters = 3	2
			4

Parameter response (-): First and third part access o.k., second part access errored

Response header	Task ref. mirrored	Response identifier = Change parameter (-)	0
	Axis mirrored	No. of parameters = 3	2
1 st parameter value(e)	Format = zero	No. of values = 0	4
2 nd parameter value(e)	Format = error	No. of values = 2	6
	Error value		8
	Errored subindex		10
3 rd parameter value(e)	Format = zero	No. of values = 0	12
			14

8.2.4.8 Request description, individual

Parameter task:

Task header	Task reference	Task identifier = Request parameter	Offset 0
	Axis = 0	No. of parameters = 1	2
Parameter address	Attribute = description	No. of elements = 1	4
	Parameter number		
	Subindex = element No. 1)		10

Positive parameter response with word (e.g. ID code):

Response header	Task ref. mirrored	Response identifier = Request parameter (+)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = word	No. of values = 1	4
	Value		6
			8

Positive parameter response with text:

Response header	Task ref. mirrored	Response identifier = Request parameter (+)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = byte	No. of values = 16	4
	Byte 1	Byte 2	6
	...	...	
	Byte 15	Byte 16	22

Negative parameter response:

Response header	Task ref. mirrored	Response identifier = Request parameter (-)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = error	No. of values = 1	4
	Error value		6
			8

1) For parameter description see PROFIBUS Profile, PROFIdrive (PNO Order No. : 3.172)

8.2.4.9 Request description, total

Parameter task:

Task header	Task reference	Task identifier = Request parameter	Offset 0
	Axis = 0	No. of parameters = 1	2
Parameter address	Attribute = description	No. of elements = 0	4
	Parameter number		
	Subindex = 0 (!)		10

Positive parameter response:

Response header	Task ref. mirrored	Response identifier = Request parameter (+)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = byte	No. of values = (Bytes)	4
	ID code		6
	(etc.)		
	...		
	...	...	
			6 + description

Negative parameter response:

Response header	Task ref. mirrored	Response identifier = Request parameter (-)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = error	No. of values = 1	4
	Error value		6
			8

8.2.4.10 Request text, individual

Parameter task:

Task header	Task reference	Task identifier = Request parameter	Offset 0
	Axis = 0	No. of parameters = 1	2
Parameter address	Attribute = text	No. of elements = 1	4
	Parameter number		
	Subindex = parameter subindex -1		10

Positive parameter response:

Response header	Task ref. mirrored	Response identifier = Request parameter (+)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = byte	No. of values = 16	4
	Byte 1	Byte 2	6
	...	...	
	Byte 15	Byte 16	22

Negative parameter response:

Response header	Task ref. mirrored	Response identifier = Request parameter (-)	0
	Axis mirrored	No. of parameters = 1	2
Parameter value	Format = error	No. of values = 1	4
	Error value		6
			8

8.2.4.11 Coding in the parameter task according to PROFIdrive Version 3

Field	Data type	Values	Comment
Task reference	Unsigned8	0x00 reserved 0x01...0xFF	
Task identifier	Unsigned8	0x00 Reserved 0x01 Request parameter 0x02 Change parameter 0x03...0x3F reserved 0x40...0x7F manufacturer-specific 0x80...0xFF reserved	
Response identifier	Unsigned8	0x00 reserved 0x01 Request parameter (+) 0x02 Change parameter (+) 0x03...0x3F reserved 0x40...0x7F manufacturer-specific 0x80 Request parameter (-) 0x81 Change parameter (-) 0x82 reserved 0x83...0xBF manufacturer-specific 0xC0...0xFF	
Axis	Unsigned8	0x00...0xFF Number 0...255	
Number of parameters	Unsigned8	0x00 reserved 0x01...0x25 Number 1...37 0x26...0xFF reserved	Limitation by DPV1 telegram length
Attribute	Unsigned8	0x00 reserved 0x10 Value 0x20 Description 0x30 Text 0x40...0x70 reserved 0x80...0xF0 manufacturer-specific	The four lower bits are reserved for a (future) expansion of "Number of elements" to 12 bits.
Number of elements	Unsigned8	0x00 Special function 0x01...0x75 Number 1...117 0x76...0xFF reserved	Limitation by DPV1 telegram length
Parameter number	Unsigned16	0x0000 Reserved 0x0001...0xFFFF Number 1...65535	
Subindex	Unsigned16	0x0000...0xFFFF Number 1...65535	
Format	Unsigned8	0x00 Reserved 0x01...0x36 Data types 0x37...0x3F reserved 0x40 Zero 0x41 Byte 0x42 Word 0x43 Double word 0x44 Error 0x45...0xFF reserved	The write end preferably enters "correct" data types ¹⁾ ; in exchange, byte, word, double word are also possible. The read end must be able to interpret all values.
Number of values	Unsigned8	0x00...0xEA Number 0...234 0xEB...0xFF reserved	Limitation by DPV1 telegram length
Error number	Unsigned16	0x0000...0x00FF Error numbers (see following table)	The higher byte is reserved.

1) See PROFIBUS Profile, PROFIdrive (PNO Order No.: 3.172)

Table 8.2-7 Coding of the fields in parameter task/response

Error number	Meaning	Use	Additional info
0x00	Non-permissible parameter number	Access to parameter that does not exist	0
0x01	Parameter value cannot be changed	Change access to a parameter value that cannot be changed	Subindex
0x02	Lower and upper value limit exceeded	Change access with value outside the value limits	Subindex
0x03	Incorrect subindex	Access to subindex that does not exist	Subindex
0x04	No array	Access with subindex to non-indexed parameter	0
0x05	Wrong data type	Change access with value that does not correspond with the data type of the parameter	0
0x06	No setting permitted (can only be reset)	Change access with value not equal to 0, where this is not permitted	Subindex
0x07	Description element cannot be changed	Change access to description element that cannot be changed	Subindex
0x08	Reserved	(PROFIdrive Profile V2: PPO write request not available in Information Report)	-
0x09	Description data not available	Access to description that does not exist (parameter value available)	0
0x0A	Reserved	(PROFIdrive Profile V2: Access group incorrect)	-
0x0B	No parameter change rights	Change access where there are no parameter change rights	0
0x0C	Reserved	(PROFIdrive Profile V2: Password wrong)	-
0x0D	Reserved	(PROFIdrive Profile V2: Text in cyclical traffic cannot be read)	-
0x0E	Reserved	(PROFIdrive Profile V2: Text in cyclical traffic cannot be read)	-
0x0F	No text array available	Access to text array that does not exist (parameter value available)	0
0x10	Reserved	(PROFIdrive Profile V2: No PPO Write)	-
0x11	Task cannot be performed due to operating state	Access is not possible due to temporary reasons which have not been specified in more detail	0
0x12	Reserved	(PROFIdrive Profile V2: Miscellaneous error)	-
0x13	Reserved	(PROFIdrive Profile V2: Data cannot be read in cyclical traffic)	-
0x14	Non-permissible value	Change access with value that lies within the value limits but is not acceptable for other permanent reasons (parameter with defined individual values)	Subindex

Error number	Meaning	Use	Additional info
0x15	Response too long	The length of the current response exceeds the maximum transferable length	
0x16	Non-permissible parameter address	Non-permissible or non-supported value for attribute, number of elements, parameter number or subindex or a combination	
0x17	Non-permissible format	Write task: non-permissible or non-supported parameter data format	
0x18	Number of values not consistent	Write task: Number of values of the parameter data do not correspond with the number of elements in the parameter address	
...			
bis 0x64	Reserved	-	-
0x65...0xFF	Manufacturer-specific (see also 8.2.10.6 "CBP2 diagnostic parameters", PKW error numbers)	-	-

Table 8.2-8 Error numbers in DPV1 parameter responses

8.2.5 Mounting methods / CBP slots

NOTE

The CBP can be directly built into Compact PLUS units. In all other types of unit in this series, it is mounted on the CUPM, CUMC or CUVC or it can be connected in the electronics box with an adaptation board.

8.2.5.1 CBP mounting slots in MC Compact PLUS units

NOTE

You can mount the CBP optional board (Communications board PROFIBUS) in any slot. Bear in mind, however, that an encoder board always needs slot C.

Position of the slots

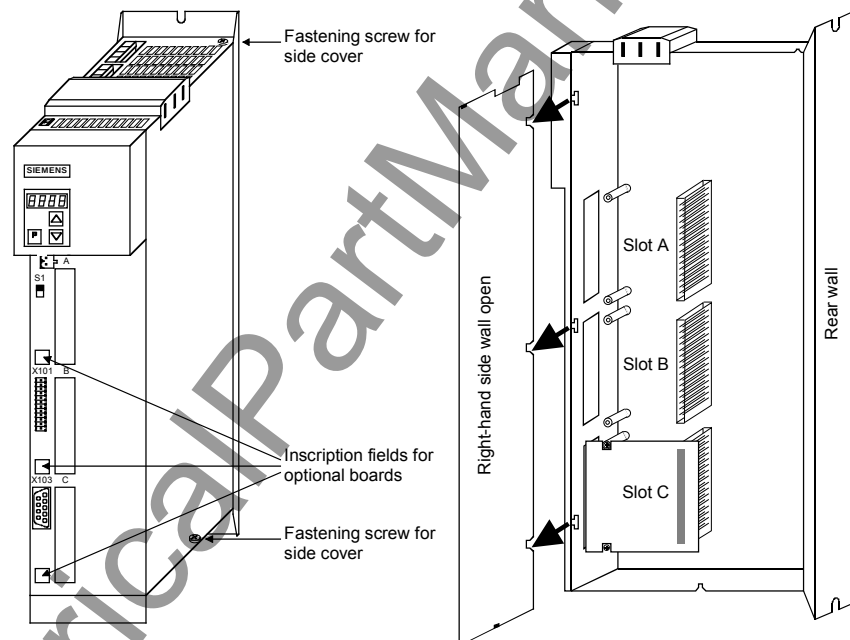


Fig. 8.2-8 Position of the slots (with side wall on the right removed)

DANGER



Due to the DC link capacitors, hazardous voltages are still present in the converter up to 5 minutes after it has been disconnected from the power supply. Before opening the converter, wait until the capacitors have completely discharged.

A maximum of two CBPs can be operated in the Compact PLUS type unit. The following configurations are defined (see function diagrams in Chapter 12):

- ◆ If two CBPs are inserted, the CBP which is inserted into the slot with the lower slot letter is considered the **first** CB/TB.
- ◆ If two CBPs are inserted, the CBP which is inserted into the slot with the higher slot letter is considered the **second** CB/TB.

8.2.5.2 CBP slots in Compact units and chassis-type units with the CUs of function classes Motion Control Performance 2 (CUPM), Motion Control (CUMC) and Vector Control (CUVC)

Slots

In the electronics box of Compact and chassis-type converters and inverters, there are up to six slots available for mounting an optional board. The slots are designated with the letters A to G. There is no slot B, however, in these types of unit; it is only used in Compact PLUS type units.

If you wish to use slots D to G, you must first mount the LBA (Local Bus Adapter, Order No. 6SE7090-0XX84-4HA0) and the corresponding adaptation board ADB (Order No. 6SX7010-0KA00).

NOTE

You can operate the CBP optional board (Communications board PROFIBUS) in any slot. Bear in mind, however, that an encoder board always needs slot C and that the LBA has to use a particular sequence of assigning slots.

The CBP can be mounted on the adaptation board in both slots, i.e. at the BOTTOM and/or at the TOP.

Position of the slots

The slots are located at the following positions:

◆ Slot A	CU board	Top
◆ Slot C	CU board	Bottom
◆ Slot D	Adaptation board in mount. pos. 2	Top
◆ Slot E	Adaptation board in mount. pos. 2	Bottom
◆ Slot F	Adaptation board in mount. pos. 3	Top
◆ Slot G	Adaptation board in mount. pos. 3	Bottom

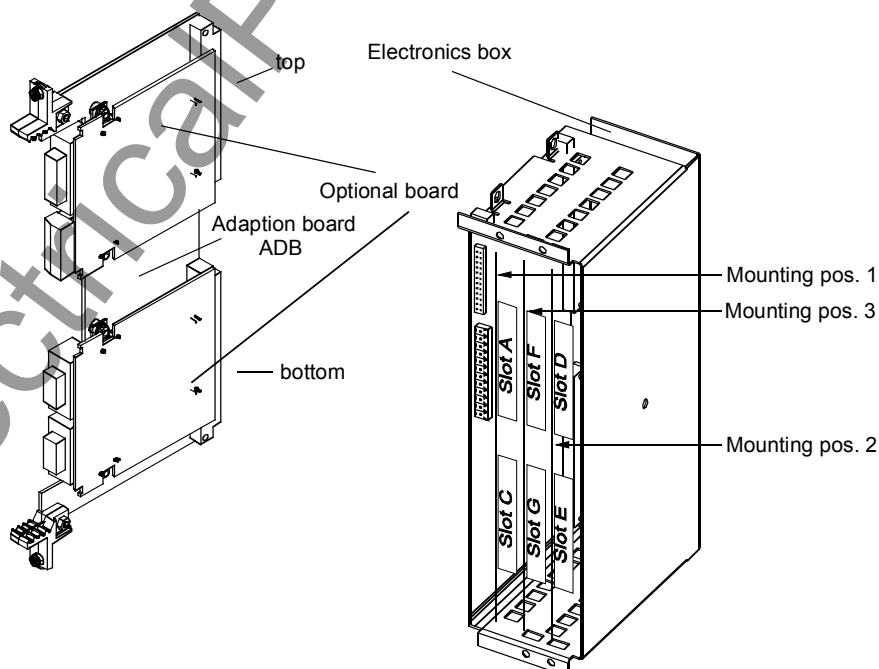


Fig. 8.2-9 Adaptation board with optional boards and position of the slots for Compact units and chassis-type units

DANGER

Due to the DC link capacitors, hazardous voltages are still present in the converter up to 5 minutes after it has been disconnected from the power supply. Before opening the converter, wait until the capacitors have completely discharged.

For technical reasons, certain sequences for assigning the slots are stipulated for the LBA.

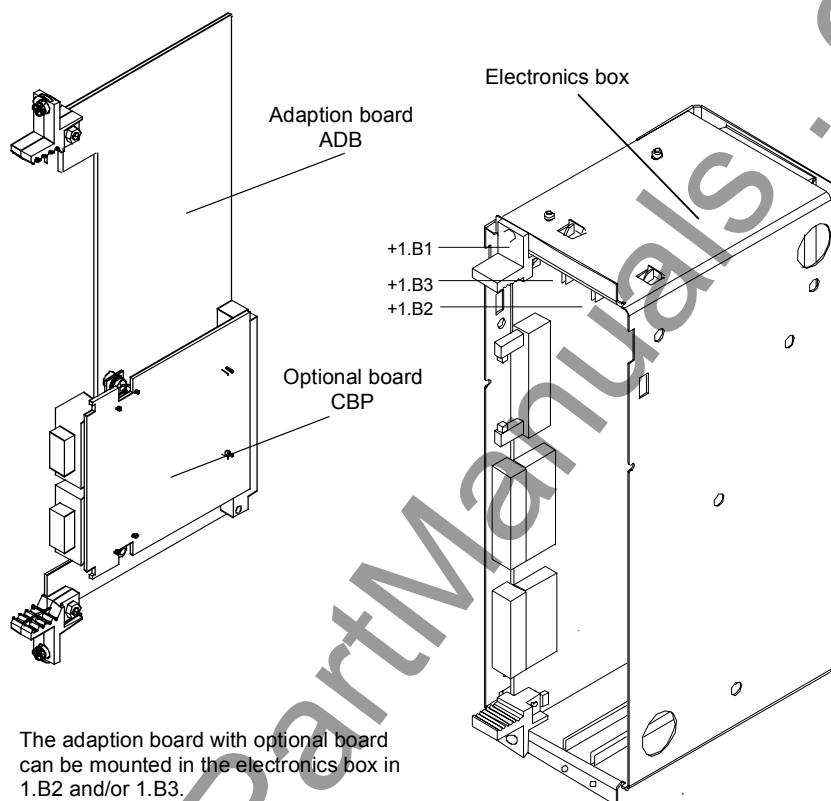
If only one adaptation board with optional boards is inserted in the electronics box, it must always be inserted in slot +1.B2 (ON THE RIGHT), i.e. mounting position 2.

If a technology board T100 / T300 or T 400 is inserted in the electronics box in addition to the adaptation board with CBP, it must be inserted in slot +1.B2 (mounting position 2). In this case, the adaptation board with CBP is inserted in slot +1.B3 (mounting position 3).

A maximum of either two CBPs or one CBP plus one T100/T300/T400 technology board can be operated in the electronics box of the converter. The following configurations are defined (see function diagrams in Chapter 12):

- ◆ The CBP is regarded as the first CB/TB if one of the following configurations exist:
 - Exactly one CBP is inserted in slots A to G on the electronics box and no T100/T300/T400 technology board is inserted.
 - If two CBPs are inserted, the CBP which is inserted in the slot with the lower slot letter.
- ◆ The CBP is regarded as the second CB/TB if one of the following configurations is present:
 - A T100/T300/T400 technology board is inserted and the CBP in the electronics box is inserted in slots A to G.
 - In the case of two CBPs, the one inserted in the slot with the higher slot letter.

8.2.5.3 CBP slots in Compact and chassis-type units with the CUs of function classes FC (CU1), VC (CU2) or SC (CU3)



The adaption board with optional board can be mounted in the electronics box in 1.B2 and/or 1.B3.

Fig. 8.2-10 Electronics box with free slots (+1.B2 and +1.B3) and adaptation board with CBP

On the adaptation board ADB (Order No. 6SX7010-0KA00), **only one** CBP can be mounted in slot X198, i.e. at the BOTTOM.

If the CBP is mounted with adaptation board, the LBA (Local Bus Adapter, LBA, Order No. 6SE7090-0XX84-4HA0) must first be mounted.

NOTE

If only one optional board is used, it must always be inserted in slot +1.B2 (on the RIGHT) in the electronics box.

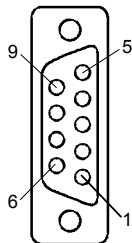
If, in addition to the CBP, a technology board (T100 / T300 or T400) is inserted in the electronics box, it must be inserted in slot +1.B2. In this case, the CBP is inserted in slot +1.B3.

8.2.6 Connecting up the CBP to the PROFIBUS

8.2.6.1 Assignment of plug-in connector X448

Connecting up

The CBP optional board has a 9-pin Sub-D socket (X448) which is provided for connecting the CBP to the PROFIBUS system. The connections are short-circuit proof and floating.



Pin	Designation	Significance	Area
1	SHIELD	Ground connection	
2	-	Not connected	
3	RxD/TxD-P	Receive/transmit data P (B/B')	RS485
4	CNTR-P	Control signal	TTL
5	DGND	PROFIBUS data reference potential (C/C')	
6	VP	Power supply Plus	5 V ± 10 %
7	-	Not connected	
8	RxD/TxD-N	Receive/transmit data N (A/A')	RS485
9	-	Reference filtered	M_EXT

Table 8.2-9 Pin assignment of X448 connection

8.2.6.2 Connecting up the bus cable by means of the RS485 bus connecting system

Maximum cable lengths

With the PROFIBUS, data transfer according to RS485 is most frequently used. A twisted, shielded copper cable with one pair of wires is used.

Up to a maximum of 124 units can be connected to a PROFIBUS phase. In one bus segment, up to 32 units can be connected together in a linear structure. If there are more than 32 nodes, repeaters (power amplifiers) must be used in order to link up the individual bus segments.

The maximum cable length depends on the baud rate (transmission speed).

The maximum cable length can be increased by using repeaters but no more than three repeaters may be connected in series.

The maximum cable lengths given in the following table can only be ensured if PROFIBUS bus cables are used (e.g. Siemens PROFIBUS-cable with MRPD 6XV 1830-0AH10).

Baud rate	Max. cable length in a segment [m]	Max. distance between 2 stations [m]
9.6 to 187.5 kbaud	1000	10000
500 kbaud	400	4000
1.5 Mbaud	200	2000
3 to 12 Mbaud	100	1000

Table 8.2-10 Permissible cable length of a segment with RS485 repeaters

**Rules for laying
cables**

When you are laying the bus cable, you must not:

- ◆ twist it
- ◆ stretch it
- ◆ or squash it

In addition to this, you must take into account any influences on electromagnetic compatibility (EMC).

For further information, see for example Chapter 3 of the Compendium or the description "Instructions for Design of Drives in Conformance with EMC Regulations" (Order No. 6SE7087-6CX87-8CE0).

Bus connectors

You need bus connectors in order to connect the PROFIBUS to a CBP. There are different types of bus connector with degree of protection IP20. Their different uses are shown in the table below.

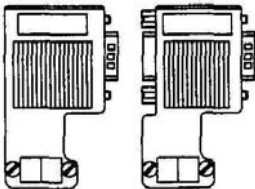
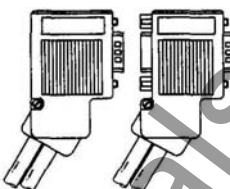
Order No.	6ES7 972-0BA11-0XA0 6ES7 972-0BB11-0XA0	6ES7 972-0BA40-0XA0 6ES7 972-0BB40-0XA0
Appearance		
PG socket	0BA11: no 0BB11: yes	0BA40: no 0BB40: yes
Max. baud rate	12 Mbaud	12 Mbaud
Terminating resistor	Can be connected as required	Can be connected as required
Outgoing cable	Vertical	slanting
Interfaces	• PROFIBUS nodes • PROFIBUS cable	• 9-pole Sub-D socket • 4 terminal blocks for wires up to 1.5 mm²
Connectable diameter of PROFIBUS cable	8 ± 0.5 mm	8 ± 0.5 mm
Recommended for	• IM 308-B • IM 308-C • S5-95U • S7-300 • S7-400 • M7-300 • M7-400 • CBP	• IM 308-B • IM 308-C • S5-95U • S7-300 • S7-400 • M7-300 • M7-400 • CBP

Table 8.2-11 Structure and application of bus connectors with IP20 protection

***) NOTE**

– These connectors must not be inserted in slots **E and G** on **Compact units**.

– If these connectors are used on **CompactPLUS** units contour faults may occur on slots **A and B**.

(See board description in Section "Connecting-up", Fig. "PROFIBUS connector".)

For more information on ordering and additional descriptions, see the A&D AS catalog "Industrial Communication" IK 10 (Order No. E86060-K6710-A101-A6).

Installing the bus cable

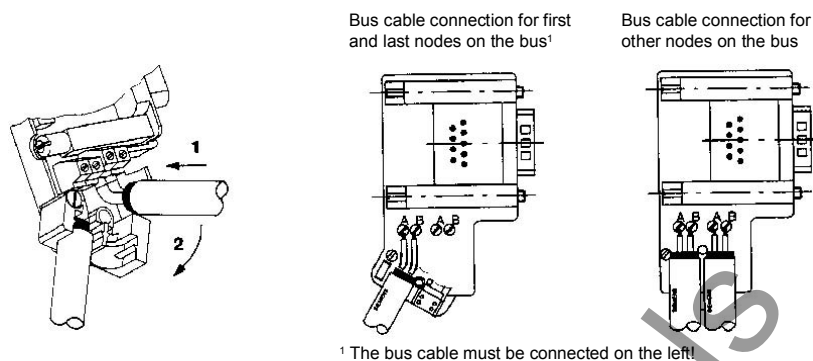


Fig. 8.2-11 Connecting up the bus cable to the bus connector

Bus termination

Each bus segment must be fitted with a resistor network, the bus termination, at each end.

If the recommended bus connectors can be used, the bus termination can be connected or disconnected by means of switches.

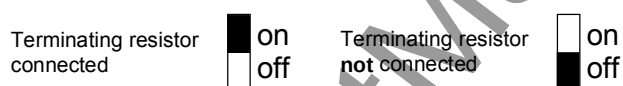


Fig. 8.2-12 Switch positions for connected or disconnected bus termination resistor

If these bus connectors are not used, the user must ensure installation of a bus termination network at the first and last bus station in accordance with the following illustration.

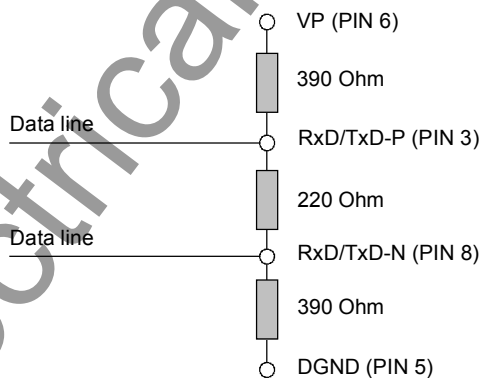


Fig. 8.2-13 Bus termination network

NOTICE

A bus segment must always be terminated at both ends with a matching resistor. This is not the case, for example, if the last slave with bus connector is not live. Because the bus connector obtains its voltage from the station, the matching resistor has no effect.

Make sure that the stations at which the matching resistor is connected is always supplied with voltage.

Pulling out the bus connector

You can pull out the bus connector with looped-through bus cable from the PROFIBUS-DP interface at any time without interrupting data transfer on the bus.

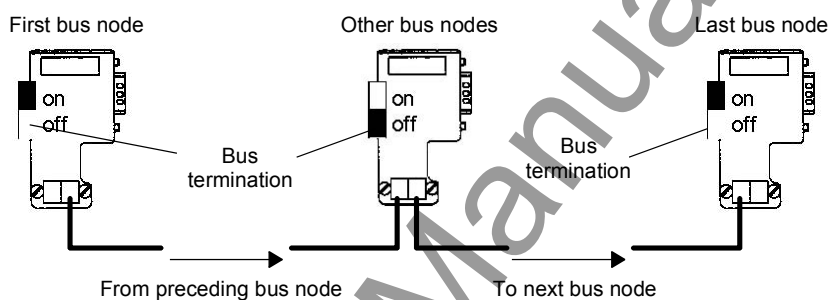
Connection example

Fig. 8.2-14 Bus segment in linear structure (max. 32 stations per segment)

8.2.6.3 Connecting the bus cable with the fiber-optic cable system

For applications in an environment which is subjected to a high level of interference, fiber-optic cables can also be used with the PROFIBUS-DP. The specification of fiber-optic-cable transmission is defined in PROFIBUS guideline No. 2.021.

For connecting fiber-optic cables to the CBP, an OLP (Optical Link Plug) can be used which provides integrated conversion of the RS485 signals in fiber-optic cables and vice versa.

Area of application

With the optical link plugs (OLPs), optical PROFIBUS networks in ring form can easily be created (single-fiber ring with plastic fiber-optic cables).

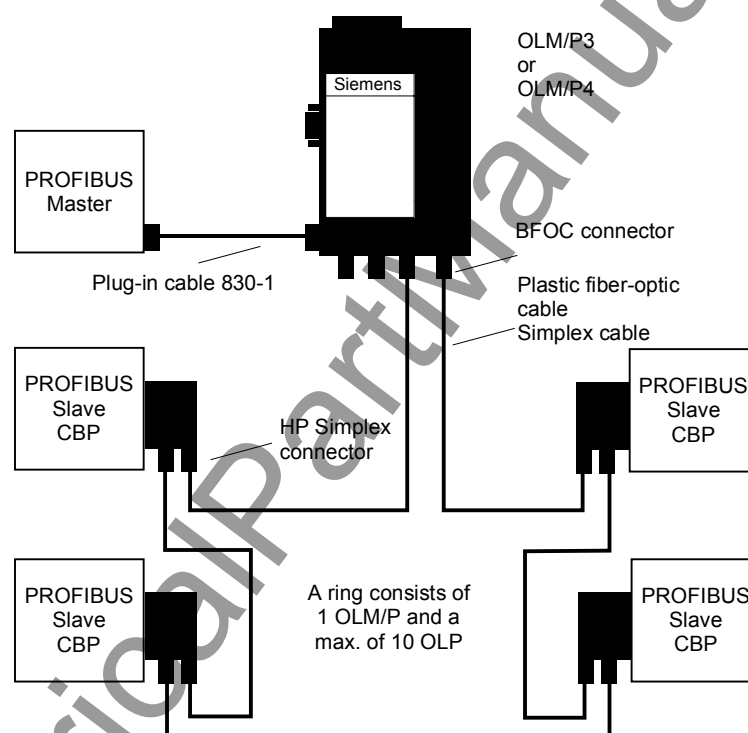


Fig. 8.2-15 Example of a system configuration with OLPs

The OLP can be directly plugged into the 9-pole SUB-D socket of the CBP. Power is supplied to the OLP via the 9-pole SUB-D connector of the CBP.

The transmission reliability of PROFIBUS networks is greatly increased by using fiber-optic cable instead of twisted two-wire cable. As a result, the bus is insensitive to interference from electromagnetic influences or overvoltages.

Considerable cost savings are achieved by using plastic fiber-optic cables which are also easy to fit. Additional grounding is no longer necessary either.

Functions

- ◆ Connection of a PROFIBUS slave to an optical single-fiber ring
- ◆ Cable length between 2 OLPs with plastic fiber-optic cable from 1 m to 25 m
- ◆ Maximum circumference of a single-fiber ring: 275 m
- ◆ Transmission rate of 93.75 kbit/s to 1.5 Mbit/s; can be adjusted by means of plug-in jumpers (this can be checked through inspection windows in the connector housing)
- ◆ OLP single-fiber rings can be integrated in PROFIBUS networks by means of OLM/Ps

Requirements for use

- ◆ One OLM/P per single-fiber ring is necessary as a coordinator.

Ordering data

OLP / OLM for PROFIBUS	Order No.
OLP Optical link plug for creating optical single-fiber rings with plastic fiber-optic cables; including 2 HP Simplex connectors and mounting instructions	6GK1 502-1AA00
OLM/P3 Optical link module for plastic fiber-optic cables, 3-channel version with signaling contact, including 2 BF OC connectors	6GK1 502-3AA10
OLM/P4 Optical link module for plastic fiber-optic cables, 4-channel version with signaling contact, including 4 BFOC-connectors	6GK1 502-4AA10

For more information on ordering and additional descriptions, see the A&D AS catalog "Industrial Communication" IK 10 (Order No. E86060-K6710-A101-A6).

8.2.6.4 Shielding of the bus cable / EMC measures

In order to ensure interference-free operation of the PROFIBUS-DP, especially in the case of data transmission with RS485, the following measures are imperative:

Shielding

- ◆ For the PROFIBUS bus cable, the shield in the bus connector should be connected to the CBP. Shielding is also provided by the shield clamps (in the event of Compact units) or by the shield clamps and cable ties (in the event of chassis-type units) on the converter housing. The following illustrations show you how to use the shield clamps. When removing the insulation from the various core ends, please ensure that the solid copper core is not damaged.
- ◆ Please ensure that the shield of each bus cable is connected to protective earth, both where it enters the cabinet as well as at the converter housing.

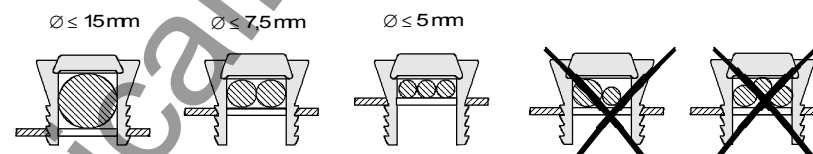
NOTE

Bus cables are to be laid at an angle of 90 ° to power cables if it is necessary that the two kinds of cable intersect.

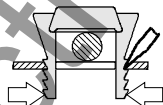
NOTE

The bus cables must be twisted and shielded and are to be laid separately from the power cables at a minimum distance of 20 cm. The braided shield and, if necessary, the underlying foil shield as well, are to be connected on both sides through a large surface area so that they are highly conductive, i.e. the shield of the bus cable between two converters is to be connected to the converter housing at both ends of the cable. The same applies to the shielding of the bus cable between the PROFIBUS-DP master and the converters.

Snap in the shield clamp



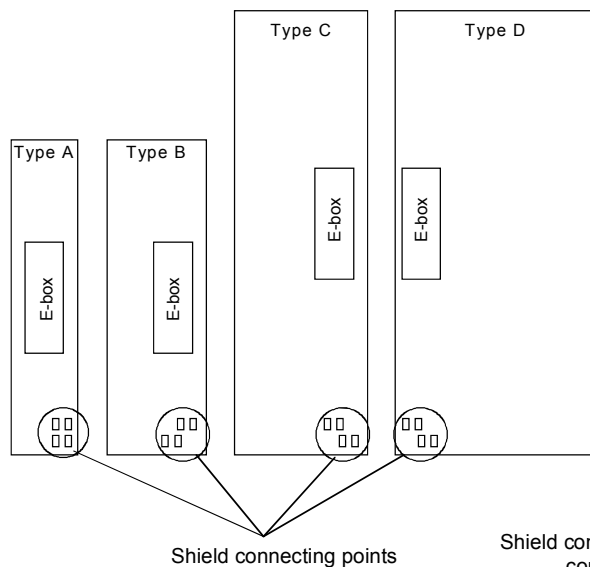
Release the shield clamp



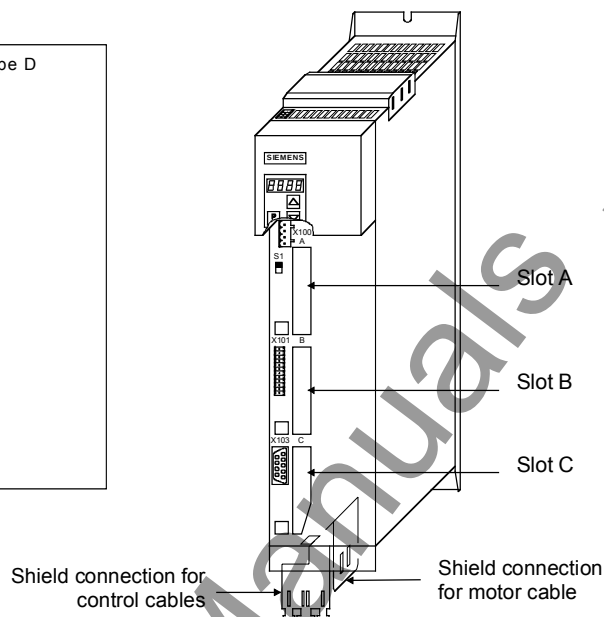
Squeeze the shield clamp together with your hand or a screwdriver and pull upwards.

Fig. 8.2-16 Using the shield clamps

Compact type and chassis type units



Compact PLUS MC:



Compact Plus VC:

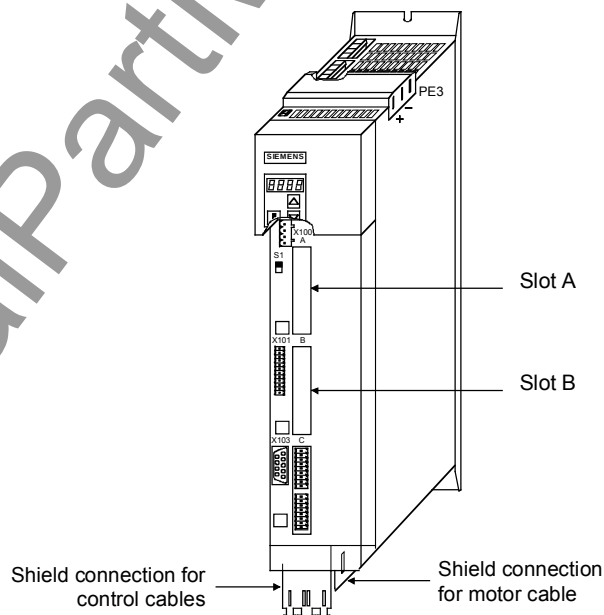


Fig. 8.2-17 Position of the shield connecting points

If so many control cables are used that two shield clamps are insufficient, the "EMC shielded housing" option is to be used.

Potential equalization

- ◆ Please avoid differences in potential (e.g. as a result of different power supply levels) between the converters and the PROFIBUS-DP master.
- ◆ Use equipotential bonding cables:
 - 16 mm² Cu equipotential bonding cables up to 200 m
 - 25 mm² Cu equipotential bonding cables over 200 m
- ◆ Route the equipotential bonding cables so that there is the smallest possible surface between the equipotential bonding cables and signal cables.
- ◆ Connect equipotential bonding cables to the ground/protective conductor through the largest possible surface area.

Laying cables

Instructions for laying cables:

- ◆ Bus cables (signal cables) must not be laid close to and parallel to power cables.
- ◆ Signal cables and the associated equipotential-bonding cables must be laid as closely together as possible and kept as short as possible.
- ◆ Power cables and signal cables must be laid in separate cable ducts.
- ◆ Shields must be connected through the largest possible surface area.

For more information on electromagnetically compatible installation of systems, see for example Chapter 3 of the Compendium or the description "Instructions for Design of Drives in Conformance with EMC Regulations" (Order No. 6SE7087-6CX87-8CE0).

8.2.7 Starting up the CBP

NOTE

With regard to basic parameterization, please note the differences to the types of unit with the older function classes FC (CU1), VC (CU2) and SC (CU3). These differences are described below.

In order to make these differences clear, these parameter numbers and other deviations are either printed in dark gray or have a dark-gray background.

8.2.7.1 Basic parameterization

NOTE

For the CBP optional board, it is not necessary to adjust the baud rate.

Basic parameterization with CUPM, CUMC, CUVC and Compact PLUS

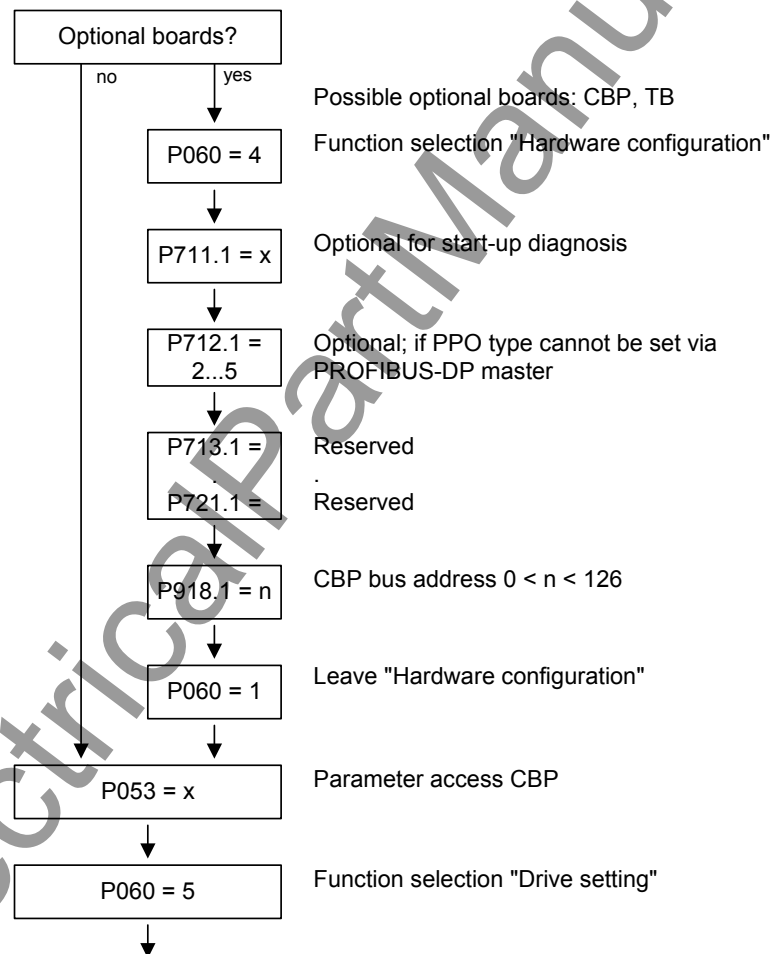


Fig. 8.2-18 Parameterization of "Hardware configuration" for CUPM, CUMC, CUVC and Compact Plus

With MASTERDRIVES Performance 2, the CB parameters P918 and P711 to P721 can also be changed in the "Drive setting" state (P60 = 5). In the case of MASTERDRIVES MC (CUMC) and MC+ (Compact+) from firmware version V1.4 onwards, the CB parameters P918 and P711 to P721 can also be changed in the "Drive setting" status (P060 = 5).

Basic parameterization for FC (CU1), VC (CU2) and SC (CU3)

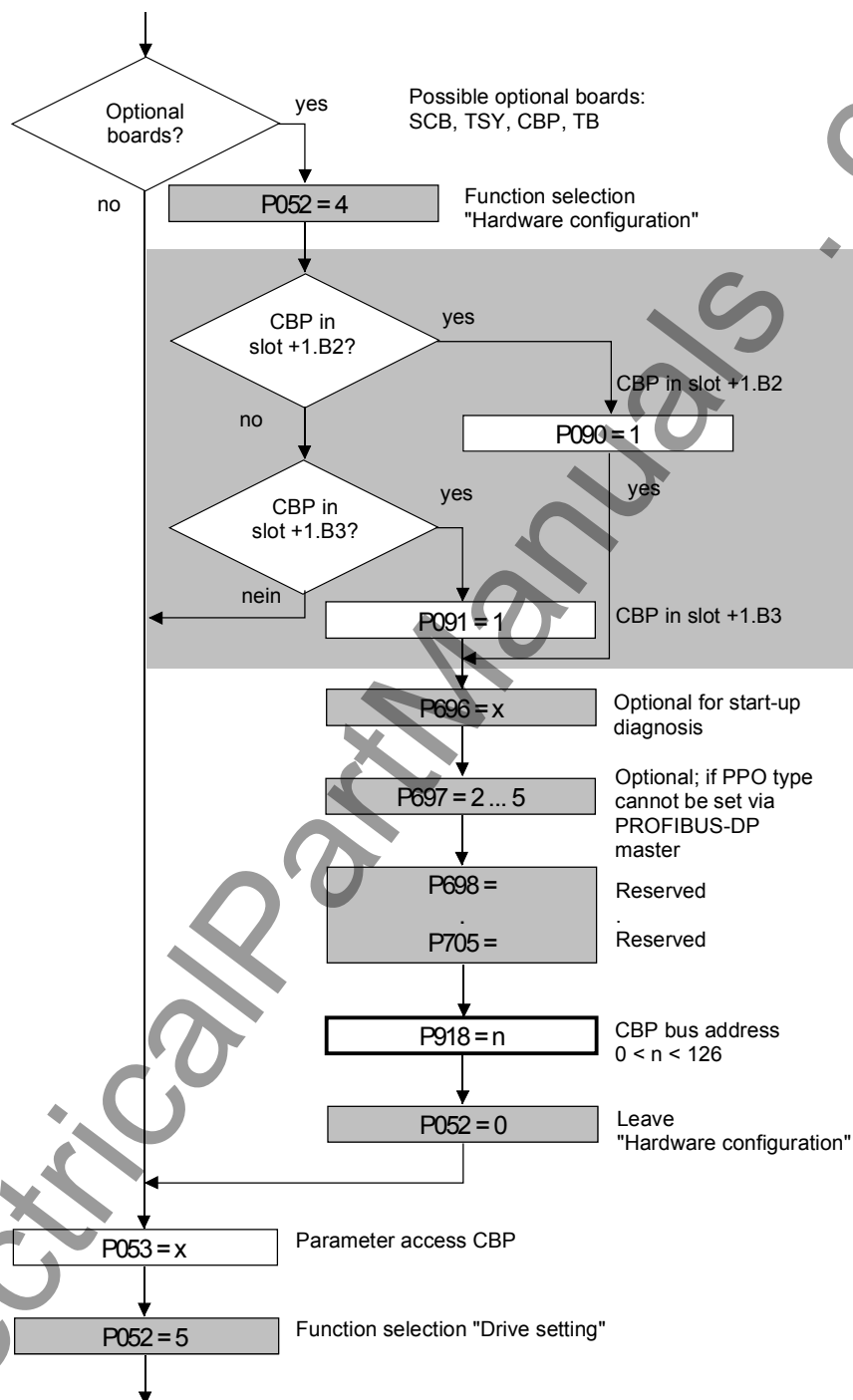


Fig. 8.2-19 Parameterization of "Hardware configuration" for FC (CU1), VC (CU2) and SC (CU3)

NOTE

All grayed out parameters are only valid for units with the functions FC (CU1), VC (CU2) and SC (CU3).

NOTE

The following conventions apply to all parameters with index (e.g. P918.x) given below:

- ◆ Index 1 is valid for the first CBP
- ◆ Index 2 is valid for the second CBP

To determine which CBP is the first and which the second, see Section 8.2.5 "Mounting methods / CBP slots".

P053 (parameter access)

This parameter is significant for the CBP if you wish to set or change parameters of the converter (including technology) by means of the PKW part of the PROFIBUS telegrams.

In this case, please set parameter P053 to an uneven number (e.g. 1, 3, 7 etc.). With parameter P053, you can define the positions (PMU, CBP etc.) from which parameters may be altered.

Example: P053 = 1: Parameter access only CBP
 = 3: Parameter access CBP+PMU
 = 7: Parameter access CBP+PMU+SCom1 (OP)

If changing parameters (= parameter access) has been enabled via the CBP (P053 = 1, 3 etc.), all other parameter settings can be made from the PROFIBUS-DP master via the bus.

For the additional setting of parameters which concern data transfer via the PROFIBUS-DP (e.g. process data (PZD) combination), you must know the PPO type used for the transfer of useful data.

P060**P052**

Function selection "Hardware setting"

P090 (board position 2) or P091 (board position 3)

You can also change these parameters if the CBP exchanges useful data via the PROFIBUS-DP. In this way, you can isolate the PROFIBUS-DP interface from the converter with the appropriate parameterization. In this case, the CBP changes over to the PROFIBUS-DP status "Static Diagnosis", i.e. the CBP causes the PROFIBUS-DP master to exit the data-exchange mode and only to request diagnostic telegrams from the CBP.

P918.x (CBP Bus Address)**P918 (CBP Bus Address)**

The bus address set in parameter P918 is not accepted until P060 has been changed from "5" to "7". If the address is changed after the CBP has been parameterized, fault F080 will be output.

An address change only becomes effective after the power supply to the electronics box has been turned off and then turned on again!

P711.x (CBP Parameter 1)	P696 (CBP Parameter 1)
With this parameter, you can activate special diagnostic information for start-up and service. During normal operation, P711 / P696 has the value 0 (default setting).	

P712.x (CBP Parameter 2)	P697 (CBP Parameter 2)
If you use a PROFIBUS-DP master system that enables you to select a PPO type (e.g. SIMATIC S7), you do not need to do anything with P712 / P697 (simply bypass this parameter P712 / P697)! If you are using a PROFIBUS-DP master system where it is not possible to specify the PPO type at the converter by means of the identification byte (e.g. CP5431 for SIMATIC S5), you can specify a PPO type with parameter P712 / P697. With the default setting (P712 / P697 = 0), the CBP automatically sets the type of PPO. P712 / P697 = 0: PPO1 (default setting) = 1: PPO1 = 2: PPO2 = 3: PPO3 = 4: PPO4 = 5: PPO5	

P713.x (CB Parameter 3)	P698 (CBP Parameter 3)
Only CBP2 Communications protocol: P713 / P698 = 0: PROFIBUS (Default setting) (P713 / P698 = 1: reserved) P713 / P698 = 2: USS Only selected parameters are relevant (see below). A change from the PROFIBUS to the USS protocol and vice versa does not come into effect until after the voltage has been switched off and then on again.	

P714.x (CB Parameter 4)	P699 (CBP Parameter 4)
Only CBP2 Write requests of a SIMATIC OP are stored permanently (EEPROM) or temporarily (RAM). P714 / P699 = 0: EEPROM (default setting) P714 / P699 = 1: RAM	

P715.x (CB Parameter 5)	P700 (CBP Parameter 5)
Only CBP2 Failure of a cross-traffic relationship is signaled as a fault or alarm. P715 / P700 = 0: Fault (default setting) In the event of failure, transmission of all setpoints to the basic unit is stopped. This leads to fault F082 P715 / P700 = 1: Alarm The failure is only signaled by alarm A088. With regard to the missing setpoints, those last received are retained.	

NOTE

After the above settings have been made, the CBP is logged-on in the converter and is ready to establish connections to the PROFIBUS-DP. It is not yet possible to process the process data via the PROFIBUS-DP after this has been done.

This additionally requires the type of process data interconnection described in the following section 8.2.7.2.

USS

For USS-relevant parameter numbers, only CBP2 with P713.x = 2:

CBP2 parameter number	Meaning	Corresponds to Scom/SCB parameter number
P918.x	Bus address	P700
P718.x (CB parameter 8)	Baud rate 6 = 9.6 kbaud 7 = 19.2 kbaud 8 = 38.4 kbaud	P701
P719.x (CB parameter 9)	Number of PKWs	P702
P720.x (CB parameter 10)	Number of PZDs	P703
P722.x	Telegram failure time	P704

Further information on the USS protocol can be found in Section 8.1, USS.

8.2.7.2 Process data interconnection in the units

Definition

Process data interconnection involves the linking up of setpoints and control bits to the RAM interface. The transferred process data only become effective when the used bits of the control words as well as the setpoints, status words and actual values are allocated (connected) to the dual-port RAM interface.

The CBP stores the received process data at fixed pre-determined addresses in the dual-port RAM. Each item of process data (PZDi, $i = 1$ to 10) is assigned a connector (e.g. 3001 for PZD1). The connector is also used to determine whether the PZDi ($i = 1$ to 10) is a 16-bit value or a 32-bit value.

With the help of selector switches (e.g. P554.1 = selector switch for bit 0 of control word 1), the setpoints or the individual bits of the control words can be assigned to a particular PZDi in the dual-port RAM. In order to do this, the connector belonging to the required PZDi is assigned to the selector switch.

NOTE

In function classes CUPM, CUMC, CUVC and Compact PLUS, the control words STW1 and STW2 are also available in bit form on so-called binectors (explanations of BICO systems can be found in Chapter 4 "Function Blocks and Parameters").

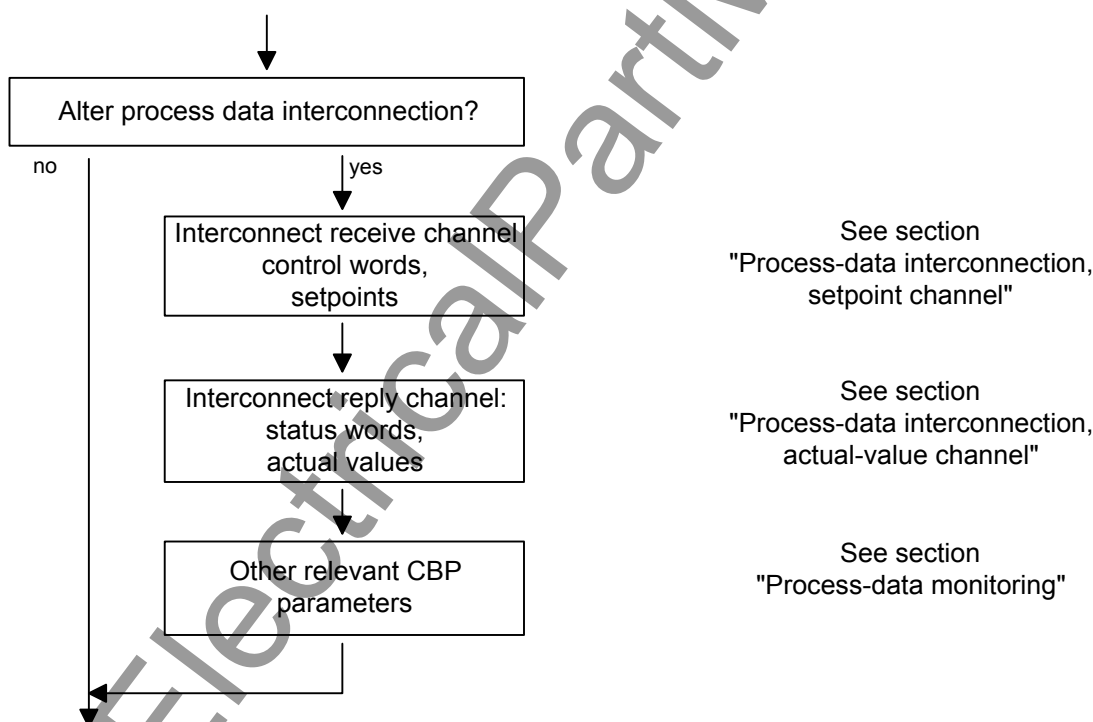


Fig. 8.2-20 Procedure for changing process data

NOTICE

Rewiring from 16 to 32 bit and vice versa should not be done while the equipment is in operation, because the changeover takes several milliseconds, during which time the data on the bus are not consistent (high and low can change places).

Examples

The following pages contain examples of how the transferred data are allocated in the units by means of process data interconnection.

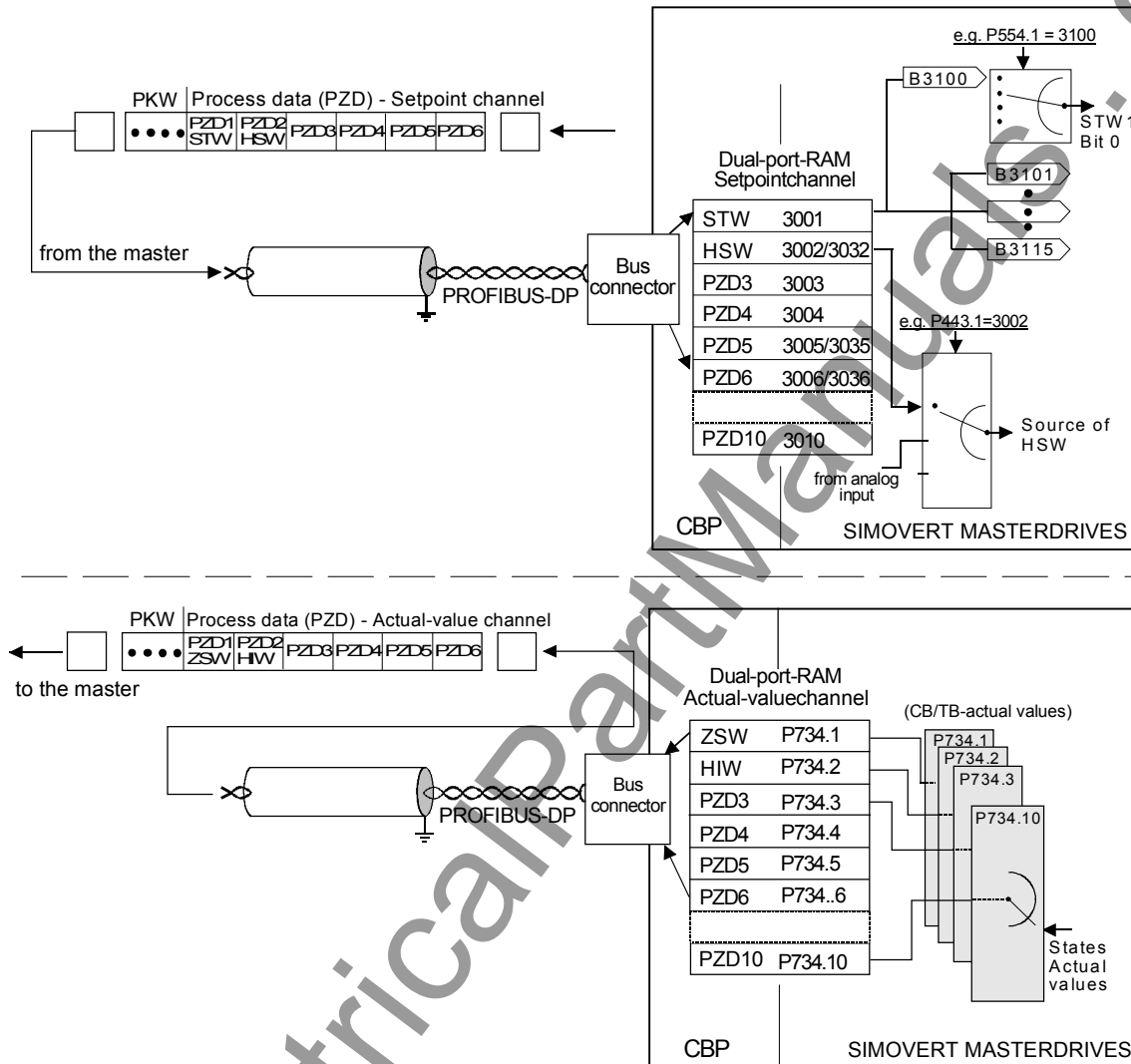


Fig. 8.2-21

Example of process data interconnection of the first CB board in function classes Motion Control Compact PLUS, CUPM, CUMC and CUVc

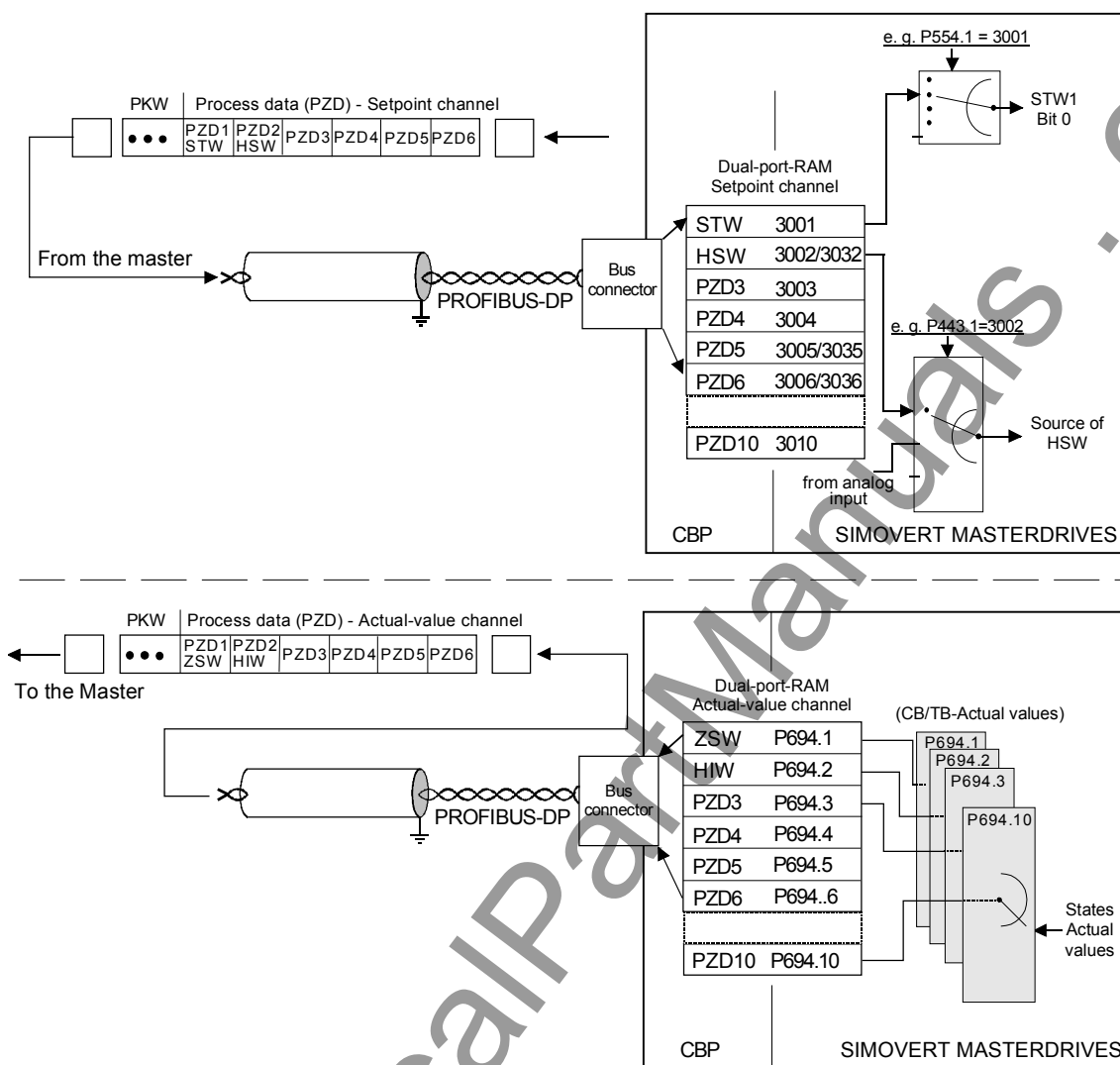


Fig. 8.2-22 Example of process data interconnection for function classes FC (CU1), VC (CU2) and SC (CU3)

Process data interconnection, setpoint channel
Master → Converter

- ◆ The "tens digit" of the connector is used to distinguish between a 16-bit process data item (e.g. 3002) and a 32-bit process data item (e.g. 3032).
- ◆ If a process data item is transferred as a 16-bit quantity, assign the required PZDi-relevant connector for a 16-bit process data item to the selector switch (see "Process data linkage"). (Example: If a 16-bit process data item is assigned to PZD2, the relevant connector is 3002).
- ◆ If a process data item is transferred as a 32-bit process data item, assign the required PZDi-relevant connector for a 32-bit process data item to the selector switch (see "Process data interconnection"). For this, use the connector of the least-significant PZDi (Example: If a 32-bit process data item is assigned to PZD2 + PZD3, the relevant connector is 3032).
- ◆ The first word (associated connector : 3001 or the binectors 3100 to 3115) of the received process data is always assigned to control word 1 (STW1).
- ◆ The second word is always assigned to the main setpoint (HSW).
- ◆ If the main setpoint is transferred as a 32-bit process data item, it is also assigned to word 3. In this case, the most-significant part of the main setpoint is transferred in word 2 and the least-significant part is transferred in word 3.
- ◆ If a control word 2 (STW2) is transferred, the fourth word (relevant connector = 3004 or binectors 3400 to 3415) is always assigned to STW2.

NOTE

In PPO types 1 and 3, the PZD part only consists of two words. Here, only control word 1 and the main setpoint (as 16-bit value) can be linked up to the dual-port RAM interface.

- ◆ The connector for the setpoint channel is always a 4-digit one. The connectors assigned to the process data (PZD1 to PZD10) are shown in the function diagram of the relevant CU board. The connectors are entered at the PMU as 4-digits values (e.g. 3001). When parameterization is done via the PROFIBUS-DP, the connector is entered via the bus and also via the PMU (e.g. connector 3001 is transferred as 3001_{hex}).

NOTE

Process data interconnection of the setpoint channel can also be carried out via the PROFIBUS-DP if P053 has previously been set to an uneven value.

Please bear in mind that control word 1 (STW1) has the value 0 during the parameterization phase (process data interconnection)!

**Interlocking of
connectors and
double connectors**

MC V1.50 and higher / CUVC V3.23 and higher

DANGER

When interconnecting connectors, binectors, and double word connectors, please note that simultaneous interconnection of a connector, and a double word connector with the same name is not permitted, because when a double word connector (e. g. KK3032) is connected, the meanings of the connectors K3002 and K3003 are swapped round (high-word and low-word exchanged).

On MASTERDRIVES MC and Compact Plus on software version V1.50 and higher and on MASTERDRIVES CUVC on software version V3.23 and higher, simultaneous use of connectors and double word connectors with the same name is mutually interlocked (see also function diagrams [121] and [131]).

Because the binectors are not included in the interlocking (to ensure compatibility for older configurations), their significance changes according to whether the pertinent word or double word is wired.

Example for the setpoint channel

PZD interconnection for the bits of control word 1 (STW1), the main setpoint (HSW) and the bits of control word 2 (STW2).

At the converter via PMU		Meaning
P554.1 = <u>3100</u>	P554.1 = <u>3001</u>	Control word 1 bit 0 (Src ON/OFF1) via DPR interface (word 1)
P555.1 = <u>3101</u>	P555.1 = <u>3001</u>	Control word 1 bit 1 (SrcON/OFF2) via DPR interface (word 1)
P443.1 = <u>3002</u>	P443.1 = <u>3002</u>	16-bit main setpoint (Src Main Setpoint) via DPR interface (word 2)
P588.1 = <u>3411</u>	P588.1 = <u>3004</u>	Control word 2 bit 28 (Src No Ext Warn1) via DPR interface (word 4)

If the factory setting of the converter is used, the above example of parameterization is a functioning way of interconnecting process data (setpoints).

- *Italics:*
Parameter number (if the PMU is a decimal number, via PROFIBUS-DP as an equivalent HEX number).
- Single underline:
Index (if the PMU is a decimal number, via PROFIBUS-DP as an equivalent HEX number).
- Double underline:
Interconnection value: defines whether the parameter selected by the *parameter number* is transferred as a 16-bit value or as a 32-bit value and at which position in the PZD-setpoint telegram (PZDi), the parameter is transferred.
 - White background = MASTERDRIVES Compact PLUS, CUPM, CUMC or CUVC (first CBP)
 - Grey background = MASTERDRIVES FC (CU1), VC (CU 2) or SC (CU 3)

Process data interconnection, actual-value channel

The actual-value process data (PZDi, i = 1 to 10) are assigned to the appropriate status words and actual values by the indexed parameter P734.i / P694.i (CB/TB actual values). Each index stands for a process data item (e.g. 5 → PZD5 etc.). Please enter the number of the connector or parameter whose value and corresponding process data item you wish to transfer in parameter P734 / P694 under the relevant index.

The status word is always to be transferred in the PZD1 word of the PZD reply (actual-value channel), and the main actual value in PZD2. What additional items are assigned to the PZD (PZD1 up to, if necessary, PZD10) is not specified. If the main actual value is transferred as a 32-bit value, then it is assigned to PZD2 and PZD3.

Example for the actual-value channel PZD interconnection for status word 1 (ZSW1), the main actual value (HIW) and status word 2 (ZSW2).

At the converter via PMU		Meaning
P734.1 = <u>32</u>	P694.1 = <u>968</u>	Status word 1 (K032 / P968) is transferred in the actual-value channel by means of PZD1.
P734.2 = <u>151</u>	P694.2 = <u>218</u>	The actual speed n/f (KK151 / P218) is transferred in the actual-value channel by means of PZD2 (here, as a 16-bit quantity; PZD3 is empty).
P734.4 = <u>33</u>	P694.4 = <u>553</u>	Status word 2 (K033 / P553) is transferred in the actual-value channel by means of PZD4.
Example: 32-bit main actual value		
P734.2 = <u>151</u>	P694.2 = <u>218</u>	The actual speed n/f (KK151 / P218) is transferred in the actual-value channel by means of PZD2 ...
P734.3 = <u>151</u>	P694.3 = <u>218</u>	... and as a 32-bit value by means of PZD3.

- **Italics:**
P734 / P694 (CB/TB actual value), if PMU is shown as a decimal number, transferred via PROFIBUS-DP as an equivalent HEX (2B6 Hex).
- **Single underline:**
Index (if PMU is a decimal number, via PROFIBUS-DP as an equivalent HEX number): Specifies at which position in the PZD actual-value telegram the actual value selected by the parameter number is transferred.
- **Double underline:**
Parameter number of the required actual value.
 - White background = MASTERDRIVES Compact PLUS, CUPM, CUMC or CUVC (first CBP)
 - Grey background = MASTERDRIVES FC (CU1), VC (CU 2) or SC (CU 3)

NOTE

If actual values are transferred as a 32-bit data item, you must enter the appropriate connector number at two consecutive words (indices).

8.2.7.3 Process data interconnection via standard telegrams

Definition PROFIdrive profile version V3 defines standard telegrams for cyclical data exchange.

Telegram selection Process data can be interconnected for standard telegrams by means of a Script file.

Structure of standard telegrams See also PROFIdrive version 3 (PNO: Order No. 3172).

Standard telegram 1:

PZD number	1	2
Setpoint	STW1	NSOLL_A

PZD number	1	2
Actual value	ZSW1	NIST_A

Standard telegram 2:

PZD number	1	2	3	4
Setpoint	STW1	NSOLL_B		STW2

PZD number	1	2	3	4
Actual value	ZSW1	NIST_B		ZSW2

Standard telegram 3:

PZD number	1	2	3	4	5
Setpoint	STW1	NSOLL_B		STW2	G1_STW

PZD number	1	2	3	4	5	6	7	8	9
Actual value	ZSW1	NIST_B		ZSW2	G1_ZSW	G1_XIST1		G1_XIST2	

Standard telegram 4:

PZD number	1	2	3	4	5	6
Setpoint	STW1	NSOLL_B		STW2	G1_STW	G2_STW

PZD number	1	2	3	4	5	6	7	8	9	...
Actual value	ZSW1	NIST_B		ZSW2	G1_ZSW	G1_XIST1		G1_XIST2		...

...			...	10	11	12	13	14
...			...	G2_ZSW	G2_XIST1		G2_XIST2	

Standard telegrams 5 and 6 are derived from standard telegrams 3 and 4 for the Dynamic Servo Control (DSC) function.

Standard telegram 5:

PZD number	1	2	3	4	5	6	7	8	9
Setpoint	STW1	NSOLL_B		STW2	G1_STW	XERR		KPC	
PZD number	1	2	3	4	5	6	7	8	9
Actual value	ZSW1	NIST_B		ZSW2	G1_ZSW	G1_XIST1		G1_XIST2	

Standard telegram 6:

PZD number	1	2	3	4	5	6	7	8	9	10
Setpoint	STW1	NSOLL_B		STW2	G1_STW	G2_STW	XERR		KPC	
PZD number	1	2	3	4	5	6	7	8	9	...
Actual value	ZSW1	NIST_B		ZSW2	G1_ZSW	G1_XIST1		G1_XIST2		...
...			...			10	11	12	13	14
...			...			G2_ZSW	G2_XIST1		G2_XIST2	

Signals:

Signal No.	Meaning	Abbreviation	Length 16/32-bit	Sign
1	Control word 1	STW1	16	
2	Status word 1	ZSW1	16	
3	Control word 2	STW2	16	
4	Status word 2	ZSW2	16	
5	Speed setpoint A	NSOLL_A	16	with
6	Actual speed A	NIST_A	16	with
7	Speed setpoint B	NSOLL_B	32	with
8	Actual speed B	NIST_B	32	with
9	Encoder 1 control word	G1_STW	16	
10	Encoder 1 status word	G1_ZSW	16	
11	Encoder 1 actual position 1	G1_XIST1	32	
12	Encoder 1 actual position 2	G1_XIST2	32	
13	Encoder 2 control word	G2_STW	16	
14	Encoder 2 status word	G2_ZSW	16	
15	Encoder 2 actual position 1	G2_XIST1	32	
16	Encoder 2 actual position 2	G2_XIST2	32	
25	Control deviation	XERR	32	with
26	Position controller gain factor	KPC	32	with

8.2.7.4 Process data monitoring

NOTE

Please note the different parameter numbers for the types of unit with the older function classes FC (CU1), VC (CU2) and SC (CU3).

In order to make these differences clear, these parameter numbers are either printed in dark gray or have a dark-gray background.

P722.x (CB/TB TIgOFF)	P695 (CB/TB TIgOFF)
With parameter P722. / P695, you can determine whether entering of process data into the dual-port RAM by the CBP is to be monitored by the converter. For parameter P722 ◆ Index 1 is applicable for the first CBP and ◆ Index 2 is applicable for the second CBP. To determine which CBP is the first one and which is the second one, see section 8.2.5 "Mounting methods / CBP slots".	

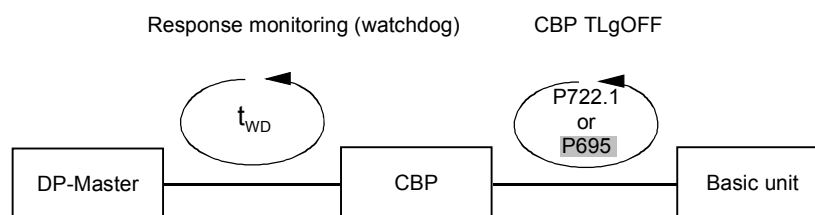
If process data monitoring has been activated, a fault in the DP master is followed by a reaction of the converter, irrespective of the reply-monitoring time in the CBP.

&	P722.x ≠ 0	P722.x = 0	P695 ≠ 0	P695 = 0
Response monitor active	Reaction Yes	Reaction No	Reaction Yes	Reaction No
Response monitor inactive	Reaction No	Reaction No	Reaction No	Reaction No

Table 8.2-12 Process data monitoring depending on P722.1/P695 and the response monitor t_{WD}

When the DP master is being configured, it is specified whether telegram traffic with the master is to be monitored by the slave (CBP). If response-monitoring is active, the PROFIBUS-DP master passes on a time value t_{WD} (watchdog time) to the Slave when a connection is made.

If the response-monitoring time expires, the CBP ceases to write process data into the dual-port RAM. When this is combined with P722.x / P695, it is therefore possible to plan your process data monitoring.

Fig. 8.2-23 Effect of t_{WD} and P722.1 / P695

Response-monitoring time t_{WD}					
P722.x P695	Yes		No		
	CPU (AG) in STOP	IM308B/C in STOP or Simatic "Supply off"	CPU (AG) in STOP	IM 308B/C in STOP	Simatic "Supply off"
0 ms	Converter continues to run with the useful data last received. Alarm A083	Converter continues to run with the useful data last received. Alarm A083/A084	Converter continues to run with the useful data last received.	Converter continues to run with the useful data last received. Alarm A083	Converter continues to run with the useful data last received.
10 ms	Fault trip with F082 after: Watchdog time + 10 ms	Fault trip with F082 after: Watchdog time + 10 ms	Converter continues to run with the useful data last received. Fault trip with F082 after restart of CPU.	Fault trip with F082 after: 10 ms	Converter continues to run with the useful data last received.

Table 8.2-13 Interaction of P722 / P695 and response monitoring (watchdog)

Always set parameter P722 / P695 to the value of 10 for operation with the CBP. Monitoring of process data is thus activated/deactivated by the value of the response monitoring time solely by the PROFIBUS-DP master! The converter monitors the entering of process data in the dual-port RAM from the instant at which the CBP enters valid process data in the dual-port RAM for the first time. Fault F082 can only be tripped after this instant!

Process data whose complete control word (PZD1) has the value zero is not passed on to the dual-port RAM by the CBP (alarm A083)! From MASTERDRIVES MC V1.62 with CBP2 V>=2.21 and standard telegram 5 (PROFIdrive Profile V3 with equidistance) onwards, the process data can be transferred to the dual-port RAM irrespective of the content of the control word.

A fault is followed by a fault trip after

- ◆ Watchdog time + 10 ms
- ◆ The 10 ms correspond to the value 10 of parameter P722 / P695 and can be neglected with respect to the response-monitoring value.
- ◆ For additional operation with a Class II master, please bear in mind the information in the section "Diagnosis with the Class II master" of Chapter 8.2.10.5.

DANGER



If the "ON" command (bit 0) is interconnected with the dual-port RAM interface, the following must be done for safety reasons:

An "OFF2" or "OFF3" command must be additionally parameterized to the terminal strip/PMU as otherwise the converter cannot be powered down by means of a defined command when there is a communications breakdown!

8.2.8 Settings for the PROFIBUS-DP master (Class 1)

PROFIBUS units have different performance characteristics. In order to ensure that all master systems can correctly communicate with the CBP in all the ways possible, the characteristic features of the CBP are summarized in the form of an electronic data sheet (data file). These so-called master files describe the characteristic features of a type of unit clearly and completely in an exactly specified format.

For the different master systems, the characteristics are summarized in a standardized master file (GSD) and, for the SIMATIC, in a type-description file specific to the SIMATIC.

Master file (GSD)

The CBP2 from V2.21 onwards supports PROFIdrive version 3. The device master file (GSD) is stored as an ASCII file (SI028045.GSD) on the floppy disk supplied with the CBP.

The GSD allows you to configure standard telegrams 1 to 6. It has been generated according to revision 4 for PROFIBUS DP-V2.

To ensure complete compatibility between CBP and CBP2 V2.10, PPO types can still be used for configuring purposes, as described below.

The CBP2 from V2.21 onwards can also be operated on the device master file for the CBP and CBP2 V2.1 (SIEM8045.GSD).

Type-description file

The type-description file is also available as an ASCII file (SI8045AX.200 and SI8045TD.200) on the floppy disk which accompanies the CBP.

Selecting the type of PPO

So-called identification bytes are transferred in the configuration telegram of the PROFIBUS-DP master. These bytes determine the type of PPO for the useful-data telegram.

These bytes can be assigned different values for selecting a particular type of PPO (except for PPO type 1). For PPO type 4, for example, either identification byte 0 = 245 and identification byte 1 = 0 can be entered or only identification byte 0 = 245. If an unknown combination of identification bytes is received, the CBP sets the bit "parameterization error" in the diagnostic telegram to the PROFIBUS-DP master.

PPO type	Identification byte 0			Identification byte 1			Identification byte 2			Identification byte 3			COMET200 Version
	Dec	Hex	COM	Dec	Hex	COM	Dec	Hex	COM	Dec	Hex	COM	Version
1	243	F3	4AX	241	F1	2AX							V4.x/V5.x
2	243	F3	4AX	243	F3	4AX	241	F1	2AX	0	0	0	V4.x/V5.x
2	243	F3	4AX	243	F3	4AX	241	F1	2AX				V4.x/V5.x
2	243	F3	4AX	245	F5	6AX							V5.x
3	241	F1	2AX	0	0	0							V4.x/V5.x
3	0	0	0	241	F1	2AX							V4.x/V5.x
3	241	F1	2AX										V4.x/V5.x
4	0	0	0	243	F3	4AX	241	F1	2AX	0	0	0	V4.x/V5.x
4	0	0	0	243	F3	4AX	241	F1	2AX				V4.x/V5.x
4	0	0	0	243	F5	6AX							V5.x
4	245	F5	6AX	0	0	0							V5.x
4	245	F5	6AX										V5.x
5	243	F3	4AX	243	F3	4AX	243	F3	4AX	241	F1	2AX	V4.x/V5.x
5	243	F3	4AX	243	F3	4AX	241	F1	2AX	243	F3	4AX	V4.x/V5.x
5	243	F3	4AX	249	F9	10AX							V5.x

Table 8.2-14 Values for the identification bytes

8.2.8.1 Operating the CBP with a SIMATIC S5

When the CBP is used with a **SIMATIC S5**, it is operated as a **standard DP slave**.

As possible master boards, the IM308 B or the IM308 C can be used, or even the CP5431 in limited form.

The planning tools COM ET200 or COM PROFIBUS are available for configuring the master station.

If older versions of these planning tools are used, you must copy the master file or type-description file from the accompanying floppy disk into the appropriate sub-directory of the planning software.

COM ET200 up to Version V4.x

When configuring the CBP, please use the SI8045TD.200 type-description file on the floppy disk.

Please copy the type-description file into the directory containing the COM ET 200 files in the PG/PC.

Example

CD C:\COMET200

COPY A:\SI8045TD.200 C:

The type of PPO is selected in the configuration mask of COM ET200 up to Version V4.x by entering identification bytes in accordance with the above table of identification bytes.

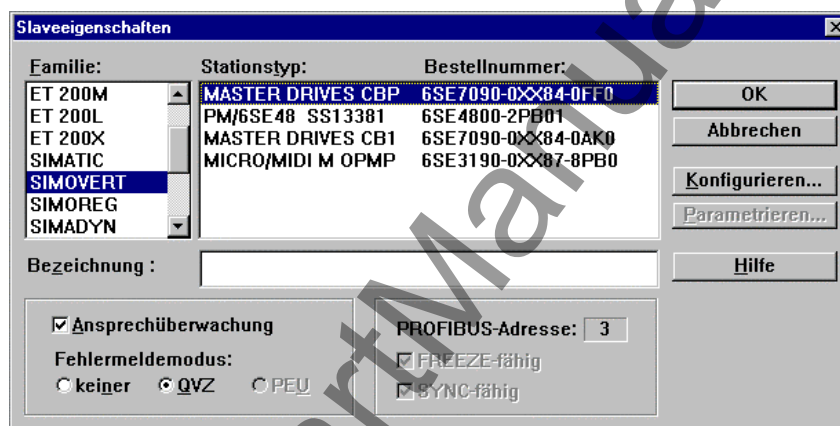
COM ET200 WIN and COM PROFIBUS

When configuring the CBP, please use the SI8045AX.200 type-description file on the floppy disk only if the CBP has not yet been included in the supplied version of the COM package.

Then copy the type-description file into the "TYPDAT5X" directory of the COM installation in the PG/PC.

From COM PROFIBUS V3.2 onwards, the CBP is included as standard and the type-description files on the floppy disk are then of no significance.

When a CBP is being configured (pull out the selector buttons "DRIVES" on the bus cable) and the suggested slave address is confirmed, a selection mask called "Slave characteristics" appears on the screen. It has the following appearance:



The required type of PPO is selected with this planning tool from a selection table called "Required configuration". This table appears automatically when the menu item "Configure" is selected.

More information on how to configure data exchange between a CBP and a SIMATIC S5 can be found in the description accompanying the DVA_S5 module package.

Using the DVA_S5 module package

The DVA_S5 module package (variable-speed drives with the SIMATIC S5) implements data transfer between SIMATIC and SIMOVERT slaves in accordance with the PROFIBUS profile for variable-speed drives and thus facilitates creation of the user program. A data module with the same appearance is always provided as the data interface, irrespective of which S5-CPU the program runs on. The programmer, therefore, does not need any detailed knowledge of the SIMATIC S5 system architecture or of the system functions which may be required.

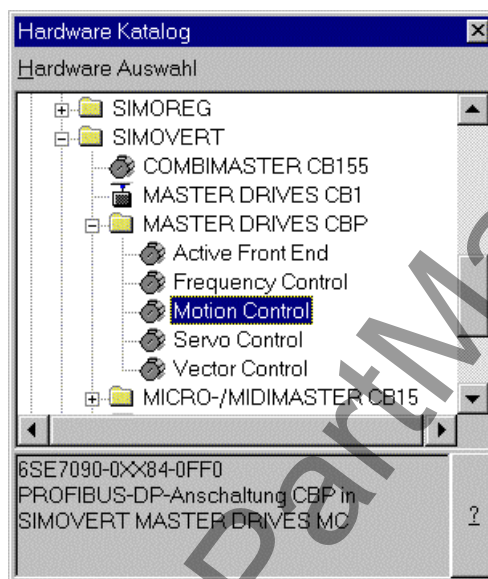
The DVA_S5 module package can be obtained from A&D WKF Fürth/Germany under MLFB 6DD1800-0SW0.

8.2.8.2 Operating the CBP with a SIMATIC S7

CBP as S7 slave	The CBP can be operated in two ways with a SIMATIC S7 :
Integrated PROFIBUS interfaces	◆ As a standard DP slave ◆ As a standard DP slave with extended functionality for SIMATIC S7 The CPUs with integrated PROFIBUS interface such as CPU315-2DP, CPU413-2DP, CPU414-2DP or CPU416-2DP etc. can be used as the possible S7 master.
CBP as a standard DP slave	The master station as well as the whole PROFIBUS network is configured in the STEP 7 hardware manager. Requirement: STEP 7 from V3.0 upwards If your STEP 7 hardware catalog does not yet contain the entry "MASTERDRIVES CBP", proceed as follows: Copy the type-description file S18045AX.200 from the supplied floppy disk into the STEP 7 index STEP7 à S7DATA à GSD. From STEP 7 version V4.01, the CBP is contained as standard in the hardware catalog, i.e. from version V4.01 onwards, the floppy disks are of no significance. In the "Extras" menu of the SIMATIC hardware configuration, then select the menu item "Update GSD files" and carry out this command. You will find the CBP in the "Hardware catalog" menu under "PROFIBUS-DP à Further field devices à Simover". It appears there under the name "MASTERDRIVES CBP".
CBP as a standard DP slave with extended functionality	To enable the CBP to be connected as a standard DP slave with extended functionality for SIMATIC S7 (e. g. acyclical communication with DriveMonitor) to the PROFIBUS-DP, a so-called DVA_S7 object manager has to be installed as an add-on to STEP 7. The DVA_S7 object manager is part of the DVA_S7 module package. STEP7 basis software, Version V3.1 and upwards, is a requirement for installation of the DVA_S7-OM. The DVA_S7-OM takes on the function of a master file or type-description file and supplements the unit characteristics stored there with all the necessary S7 characteristics.
S7 diagnosis	If the CBP is configured in SIMATIC S7 using the DVA_S7 object manager, a diagnosis alarm is automatically generated for the converter fault in the S7-CPU. This diagnosis alarm is derived from bit 3 of the status word (collective fault) and results in a STOP of the S7-CPU if the OB82 (diagnostics organization block) is not programmed. For the correct processing of the diagnosis alarm, the status word of the converter always has to be transferred unchanged as the first word from the converter to the CBP (see section "Process data interconnection").
NOTE	When a converter fault occurs, the CBP2 does not generally trigger a diagnosis alarm.

The behavior of the S7-CPU during complete failure of a configured drive or during an interruption in the bus cable can be controlled by programming the relevant system organization modules OB86 and OB122. If these system modules are not programmed, the S7-CPU also goes into the STOP state if a configured drive fails or if a bus is interrupted. Refer to Chapter 3 of the programming manual for the S7-300/400 for detailed descriptions on the indicated system organization modules.

After installation of the DVA_S7-OM, the CBP is shown as follows in the hardware catalog:



The type of PPO is selected in the hardware manager from the register "Configuration" of the "Characteristics – DP slave" mask which is automatically shown on the screen when the selection (e.g. Motion Control) is confirmed.

More information on planning data exchange between a CBP and a SIMATIC S7 can be found in the description accompanying the DVA_S7 module package.

If the DVA_S7 module package is not used, the system features regarding data consistency have to be observed by the user program. In particular, this means that access can only be made via the system functions SFC14 and SFC15 to all consistent data areas > 4 bytes.

The PKW and the PZD parts are regarded as two independent consistent data areas.

	PKW	PZD (4, 12 or 20 bytes)
PPO1	(8 bytes)	(4 bytes)
PPO2	(8 bytes)	(12 bytes)
PPO3	–	(4 bytes)
PPO4	–	(12 bytes)
PPO5	(8 bytes)	(20 bytes)

CP342-5DP

At the present time, the CBP can be operated with a CP342-5DP only as a standard DP slave because S7 functions are not yet supported by the CP342-5DP. In order to operate the CBP as a standard slave, the equipment master file or the type-description file must be incorporated into the STEP7 basic software (see integrated DP interfaces).

The DVA_S7 module package

The SIMATIC DVA_S7 module package (variable-speed drives on SIMATIC S7) implements data transfer between the drive and SIMATIC S7 in accordance with the PROFIBUS profile for variable-speed drives and thus facilitates creation of the user program. A data module with the same appearance is always provided as the data interface, irrespective of which S7 CPU the program runs on. The programmer does not therefore need any detailed knowledge of the SIMATIC S7 system architecture or of the necessary system functions.

As already mentioned, the DVA_S7 object manager is part of the scope of supply of the DVA_S7 module package.

The DVA_S7 module package can be obtained from A&D WKF Fürth/Germany under MLFB 6SX 7005-0CB00.

8.2.8.3 Operating the CBP with a non-Siemens system

When used with a non-Siemens master system, the CBP can be operated only as a standard DP slave.

Required master file

The equipment master file (GSD file) on the floppy disk contains all the information which a DP master system needs for integrating the CBP as a standard DP slave in its PROFIBUS configuration.

If the non-Siemens master system allows direct integration of a master file, the SIEM8045.GSD file can be copied into the relevant sub-directory.

If this is not possible, the required information will have to be taken from the SIEM8045.GSD file.

8.2.8.4 Operating the CBP2 with extended functions with a SIMATIC S7

The extended functions "Cross traffic" and "Clock synchronization" are described in detail in PROFIBUS Profile Drive Technology, Version 3.x, Order No. 3.171 (German), or 3.172 (English).

DriveES SlaveOM

The functions described here presuppose the planning tool, STEP7, and driveES with the slave OM for the CBP2.

- ◆ Free configuration: Up to 16 process data can be configured in each case, separated into setpoints and actual values.
- ◆ Cross traffic: Direct slave-to-slave communication without going the long way round via the DP master.
- ◆ Clock synchronization: Synchronization of master and slave applications at the isochronous PROFIBUS.

Free configuration is possible with all DP masters which are configured with STEP7.

Cross traffic and clock synchronization presuppose DP masters which support this functionality, i.e. all S7-CPU's, for example, with the characteristic "equidistance".

Configuration

For free configuration and cross traffic, carry out configuration completely with the slave OM in the "Configuration" register. In the drive, only correct interconnection of the setpoints and actual values has to be carried out.

Clock synchronization

Configure Clock synchronization with the slave OM in the "Clock synchronization" register. In addition, some parameters in the drive have to be set (MASTERDRIVES MC only).

Detailed help can be obtained in the on-line help for the slave OM.

8.2.8.5 CBP2 with cross traffic operated with a SIMATIC S7

The cross traffic function enables direct slave-to-slave communication on the PROFIBUS without having to go the long way round via the DP master. A DP master, however, is needed "to keep time".

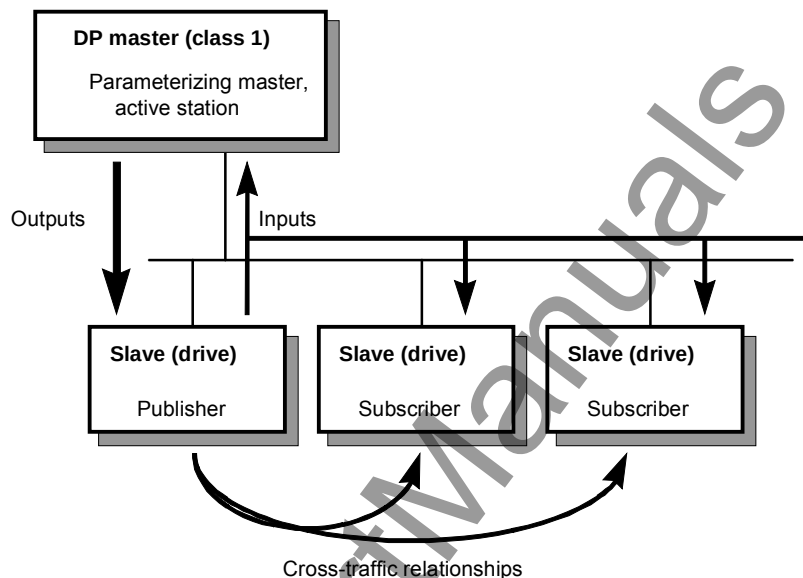


Fig. 8.2-24 Cross traffic

Configurations

With cross traffic, you can configure communication between DP slaves in various ways, e.g.

- ◆ "Broadcast": Stipulation of a master setpoint from a master drive to all drives.
- ◆ "Peer-to-peer": Passing on a setpoint from one drive to the next.

Definitions:

Encoder

- ◆ Cross-traffic encoder (publisher): All inputs of a DP slave capable of cross traffic are transmit data in relation to cross traffic. They can be received by the DP master or by DP slaves capable of cross traffic. Transmitting takes place automatically by means of a broadcast. Explicit configuration of the cross-traffic encoder is not necessary.

Receiver

- ◆ Cross-traffic receiver (subscriber): The sources for the setpoints are specified by means of configuration. The outputs of the DP master or the inputs of a DP slave as the cross-traffic encoder are possible sources (in the case of drives, their actual values). There are no restrictions on the way in which master outputs and slave inputs are mixed (with word granularity).

Drives capable of cross traffic can also receive data from themselves (feedback loop).

You need:

- ◆ STEP7 from Version 5.0 with Servicepack 2 or Servicepack 4 (Servicepack 3 is not suitable) or Version 5.1
- ◆ DriveES with slaveOM for CBP2
- ◆ S7-Profibus-Master with the "equidistance" property
- ◆ DP slaves which are capable of cross traffic as communication partners (e.g. drives or ET200)
- ◆ CBP2

Cross traffic is independent of the basic unit used. The functionality is completely provided in the CBP2.

You can configure cross traffic with the slave OM in the mask, "Configuration".

Quantities

Receive/transmit data: maximum of 16 words of setpoints/actual values per drive, can be divided up in any way on DP master and DP slaves capable of cross traffic.

Number of transmission channels: a broadcast channel which the DP master and any number of DP slaves can receive.

Number of receive channels: max. eight.

Example

The following illustration contains a cross-traffic configuration with two cross-traffic encoders (publishers) and a drive with CBP2 as the cross-traffic receiver (subscriber).

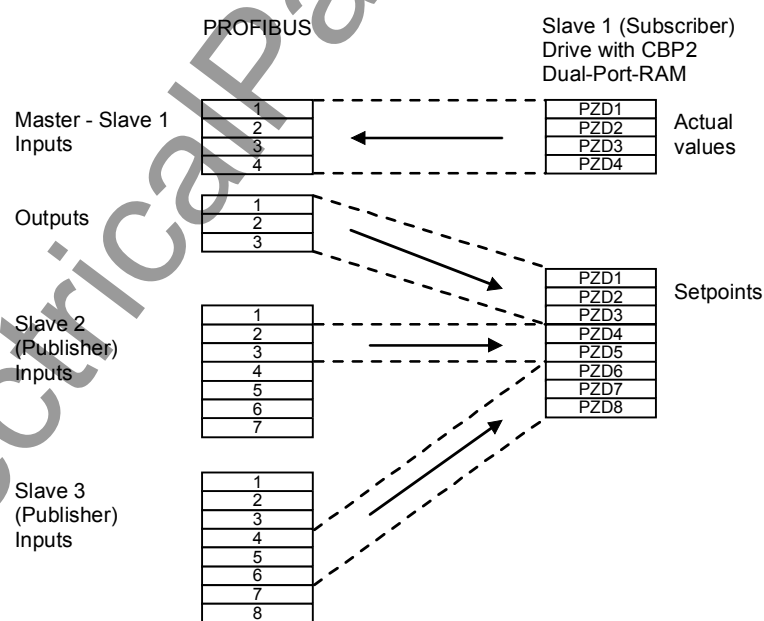


Fig. 8.2-25 Example of a cross-traffic configuration

8.2.8.6 CBP2 with clock synchronization operated with a SIMATIC S7

CBP2 with clock synchronization is not supported by MASTERDRIVES VC.

Clock synchronization enables the synchronization of several MASTERDRIVES MC at the isochronous PROFIBUS.

You need

- ◆ STEP7 Version 5 or higher
- ◆ DriveES with slave OM for CBP2
- ◆ SIMATIC S7 as PROFIBUS master with "clock synchronization" feature
- ◆ CBP2 from V2.21
- ◆ MASTERDRIVES MC (Compact or Compact PLUS) software version V1.4 or higher

You can configure clock synchronization in the Step7 HW configurator -> Object properties -> tab "Clock synchronization".

Cycle times

The isochronous DP cycle must have been tuned to the pulse frequency of the drive. The following combinations are possible with the default pulse frequency of 5.0 kHz and a PROFIBUS transmission rate of 12 Mbit/s:

Synchronized time slot in the basic unit	DP cycle	Max. number of DP slaves
T4	3.2 ms	11
T5	6.4 ms	27

STEP7 bus configuration

Measures relating to bus configuration:

- ◆ First, configure all DP slaves, possibly with cross traffic. The quantities and the minimum DP cycle on the PROFIBUS are thus defined.
- ◆ In the "Equidistance" register (can be reached under "PROFIBUS", "Properties", "Supply settings", "Options"), activate the isochronous bus cycle. You must repeat this step if you alter the bus configuration (deactivate Equidistance and re-activate)!
- ◆ In the "clock synchronization" tab of the drive, activate "Clock synchronization" for every drive. As default, time slot T4 is set at 3.2 ms.
- ◆ It must be ensured that the difference between "DP cycle" and "Equidistance master cycl. part" is at least 1 ms. The CBP2 needs this time to pass the data between PROFIBUS and the basic board consistently.
- ◆ The "Align" button ensures that all MASTERDRIVES MC are given the same settings for clock synchronization and that the DP master adopts the isochronous DP cycle.

Drive configuration

Configuration of MASTERDRIVES MC:

- ◆ Enabling the source for clock synchronization by means of P744:

P744.1	P744.2	Synchronization source
0	1	First CBP2
1	1	Second CBP2
0	0	(First SLB)
1	0	(Second SLB)

- ◆ With clock-synchronous PROFIBUS it is possible to switch on cycle monitoring with P723 = 1.
Function: Synchronization telegrams that are received outside the bus cycle ($\pm 12.8 \mu\text{s}$) are ignored.
Advantage: During faulty operation, incorrect and offset synchronization telegrams are detected and do not cause synchronization failure.
- ◆ All tasks to be synchronized have to be in the same time slot, especially the processing of setpoints and actual values to and from the CBP2.

Diagnosis

Diagnosis of clock synchronization in MASTERDRIVES MC:

- ◆ B0043 = 1: Application is synchronous with bus clock
- ◆ r748.9: should fluctuate between 65515 and 20

For further diagnostic parameters, see section "Diagnosis and Troubleshooting".

Times

The following times will help you when calculating the bus bandwidth available (12 Mbit/s):

- ◆ 150-200 μs "Equidistance master cycl. part" per DP slave (cyclical services master class 1)
- ◆ Approx. 600 μs for "Equidistance master acyclical part" (acyclical services master class 1)
- ◆ Approx. 700 μs for a max. of one additional active node (master class 2)
- ◆ 1000 μs computing time for CBP2, parallel to acyclical services

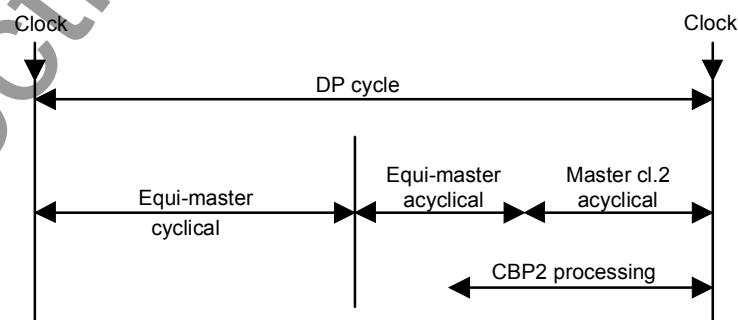


Fig. 8.2-26 Breakdown of the DP cycle at the isochronous PROFIBUS

Two time conditions which have to be complied with are

- ◆ "DP cycle" > "Equidistance master cyclical part" + 1000 µs
- ◆ "DP cycle" > "Equidistance master cyclical part" + "Equidistance master acyclical part" + "Master class 2"

Auxiliary conditions

The following auxiliary conditions are to be complied with at the present time when clock synchronization at the isochronous PROFIBUS is used:

- ◆ Transfer rate: 12 Mbit/s (the performance of lower transfer rates is too small for drives)
- ◆ Maximum number of nodes: 31
- ◆ Maximum distance: 100 m
- ◆ Only one master class 1 (the equidistance master)
- ◆ A maximum of one additional master (class 2, PG); it is recommended that no additional master be used
- ◆ No repeater, no fiber optic cables (they cause different dead times)
- ◆ The CBP2 must be directly plugged onto the basic board (also via adapter board). There must not be a technology board plugged in between the basic unit and the CBP2

Comparison PROFIBUS / SIMOLINK

PROFIBUS offers you a bus system for all tasks. With SIMOLINK, you can achieve better performance in clock synchronization. The following table shows the differences in clock synchronization:

Criterion	PROFIBUS	SIMOLINK
Medium	Copper	Glass / plastic
Distance	100 m (12 Mbit/s)	Glass: 300 m Plastic: 40 m per node
Max number of nodes	31 (no repeater)	200
Number of slaves/cycle	11 / 3.2 ms; 27 / 6.4 ms	100 per ms / < 1 ms
Max. telegram length	16 words	n times 2 words

8.2.8.7 CBP2 with clock synchronization on a PROFIBUS master in accordance with PROFIdrive V3

CBP2 with clock synchronization is not supported by MASTERDRIVES VC.

The CBP2 from V2.21 enables the clock cycles of several MASTERDRIVES MC to be synchronized according to PROFIdrive version 3 on the isochronous PROFIBUS.

You will need:

- ◆ A configuring tool that supports clock synchronization to PROFIdrive V3 (e.g. SIMOTION SCOUT)
- ◆ A PROFIBUS master that supports clock synchronization to PROFIdrive V3 (e.g. SIMOTION C, P or D)
- ◆ CBP2 from V2.21
- ◆ MASTERDRIVES MC (Compact or Compact Plus), software version V1.6

Cycle times

The pulse frequency of the drive must be set to 5.3 kHz for clock synchronization in accordance with PROFIdrive V3. The following combinations are possible with a PROFIBUS transmission rate of 12 Mbit/s:

Synchronized time slot in basic unit	DP cycle
T4	3 ms
T5	6 ms

Drive configuring

Configuring the MASTERDRIVES MC:

- ◆ Set pulse frequency to 5.3 kHz. Select P340 = 5.3 in the drive settings screen.
- ◆ Enable the source for clock synchronization via P744:

P744.1	P744.2	Synchronization source
0	1	First CBP2
1	1	Second CBP2
0	0	(first SLB)
1	0	(second SLB)

- ◆ A cycle monitoring function can be activated with P723 = 1 for the isochronous PROFIBUS.
Function: Synchronization telegrams that are received outside the bus cycle ($\pm 12.8 \mu\text{s}$) are ignored.
Advantage: Under fault conditions, faulty and displaced telegrams are detected so that synchronization errors can be avoided.
- ◆ All the tasks to be synchronized must be in the same time slot, especially the processing of setpoints and actual values to and from the CBP2.

Diagnosis

Diagnosis of clock synchronization in MASTERDRIVES MC:

- ◆ B0043 = 1: Application is synchronous with bus clock
- ◆ r748.9: Should fluctuate between 65515 and 20

For other diagnostic parameters, see Section "Diagnosis and troubleshooting"

Times

The following time data help you to calculate the available bus bandwidth (12 Mbit/s):

- ◆ 150-200 μ s "Isochronous master cyclical share" per DP slave (cyclical services class 1 master)
- ◆ Approx. 600 μ s for "Isochronous master acyclical share" (acyclical services class 1 master)
- ◆ Approx. 700 μ s for max. one further active node (class 2 master)
- ◆ 1000 μ s computation time for CBP2, in parallel to acyclical services

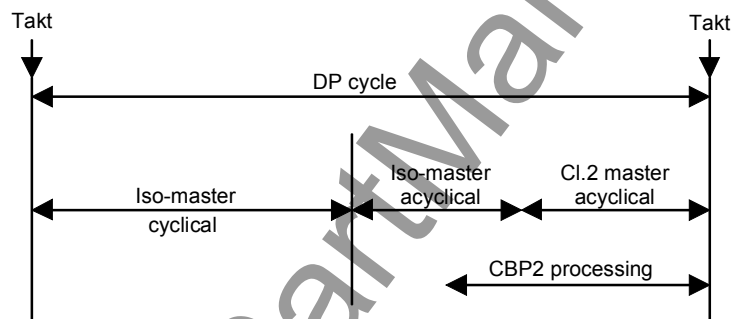


Fig. 8.2-27 Splitting of DP cycle on isochronous PROFIBUS

Two mandatory time conditions:

- ◆ "DP cycle" > "Isochronous master cyclical share" + 1000 μ s
- ◆ "DP cycle" > "Isochronous master cyclical share" + "Isochronous master acyclical share" + "Class 2 master"

Supplementary conditions

The following supplementary conditions currently apply to clock synchronization on the isochronous PROFIBUS:

- ◆ Transmission rate: 12 Mbit/s (performance afforded by lower transmission rates is too low for drives)
- ◆ Maximum number of nodes: 31
- ◆ Maximum distance: 100 m
- ◆ For class 1 master only (isochronous master)
- ◆ A maximum of one other master (class 2, PG), ideally no other master
- ◆ No repeaters, no fiber optics (which cause varying deadtimes)
- ◆ The CBP2 must, of course, be inserted directly in the basic board (or via an adapter board). No technology board may be installed between the basic board and the CBP2.

8.2.9 MASTERDRIVES as PROFIdrive V3-Slave

CBP2 with clock synchronization is not supported by MASTERDRIVES VC.

Requirements

Variable speed electric drives in automated units, from the simple frequency converter to the highly dynamic servo controller, are now increasingly being connected to higher-level open-loop and closed-loop control systems via digital interfaces.

In current systems the standard speed interface is one in which the speed command is given by the higher-level automation system and is tracked by the drive. For monitoring, the actual speed value is generally reported back to the automation system.

So that the digital field bus interface can also be used for motion control with multiple axes in decentralised automation concepts, the current standard field buses must be supplemented by specific properties.

The following requirements must be met:

Clock synchronization

In the case of a central motion controller, which performs interpolation and position control, the control circuit must be connected via the bus. In the setpoint direction the speed setpoint is transmitted to the drive. In the actual value direction the drive returns the actual position value. In order to be able to achieve sufficiently high circuit gain for the dynamic required, the dead times must be small and above all absolutely constant. If the motion control task requires coordination of several axes, the position actual value must be recorded exactly simultaneously and evaluated synchronously in the motion controller, and the setpoints must take effect exactly simultaneously in the axes. Actual value recording, transmission and setpoint setting are therefore clock synchronous with the position controller.

New functions of the PROFIdrive-Profile Version 3

This document contains the relevant extracts from the Profi-Drive-Profil document (PROFIBUS Nutzerorganisation e. V., Order no.: 3.171, <http://www.profibus.com>), in which the use of clock synchronous transmission, of cross traffic and a position encoder interface and the standardisation and configuration of the setpoints and actual values for the drives are manufacturer-neutral and openly standardised.

Definitions**General**

- ◆ **Output data:**
Data which a slave receives cyclically from the master and forwards to the slave application or the I/O's.
- ◆ **Input data:**
Data sent cyclically to the master by a slave.
- ◆ **Process data:**
For drives, all input and output data
- ◆ **Technological functions:**
Controls **and** sequence control for automation of application-specific processes

Clock synchronization

- ◆ **Synchronization:**
Creating synchronism (simultaneity, time consistency)
- ◆ **Clock synchronization:**
Clock synchronization refers to the synchronization of the sample times of the control with the connected drives.
- ◆ **Clock synchronous application:**
in the control software in digital drives and control system. The starting times and the length of the sampling times in various devices are synchronized exactly with each other.
- ◆ **Equidistance:**
The same distance. The bus cycle time is always constant. Any free bus time is filled up with blank telegrams.
- ◆ **Isochronous mode:**
Profibus service for clock synchronization; produces a time-constant, i.e. equidistant bus cycle with a clock pulse signal at the start of the cycle.

8.2.9.1 Incorporation of drives in automation systems / plant characterization

This section presents the different variants for incorporation of drives in automation systems.

Application classes

Nowadays, drive applications are implemented in many different ways. The following table defines the various application classes in which drives are used. The application classes are typical examples from the total spectrum of electrical drive engineering and are not necessarily covered by a specific equipment characteristic.

	Application class	Interface	Functions 2)
1	Standard drive	n-set, i-set	Cyclical interface 1)
2	Standard drive with distributed technology controller (continuous process)	Technological set/actual values (command variable)	Cyclical interface with cross traffic 1)
3	Basic positioning drive with distributed position control and interpolation	pos-set, travel tasks	Cyclical interface 1)
4	Positioning with central interpolation and position control Optional: DSC (Dynamic Servo Control)	n-set x-actual additional for DSC: Δx (x_{err}), K_v (k_{PC})	Cyclical interface clock synchronous encoder interface, DSC (see chapter 2.4)
5	Positioning with central interpolation and distributed position control	x-set	Cyclical interface clock synchronous 3)
6	Motion control in clocked processes or distributed angular synchronism	Command variable, motion commands	Cyclical interface clock synchronous and with cross traffic

1) The cyclical interface can also be operated clock synchronously if, for example, simultaneity of action in several drives is required.

2) For all application classes: Acyclical interface for parameters, diagnosis, identification

3) This application class is not described in this document.

Table 8.2-15 Application classes

NOTE

When the MASTERDRIVES unit is used as PROFIdrive V3 slave, only application classes 1 and 4 are used!
Several MASTERDRIVES can be connected to **one** PROFIdrive V3 master in class 1 as slave (Monomaster operation).

Standard drive (Standard telegram 1 and 2)

Application class 1:

In the simplest case the drive is controlled via a main setpoint (e. g. speed setpoint) via PROFIBUS (Figure 1). The complete speed control takes place in the drive controller. The automation device includes all the technological functions for the automation process. PROFIBUS acts only as transmission medium between the automation system and drive controller. The normal cyclical data communication of the PROFIBUS-DP is used (Data Exchange). This application is primarily used in the area of classic drive technology (e.g. materials handling technology). A PLC is used mainly as the automation system. Clock synchronism and cross traffic on the bus system are not necessary for this application class.

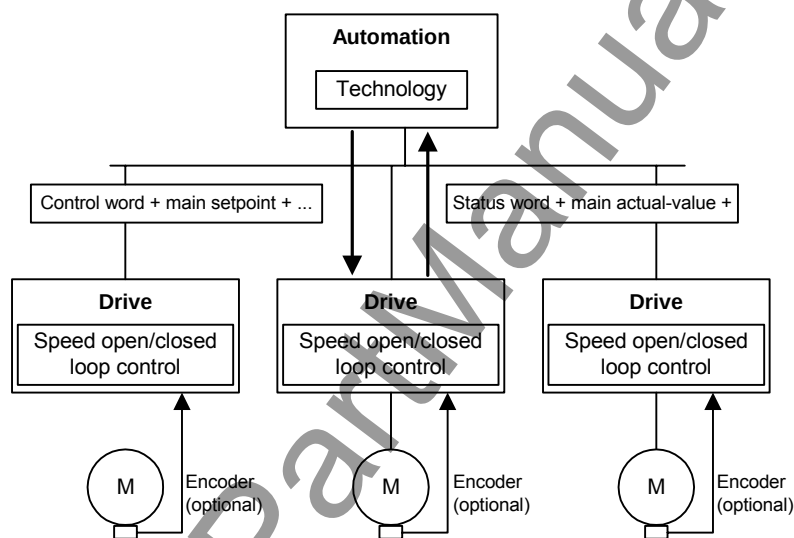


Fig. 8.2-28 Application class 1

Positioning with central interpolation and position control (standard telegram 3 to 6)

Application class 4

Application class 4 (Figure 2) handles the position control through PROFIBUS-DP. Drive applications for handling equipment and robot applications often require a coordinated motion cycle by several drives. The motion control is mainly implemented via a central automation unit (NC). These controls calculate special setpoint profiles for each drive, so that certain tracks can be travelled through the combined action of several drives (for example for the XYZ axis). The automation system comprises not only the necessary technology functions for the automation process but also the functions for interpolation and position control of the drive. Speed setpoints and actual values and actual position values are exchanged via PROFIBUS-DP. The drive controller essentially only comprises the algorithms for speed control and recording of actual position values. As position control is carried out via the bus system, this variant places very high demands on the clock synchronisation of the bus system.

To increase the stiffness and dynamism of the control circuit, the DSC functionality can be used additionally.

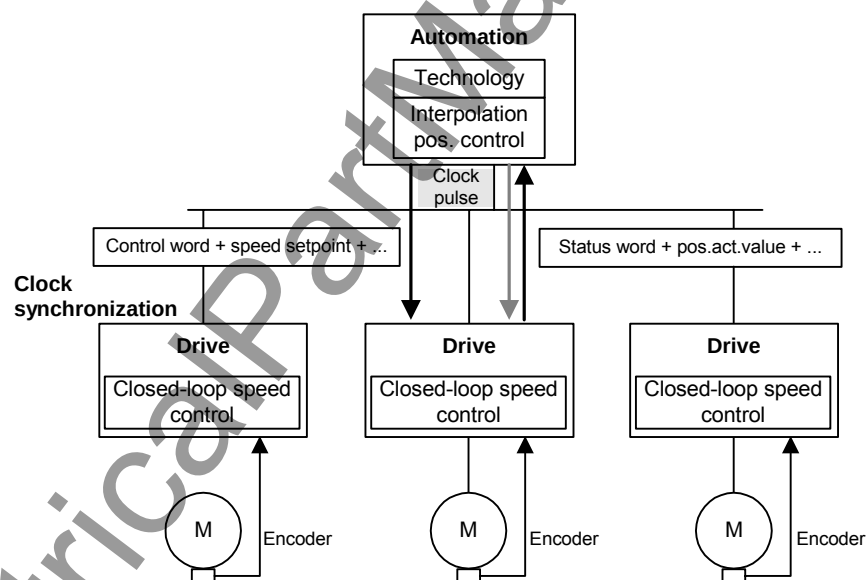


Fig. 8.2-29 Application class 4

Clock synchronous monomaster operation (main application case)

With this operation the drives make high demands on time synchronism at the bus. Through an isochronously transmitted global control the master class-1 (e.g. SIMOTION) passes on clock pulse information to the slaves (MASTERDRIVES), which synchronise themselves with this pulse.

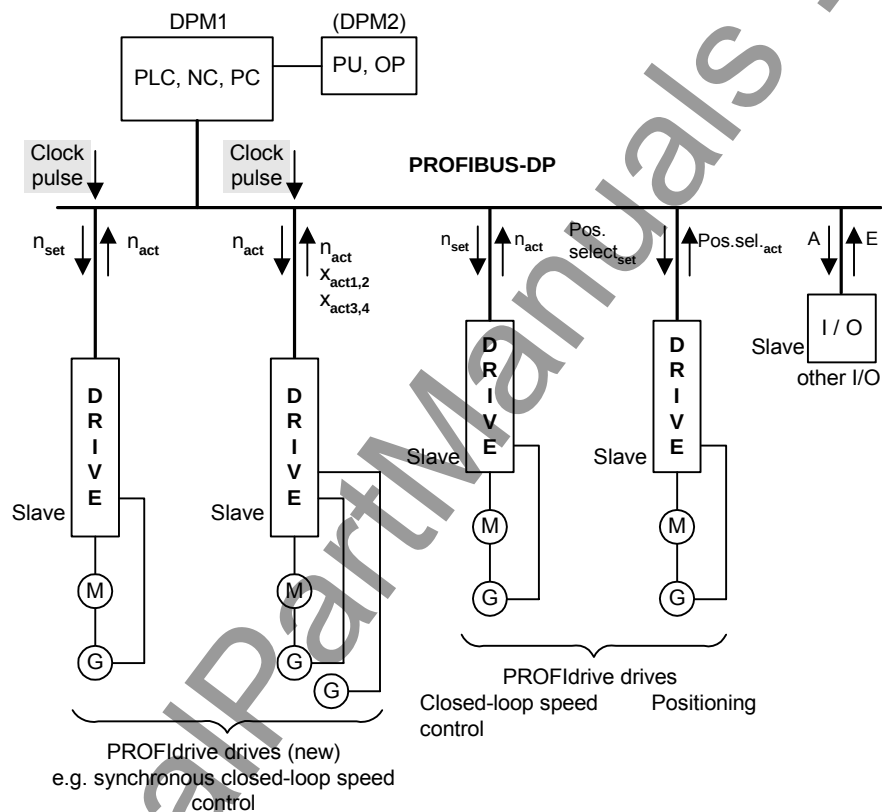


Fig. 8.2-30 Clock synchronous monomaster operation (DPM1)

The times for the setpoint transfer and actual value recording of the slaves and the time of the control by the higher-level master can take place through time parameters. The time parameters relate to the clock pulse.

In the DP cycle there must be sufficient time for the following communication elements:

- ◆ cyclical data exchange with all slaves on the bus
- ◆ a non-cyclical data channel (DPV1)
- ◆ telegram repeats
- ◆ diagnosis request

A local control device (PU, OP) at the DPM1 must communicate with the slaves via the DPM1 master.

8.2.9.2 Communication model

Clock synchronous communication

Clock synchronous communication is implemented by the use of an isochronous clock pulse signal on the bus system. This cyclical, isochronous pulse is transmitted as global control signal by the master to all bus participants. Master and slave can thus synchronise their applications to this signal.

Special error mechanisms in every participant permit stable communication even with sporadic failure of the system pulse.

For drive technology the clock synchronous communication forms the basis for drive synchronization. For this not only is the telegram traffic on the bus system implemented in an isochronous time slot, but also the internal control algorithms, such as the speed and current controllers in the drive or the controllers in the higher-level automation system, are synchronized in time.

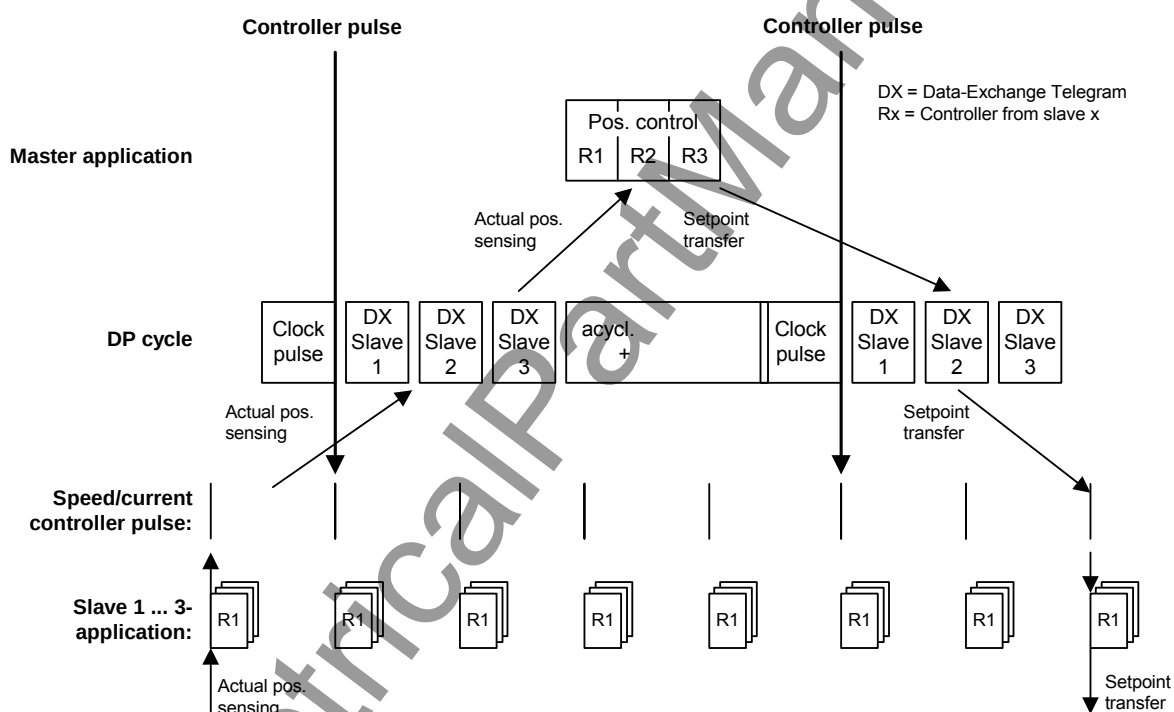


Fig. 8.2-31 Clock synchronous communication

8.2.9.3 Drive control

Commands (control words)

Control word 1

Contents of the bits in PROFIdrive control word 1. The existing bits are contained in MASTERDRIVES control word 1 (see also Chapter 10.1).

Bit	Meaning	
	Operating mode speed control	Operating mode positioning
0	ON / OFF 1	
1	Operating condition / OFF 2	
2	Operating condition / OFF 3	
3	Enable operation / disable operation	
4	Operating condition / disable ramp-function generator	Operating condition / reject travel task
5	Enable ramp-function generator / stop ramp-function generator	Operating condition / pause
6	Enable setpoint / disable setpoint	Activate travel task (edge)
7	Acknowledge / no meaning	
8	Inching 1 ON / Inching 1 OFF	
9	Inching 2 ON / Inching 2 OFF	
10	Control from PLC / no control	
11	Device-specific	Start referencing / terminate referencing
12 - 15	Device-specific	

Explanation: To the left of the forward slash is the meaning for bit value = 1, to the right that for bit value = 0.

Control word 2

Contents of the bits in PROFIdrive control word 2.

IMPORTANT

This is different from the previous MASTERDRIVES control word 2.

Bit	Meaning
0 - 11	Device-specific
12 - 15	Master life sign for clock synchronization

8.2.9.4 Checkback messages (status words)

Status word 1 Contents of the bits of PROFIdrive status word 1. The existing bits are contained in MASTERDRIVES status word 1 (see also chapter 10.2).

Bit	Meaning	
	Operating mode speed control	Operating mode positioning
0	Ready to start / Not ready to start	
1	Ready for operation / Not ready for operation	
2	Operation enabled / Operation disabled	
3	Fault / Fault-free	
4	no OFF 2 / OFF 2	
5	no OFF 3 / OFF 3	
6	Start disabled / start not disabled	
7	Warning / no warning	
8	Setpoint / actual in tolerance range / Setpoint/actual not in tolerance range	no tracking error / tracking error
9	control required / local operation	
10	f or n reached / f or n not reached	set position reached / outside set position
11	device-specific	Reference point set / no reference point set
12	device-specific	setpoint acknowledgment (edge)
13	device-specific	drive stationary / drive travelling
14 - 15	device-specific	

Explanation: To the left of the forward slash is the meaning for bit value = 1, to the right that for bit value = 0.

Status word 2 Content of bits in PROFIdrive status word 2.

IMPORTANT This differs from the previous MASTERDRIVES status word 2.

Bit	Meaning
0 - 11	Device-specific
12 - 15	Slave life sign for clock synchronization

8.2.9.5 Setpoints / Actual values

Both the setpoints on the drives and the actual values from the drives are transmitted as PZD (process data). Process data transmission takes place by means of the Data Exchange Service.

Standard signals

All the abbreviations for the standard telegrams are explained below.

Abbreviation	Meaning	Length [16-/32-Bit]	Description
STW1	Control word 1	16	see section 8.2.9.3
ZSW1	Status word 1	16	see section 8.2.9.4
STW2	Control word 2	16	see section 8.2.9.3
ZSW2	Status word 2	16	see section 8.2.9.4
NSOLL_A	Speed setpoint A	16	
NIST_A	Speed actual value A	16	
NSOLL_B	Speed setpoint B	32	
NIST_B	Speed actual value B	32	
G1_STW	Encoder-1 control word	16	see section 8.2.9.9
G1_ZSW	Encoder-1 status word	16	see section 8.2.9.9
G1_XIST1	Encoder-1 position actual value-1	32	see section 8.2.9.9
G1_XIST2	Encoder-1 position actual value-2	32	see section 8.2.9.9
G2_STW	Encoder-2 control word	16	see section 8.2.9.9
G2_ZSW	Encoder-2 status word	16	see section 8.2.9.9
G2_XIST1	Encoder-2 position actual value-1	32	see section 8.2.9.9
G2_XIST2	Encoder-2 position actual value-2	32	see section 8.2.9.9
XERR	Control error	32	
KPC	Position controller gain factor	32	

Table 8.2-16 Standard signals– Abbreviations

Standard telegrams, see Subsection 8.2.7.3.

8.2.9.6 Dynamic Servo Control (DSC)

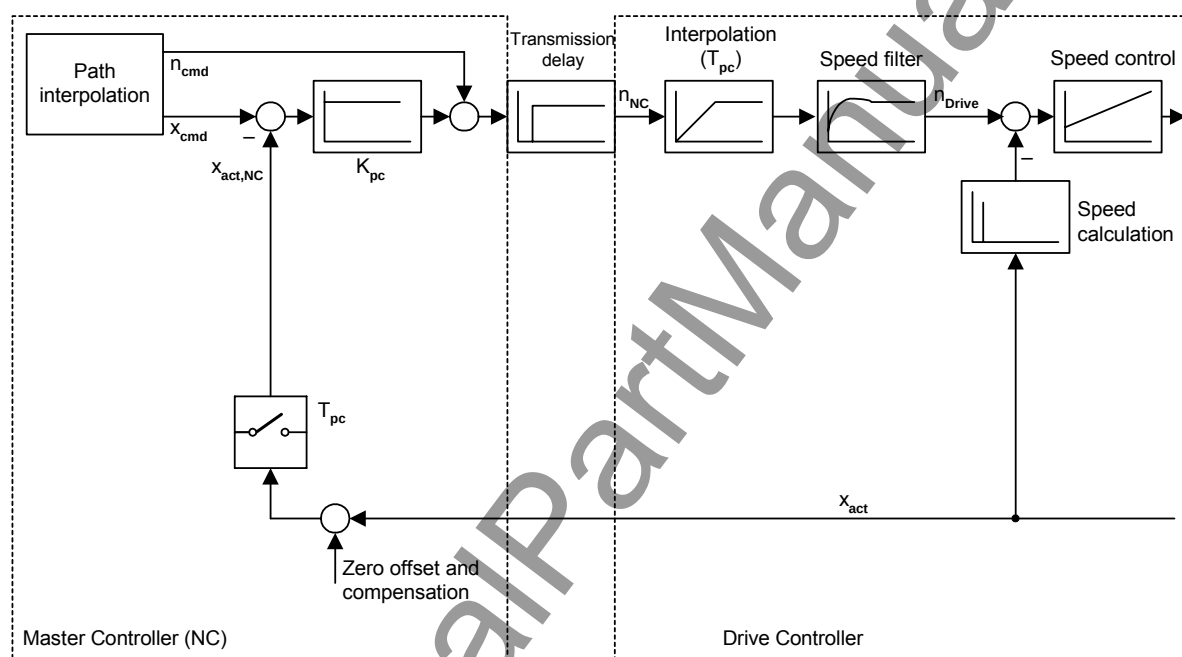
Features

This function improves the dynamic of the position control circuit, in that it minimizes the dead times which normally occur with a speed setpoint interface. For this only a relatively simple extension of the transmitted setpoints and an additional feedback network in the drive are needed.

The function is upwardly compatible with the speed setpoint interface. If necessary this can be switched to the speed setpoint interface during operation.

Structure

The control circuit based on the speed setpoint interface generally has the following structure:



n_{cmd} : speed command
 x_{cmd} : position command
 x_{err} : position error command

x_{act} : actual position
 T_{pc} : position controller sampling time ($= T_{MAPC}$)
 K_{pc} : position controller gain

Fig. 8.2-32

Structure of the position control circuit based on the speed setpoint interface without DSC

With DSC the actual position calculated inside the drive is also fed back directly:

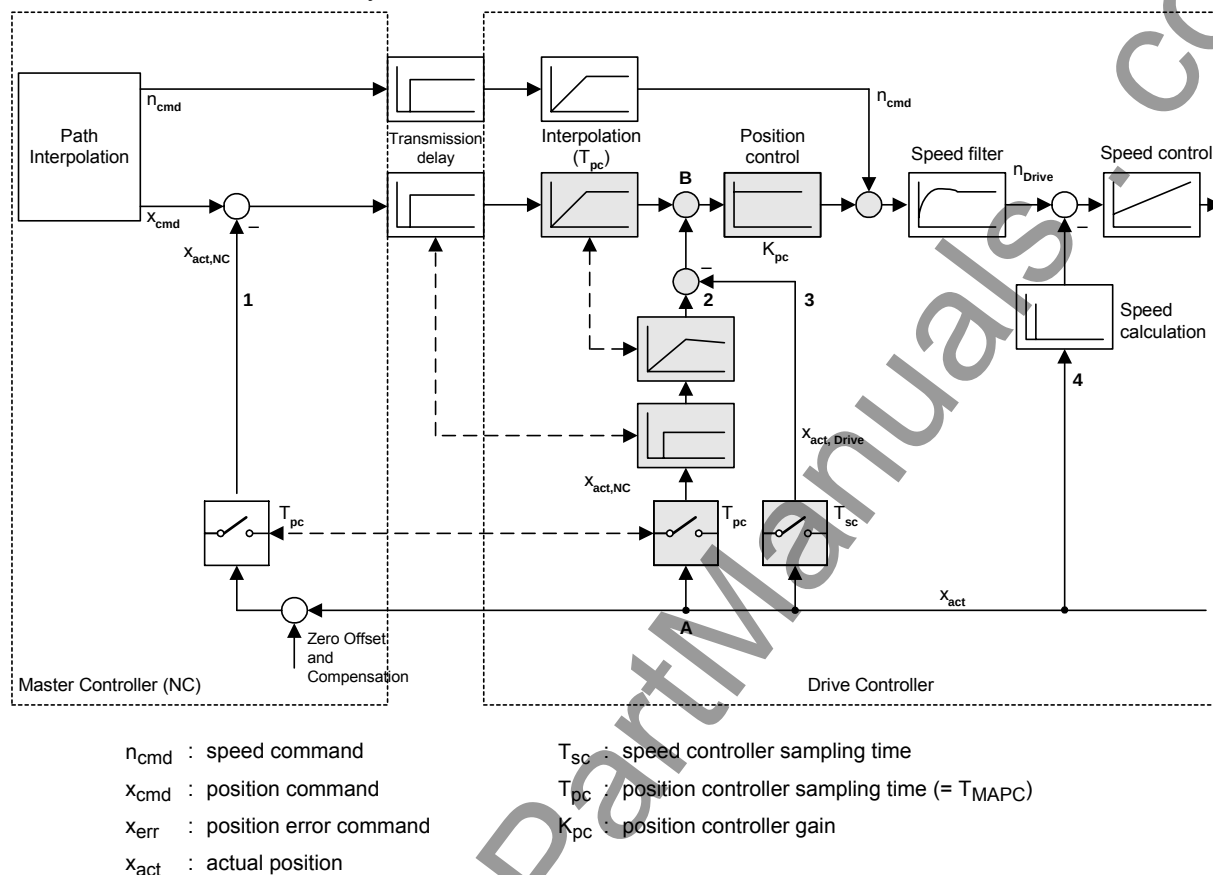


Fig. 8.2-33 Structure of the LR circuit based on the speed setpoint interface with DSC

For this to be possible, the control error calculated in the master is transmitted as well as the speed setpoint. The additional feedback network can use the internal formats in the drive for position presentation and is thus independent of the position presentation in the master. The above presentation assumes that the network is calculated in the speed controller sampling time T_{SC} which will be possible in many cases. In this way the maximum possible dynamic improvement is achieved. However, greater clock times T are also possible, if the calculation time is short ($T_{SC} \leq T \leq T_{PC}$).

Operation

The structure contains a total of three feedback branches for the actual position value (No. 1, 2 and 3). Feedback branch no. 2 compensates fully for the action of no. 1 in respect of the actual value transmitted by the drive x_{act} , so that the dead time in branch no. 1 no longer has to be considered for the stability of the position control circuit. As a result, the position control circuit is initially open. Feedback branch no. 3 closes the circuit again, but with a shorter delay, so that greater gains can be set.

The absolute reference of the position actual value is only created in the master (addition point "Zero offset and compensation"). The same absolute reference is contained in the position command x_{cmd} . The control error calculated in the master x_{err} thus remains free of zero points. The drive needs to know nothing about zero points and reference points.

Interface

In the setpoint direction two additional signals are transmitted:

1. Controller error x_{err}
2. Position controller gain factor k_{pc}

The standard telegrams 5 and 6 defined for the function Dynamic Servo Control (DSC) are explained further in section 8.2.7.3.

If both signals x_{err} or k_{pc} are configured, the feedback network in the drive is activated. If only one of the two signals is projected, it is assumed that this serves other purposes and the feedback network is not activated.

The position controller gain K_{PC} transmitted via the Profibus has the unit 1/1000 1/s.

Operating statuses

From the drive point of view there are two operating statuses, which can be distinguished on the basis of $k_{pc} = 0$ or $k_{pc} \neq 0$:

1. $k_{pc} = 0$: feedback network inactive, position control circuit in the drive opened. The master normally uses this to open the position control circuit fully, e.g. in spindle operation or with errors. However, it can also switch back to conventional position control in this way, without re-configuring the drive. The drive can assume that $x_{err} = 0$ is being transmitted. The speed setpoint is entered via n_{cmd} .
2. $k_{pc} \neq 0$: feedback network active, position control circuit is enclosed in the drive. Via n_{cmd} a speed pre-control value is entered, which may also be zero.

A switch between these two statuses can be made by the master at any time. The master can also alter the value of k_{pc} at any time, e.g. in order to make dynamics adjustments in gear changes or compensation for non-linear gears.

Boundary conditions

Feedback branch 2 must exactly simulate the action of feedback branch 2 between points A and B. Both branches must

1. work with an actual value originating from the same time and scanned with the same frequency
2. display the same deceleration
3. contain the same fine interpolation

This is indicated by the broken arrows in the structure diagram.

The speed filter shown in the structure diagram is optional and has nothing to do with the DSC function. It has been drawn in to make the difference from conventional position control easier to recognize.

Sample application

Linking the MASTERDRIVES MC to SIMOTION using standard telegram 5.

Fig. 8.2-34 shows an overview of the interconnection required for standard telegram 5. This connection is made using the scriptfile on the SIMOTION CD:

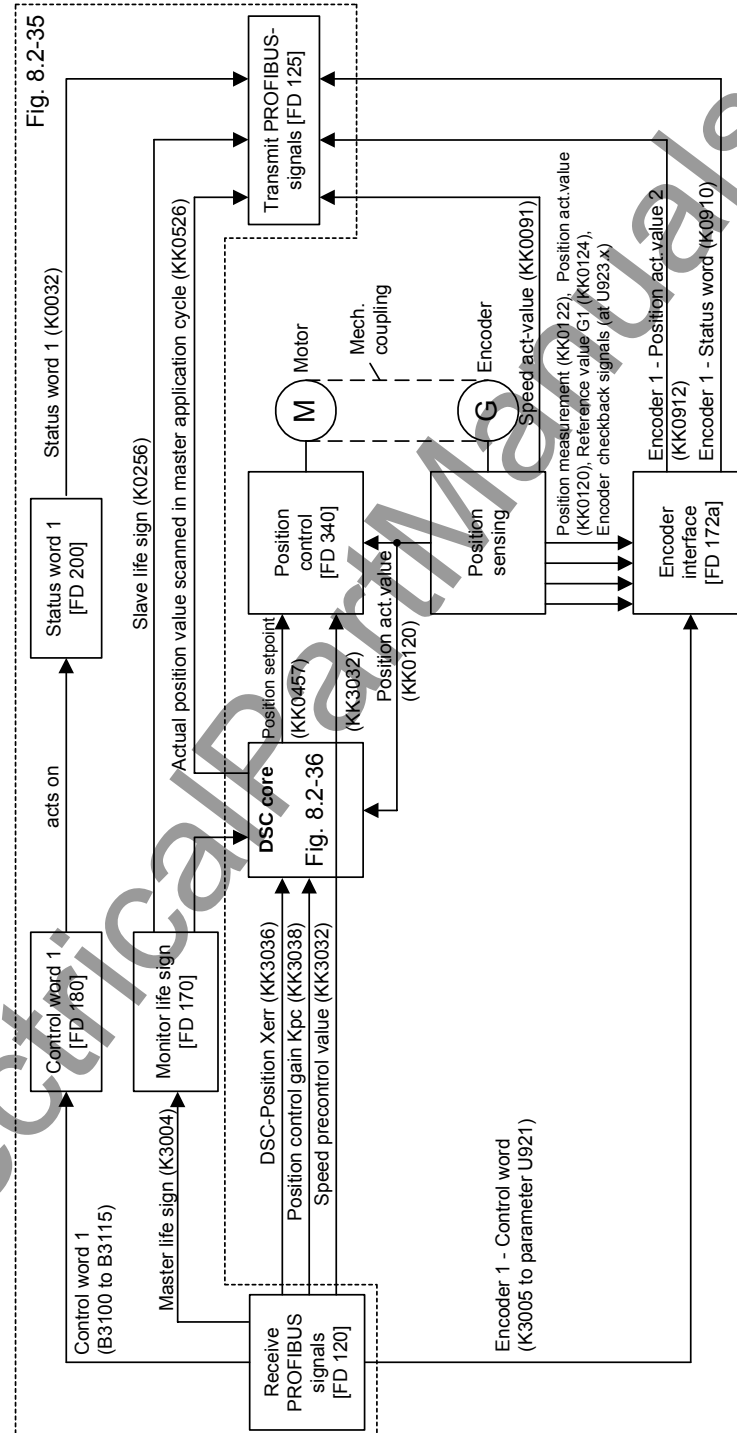


Fig. 8.2-34 Block diagram: MASTERDRIVES as PROFIdrive version 3 slave

In Fig. 8.2-35 you see on the left reception of the setpoints and control signals from the Profibus, on the right the transmission of actual values and status data to the Profibus, as is specified in the standard telegram. In the centre of Fig. 8.2-35 the meaning and softwiring of the individual bits in control word 1 are presented. At the bottom of Fig. 8.2-35 is the monitoring of the master life sign, the creation of the binector “master application pulse” and the slave life sign.

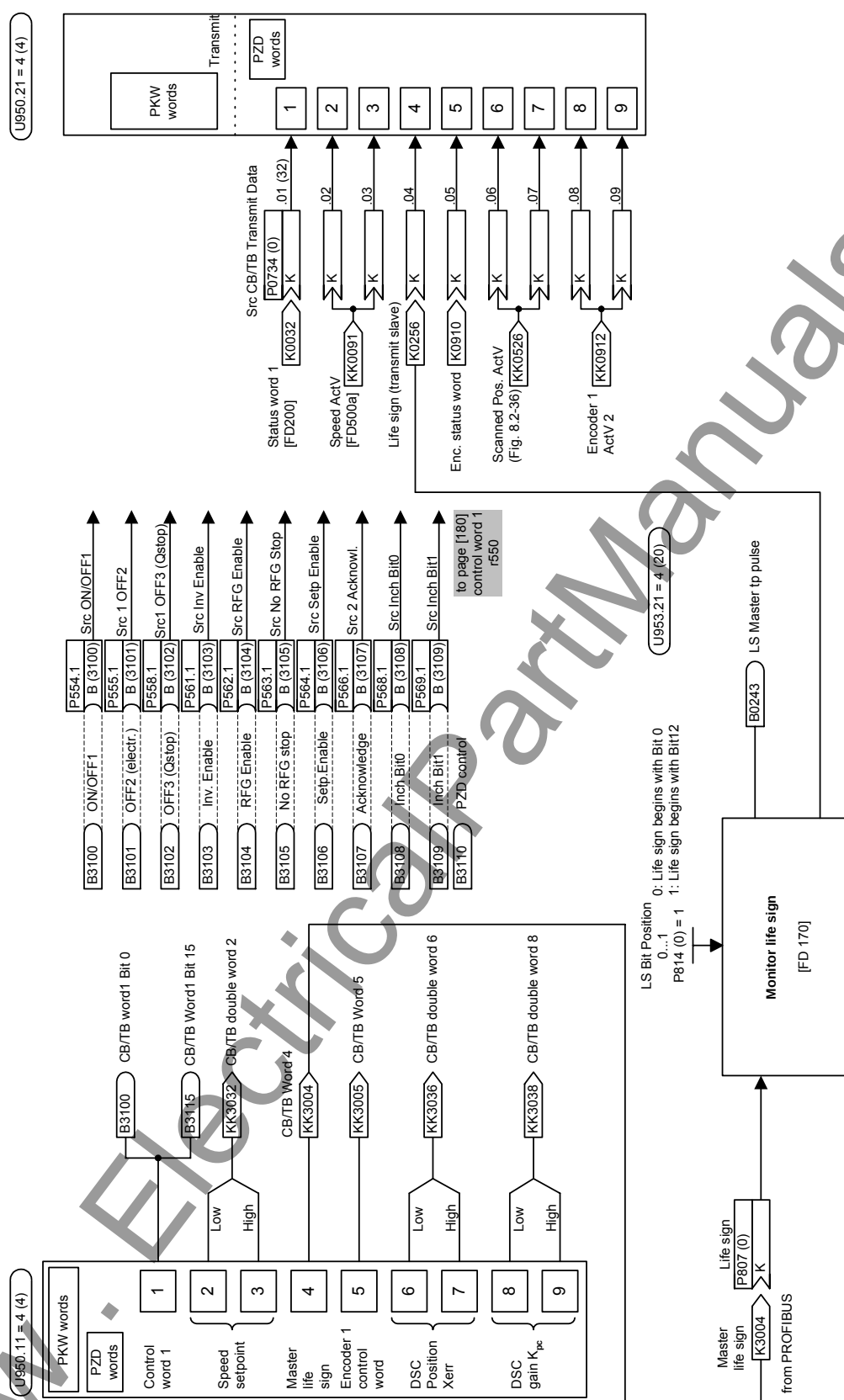


Fig. 8.2-35 PROFIBUS communication

Implementation of DSC with free blocks (see Fig. 8.2-36)

In MASTERDRIVES free calculating blocks are available, with the aid of which the DSC function is implemented.

The individual elements of this implementation are described below:

**Shift multiplier top
centre**

The shift multiplier is used for conversion of the DSC gain DSC_GAIN from an integer to a percentage value, which in turn is needed as input for the KP adjustment position controller.

**Switch bottom
centre**

If the master application pulse (MAPC) is a multiple of the Profibus (DP) pulse, the relevant position actual value, which enters the position controller on the master side, must be stored internally. This is achieved by scanning the actual position value synchronously with the life sign of the Profibus master.

**Delay element
bottom right**

This dead time element delays the scanned actual position value. The dead time must be selected according to the action time of a new actual position value via the master position controller. With the combination of SIMOTION as bus master and MASTERDRIVES as slave this dead time in our experience is always 4 DP cycles, so the value 4 should be entered in U401.

Adder centre right

With the aid of this adder the current position setpoint is calculated from the DSC position Xerr and the delayed actual position value.

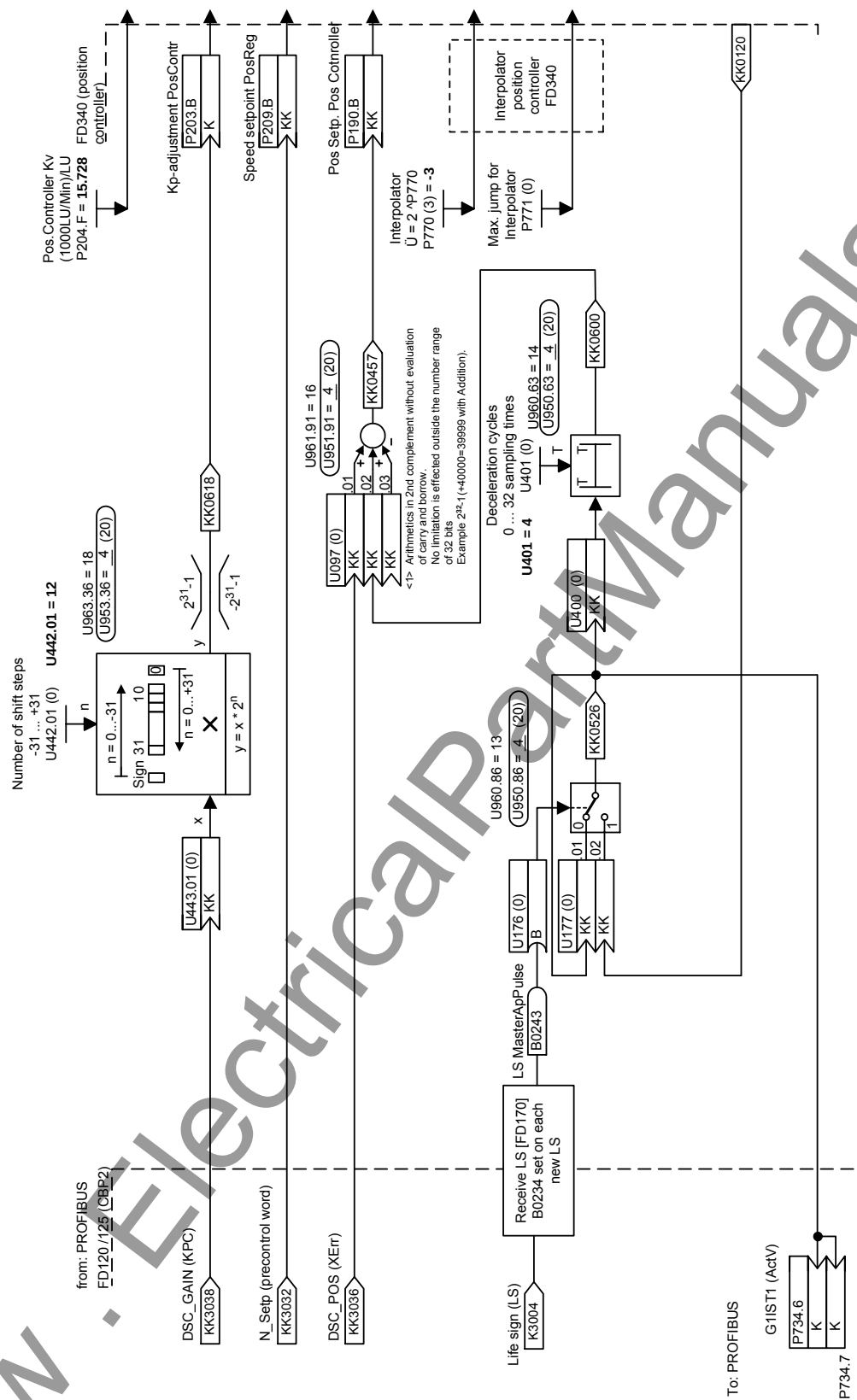


Fig. 8.2-36 DSC core

8.2.9.7 Communication interface

Node address

P918: Node address

Value range 0 – 125 (126 is reserved for start-up purposes)

The node addresses 0, 1 and 2 are generally occupied by master and configuration tools and therefore should not be used for slaves on the PROFIBUS. Address 3 is the first appropriate node address to be used for a slave on the PROFIBUS.

8.2.9.8 Clock synchronous application

Course of an isochronous DP cycle

Example (simplest DP cycle, standard case for MASTERDRIVES)

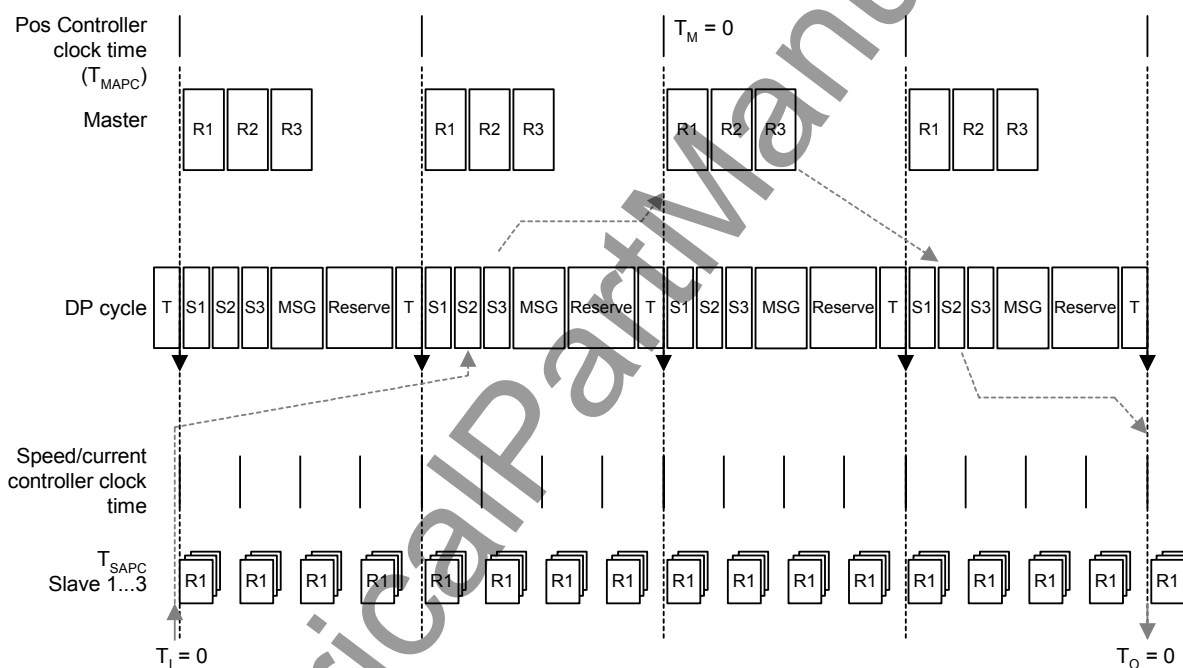


Fig. 8.2-37 Example: simplest DP cycle

In this example four DP cycles are needed for a response in the position control circuit.

- ◆ 1. Actual value recording (in slave)
- ◆ 2. Actual value transmission (slave -> master)
- ◆ 3. Position controller (in master)
- ◆ 4. Setpoint transmission (master -> slave)

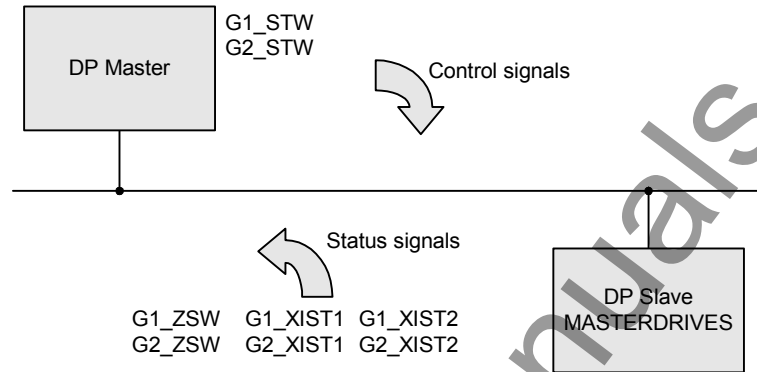
This model makes few demands on the computing output of the master, but leads to an increase in the control-specific dead time:

$$\text{Dead time} = 4 \cdot T_{DP}.$$

8.2.9.9 Encoder interface (from SW 1.6)

Process data of the encoder interface

The encoder interface consists of the following process data:



NOTE:

G1_... Encoder 1 -> Motor encoder

G2_... Encoder 2 -> External enc.

Fig. 8.2-38

NOTES

- ◆ The process data of the encoder interface can be incorporated in the telegram via the process data configuration
Encoder 1: Standard telegram 3, 4, 5, 6
Encoder 2: Standard telegram 4, 6

The description of these process data can be obtained from the literature:
Literature: /PPA/, PROFIDRIVE Profile Drive Technology (order no. 3.171) Chapter 4.6.
- ◆ For operation of a MASTERDRIVES MC on SIMOTION or another PROFIdrive Master using standard telegrams 3 to 6, the encoder interface in the frequency converter according to PROFIdrive Profil Drive Technology Version 3 should be used.
- ◆ The encoder interface uses the basic unit functionality of MASTERDRIVES MC. The description can be taken from this Compendium on the basis of the function diagrams.

Gx_STW
Encoder x-control word

x: Place-marker for encoder 1 or 2 to control the encoder functionality

Bit	Value	Meaning	Comments
0	1	Functions:	Function 1-4:
1	1	Reference marks	Request reference marks search (Bit 7 = 0)
2	1	Search	Bit 0: Function 1 (Reference mark 1)
3	1	or	Bit 1: Function 2 (Reference mark 2)
		flying	Bit 2: Function 3 (Reference mark 3)
		measurement	Bit 3: Function 4 (Reference mark 4)
			Request flying measurement (Bit 7 = 1)
			Bit 0: Function 1 (Scanner 1 pos. edge)
			Bit 1: Function 2 (Scanner 1 neg. edge)
			Bit 2: Function 3 (Scanner 2 pos. edge)
			Bit 3: Function 4 (Scanner 2 neg. edge)
4-6	1-3		Command:
			0: ----
			1: Activate function x Bit 4=1
			2: Read value x Bit 5=1
			3: Terminate function x Bit 4 and Bit 5 =1
			4-7: reserved
7	0 / 1		Mode:
			Bit 7 = 0: Reference marks search (zero mark or zero mark and BERO)
			Bit 7 = 1: Flying measurement (only BERO)
8			reserved
9			reserved
10			reserved
11	0 / 1	Reference point mode	Reference point mode:
			Bit 11 = 0: Set reference point
			Bit 11 = 1: Shift reference point
12	1	Request to set / shift reference point	Request to set/shift reference point. The setting or shift value can be adjusted according to the device. Consideration in Gx_XIST1, Gx_XIST2
13	1	Request absolute value cyclically	Request for additional cyclical transmission of the absolute position actual value in Gx_XIST2. Use e.g. for: - additional measuring system monitoring - synchronization in run-up

Bit	Value	Meaning	Comments
14	1	Activate parking encoder	Request to switch off monitoring of the measuring system and actual value recording in the drive. It is then possible to remove one encoder (or motor with encoder) on the machine, without having to change the drive configuration, or without causing a fault. NOTE: Before the encoder is de-parked, an acknowledgement of the outstanding encoder errors must be carried out (ACKNOWLEDGMENT to the PMU) WARNING: Except absolute value encoder P183.1 = xx2x. Here the parking of the encoder is terminated with error! (Error code = 1) Absolute value encoders may never be removed from the frequency converter while live!
15	1	Acknowledge encoder error	Request to return a encoder error (Gx_ZSW, Bit15) .

Table 8.2-17 Encoder control word

Re function 1 - 4 (BIT 0 to BIT3):

The function bits are forwarded to the binectors B910 to B917 (see also FP172x) for optional functionalities.

G1STW Bit 0 = B910

G2STW Bit 0 = B914

Bit 1 = B911

Bit 1 = B915

Bit 2 = B912

Bit 2 = B916

Bit 3 = B913

Bit 3 = B917

Applies only to flying measurement (configuration of digital input)

Bit	Meaning		
0	Function 1	Digital input 4 enabled	Scanner positive edge (B929)
1	Function 2	Digital input 4 enabled	Scanner negative edge (B930)
2	Function 3	Digital input 5 enabled	Scanner positive edge (B926)
3	Function 4	Digital input 5 enabled	Scanner negative edge (B927)

NOTE

-
- ◆ Bit x = 1 Function active
 - ◆ Bit x = 0 Function inactive
 - ◆ In P647.B for Dig. Inp. 4 or. P648.B for Dig. Inp. 5 it is set whether there is a configuration of the position measurement memory per binector.
In that case the P647/P648 must be set to 5 (see also function diagram 90 terminals / digital inputs).
 - ◆ The use of the configuration of the digital inputs is created per binector circuit (see also FP172x).
-

Re function set / shift reference point (BIT 12):

- ◆ Set reference point motor encoder is issued to B920
- ◆ Shift reference point motor encoder is issued to B922
- ◆ Set reference point external encoder is issued to B921
- ◆ Shift reference point external encoder is issued to B923

NOTE

The use of this function is created per binector circuit (see also FD172x).

Input Src measured value valid U923.7 (G1) and U923.8 (G2):

The binector 70 or 71 measured value is validly evaluated.

If a measuring command is generated without valid measured value, it leads to the error "encoder interface" (status SD3).

The error code 4 to 7 is generated, depending on the status of the encoder interface.

Encoder status word Encoder x-status word:

x: place marker for encoder 1 or 2

-> to indicate statuses, acknowledgements, errors, etc.

Bit	Value	Meaning	Comments
0	1	Functions:	Status: Function 1-4 active (Reference marks search / Flying measurement)
1	1	Reference marks – search	Bit 0: Function 1 (Reference mark 1 / Scanner 1 pos. edge)
2	1		Bit 1: Function 2 (Reference mark 2 / Scanner 1 neg. edge)
3	1	or	Bit 2: Function 3 (Reference mark 3 / Scanner 2 pos. edge)
			Bit 3: Function 4 (Reference mark 4 / Scanner 2 neg. edge)
			Simultaneous setting of Bit 4-7 -> Terminate function 1-4 (device-specific error code in Gx_XIST2)
4	1	Flying measurement	Status: Value 1-4 present (Reference mark / scanner)
5	1		Bit 4: value 1 (Reference mark 1 / scanner 1 pos. edge)
6	1		Bit 5: value 2 (Reference mark 2 / scanner 1 neg. edge)
7	1		Bit 6: value 3 (Reference mark 3 / scanner 2 pos. edge)
			Bit 7: value 4 (Reference mark 4 / scanner 2 neg. edge)
			Simultaneous setting of Bit 0-3 -> Terminate function 1-4 (device-specific error code in Gx_XIST2)
8	1	Scanner 1 deflected	Static status scanner 1 (U923.5)
9	1	Scanner 2 deflected	Static status scanner 2 (U923.6)
10			reserved, set to zero
11			encoder error acknowledgement in processing
12	1	Set/shift reference point executed	Acknowledgement for "Request set / shift reference point" Gx_STW, Bit 11, 12). Consideration at Gx_XIST1, Gx_XIST2
13	1	Transmit absolute value cyclically	Acknowledgement for "Request absolute value cyclically" (Gx_STW, Bit 13). Cyclical transmission of the absolute actual position value in Gx_XIST2.
14	1	Parking encoder active	Acknowledgement for "Activate parking encoder" (Gx_STW, Bit 14).
15	1	Encoder error	Indicates an error of the encoder or the actual value recording. A device-specific error code is in Gx_XIST2. If several errors occur, the first one is displayed.

Table 8.2-18 Encoder status word

Status diagram, statuses and transitions of encoder interface

Status diagram:

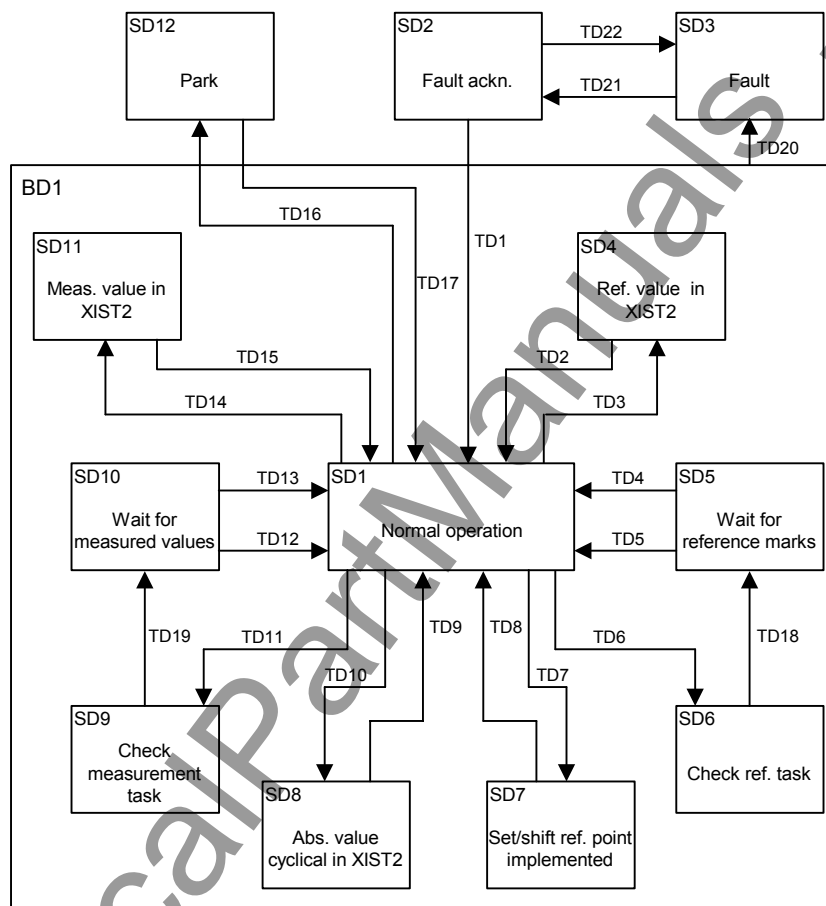


Fig. 8.2-39 Status diagram of encoder interface showing the statuses and transitions

Statuses		Action	Explanation	Code
SD1	Normal operation	None	Encoder interface is working normally	Gx_ZSW-Bit 0-7 = 0000 0000b, Gx_ZSW-Bit 10-15 = 00 0000b
SD2	Error acknowledgement	Error is acknowledged	Error acknowledgement is being processed.	Gx_ZSW-Bit 11 = 1
SD3	Error	Error is on	An error is on.	Gx_ZSW-Bit 15 = 1, Gx_ZSW-Bit 11 = 0
SD4	Reference value in XIST2	Load reference value in XIST2	The reference value in XIST2 is loaded.	Gx_ZSW-Bit 4-7 <> 000b
SD5	Wait for reference marks	Wait for reference marks	The reference mark is expected.	Gx_ZSW-Bit 0-3 <> 0000b
SD6	Check reference task	Check reference task	The reference task is checked.	None
SD7	Set / shift reference point executed	Set or shift reference point	The reference point is either set or shifted.	Gx_ZSW-Bit 12 = 1
SD8	Absolute value cyclical in XIST2	Load absolute value cyclically in XIST2	An absolute value in XIST2 is loaded cyclically.	Gx_ZSW-Bit 13 = 1
SD9	Check measurement command	Check measurement task	The measurement task is checked.	None
SD10	Wait for measured values	Wait for measured value	The measured value is expected.	Gx_ZSW-Bit 0-3 <> 0000b
SD11	Measured value in XIST2	Load measured value in XIST2	The measured value in XIST2 is loaded.	Gx_ZSW-Bit 4-7 <> 000b
SD12	Park	None	The encoder interface is in a condition in which it is not reporting an error and is not participating in the bus.	Gx_ZSW-Bit 14 = 1

Change of status:

	from	to	condition
TD1	SD2 (error acknowledgement)	SD1 (normal operation)	Gx_STW-Bit 15 = 0 and error corrected
TD2	SD4 (Reference value in XIST2)	SD1 normal operation	Gx_STW-Bit 4-6 = 000b
TD3	SD1 (Normal operation)	SD4 (reference value in XIST2)	Gx_STW-Bit 7 = 0 and Gx_STW-Bit 4-6 = 010b and Gx_STW-Bit 0-3 <> 0000b and Ref value_X_found = 1
TD4	SD5 (Wait for reference marks)	SD1 (normal operation)	Gx_STW-Bit 4-6 = 000b and reference marks found
TD5	SD5 (Wait for reference marks)	SD1 (normal operation)	Gx_STW-Bit 4-6 = 011b
TD6	SD1 (Normal operation)	SD6 (check reference task)	Gx_STW-Bit 7 = 0 and Gx_STW-Bit 4-6 = 001b and Gx_STW-Bit 0-3 <> 0000b
TD7	SD1 (Normal operation)	SD7 (set/shift reference point)	Set reference point: Gx_STW-Bit 12 = 1 and Gx_STW-Bit 11 = 0 or Shift reference point: Gx_STW-Bit 12 = 1 and Gx_STW-Bit 11 = 1
TD8	SD7 (Set/shift reference point)	SD1 (normal operation)	Gx_STW-Bit 12 = 0
TD9	SD8 (Absolute value cyclical in XIST2)	SD1 (normal operation)	Gx_STW-Bit 13 = 0
TD10	SD1 (Normal operation)	SD8 (absolute value cyclical in XIST2)	Gx_STW-Bit 13 = 1
TD11	SD1 (Normal operation)	SD9 (check measurement task)	Gx_STW-Bit 7 = 1 and Gx_STW-Bit 4-6 = 001b and Gx_STW-Bit 0-3 <> 0000b
TD12	SD10 (Wait for measurement task)	SD1 (normal operation)	Gx_STW-Bit 4-6 = 011b
TD13	SD10 (Wait for measurement task)	SD1 (normal operation)	Gx_STW-Bit 4-6 = 000b and measured values found
TD14	SD1 (Normal operation)	SD11 (measured value in XIST2)	Gx_STW-Bit 7 = 1 and Gx_STW-Bit 4-6 = 010b and Gx_STW-Bit 0-3 <> 0000b and measurement value_X_found = 1
TD15	SD11 (Measured value in XIST2)	SD1 (normal operation)	Gx_STW-Bit 4-6 = 000b
TD16	SD1 (Normal operation)	SD12 (parking)	Gx_STW-Bit 14 = 1
TD17	SD12 (Parking)	SD1 (normal operation)	Gx_STW-Bit 14 = 0
TD18	SD6 (Check reference task)	SD5 (wait for reference marks)	Task permitted
TD19	SD9 (Check measurement command)	SD10 (wait for measurement command)	Command permitted
TD20	from any status in BD1	SD3 (error)	Error occurred or a command is inadmissible
TD21	SD3 (error)	SD2 (error acknowledgement)	Gx_STW-Bit 15 = 1
TD22	SD2 (error acknowledgement)	SD3 (error)	Gx_STW-Bit 15 = 0 and error still exists

Error code in Gx_IST2:

Gx_XIST2	Meaning	Possible causes / description
1	Encoder sum error	The error description can be taken from the following faults (see appendix: "Faults and Warnings") (cannot be acknowledged via the encoder interface): • Fault F051 Encoder fault See r949 the 100th place for: 0xx: Motor encoder 1xx: External encoder • Fault F054 Encoder board initialization error • Internal error encoder interface • Parking with released absolute value encoder inadmissible
2	Zero mark monitoring	The error description can be taken from the following faults (see appendix: Faults and Warnings): • Fault F051 r949=x26 or r949=x27 Zero pulse Encoder See r949 the 100th place for: 0xx: Motor encoder 1xx: External encoder
4	Terminate reference mark search	• No SBP in the case of the external encoder • Valid measured values lacking (B0070 or B0071) • Acknowledge encoder error active • Parking encoder / axis active • Request absolute value cyclically active • Set/shift reference point active • Mode (BIT 7 = 1) Flying measurement active • or a reserved bit is used
5	Terminate collect reference value	• Valid measured values lacking (B0070 or B0071) • Acknowledge encoder error active • Parking encoder / axis active • Request absolute value cyclically active • Set/shift reference point active • Mode (BIT 7 = 1) Flying measurement active • or a reserved bit is used
6	Terminate flying measurement	• No SBP in the case of the external encoder • Valid measured values lacking (B0070 or B0071) • Acknowledge encoder error active • Parking encoder/axis active • Request absolute value cyclically active • Set/shift reference point active • Mode (BIT7=0) Reference marks search active • or a reserved bit is used

Gx_XIST2	Meaning	Possible causes / description
7	Terminate collect measured value	- Valid measured value lacking (B0070 or B0071) - Acknowledge encoder error active - Parking encoder/axis active - Request absolute value cyclically active - Set/shift reference point active - Mode (BIT7=0) Reference marks search active - or a reserved bit is used
8	Terminate absolute value transmission on	EnDat-encoder (Multiturn) not usable parameter P183 is not set to xxx2 enable position recording with Multiturn.
A	Error in reading absolute track of the absolute value encoder (EnDat-encoder)	Fault F051 Encoder fault SSI/EnDat See r949 x30...x60 0xx: Motor encoder 1xx: External encoder

Table 8.2-19 Error code in Gx_XIST2

Boundary conditions and rules for connection of encoder 1 (Motor encoder)

The following boundary conditions and rules exist:

1. The function "Referencing **only** with rough pulse" is not supported in the standard interconnections. For this measurement via the measured value memory of the basic unit must be used. If the measured value memory is to be used both for referencing and for measurement, corresponding interconnections in the basis unit via free blocks must be used.
2. The function "Referencing **only** with zero mark" must be ensured via corresponding interconnections in the basic unit using B931 "Trigger rough pulse motor encoder".
In this a rough pulse is produced as soon as the release of the reference point recording is switched to enabled. The next zero pulse is then recognized.

Boundary conditions and rules for connection of encoder 2 (External encoder)

1. The functions "Referencing" and "Measured value memory" is **only** supported by the SBP (pulse encoder board) . For this the measured value memory should be used via direct input to the SBP (see function diagram 335).
2. The rough pulse in the external encoder is only evaluated directly on the encoder board (see function diagram 255).
3. The "Referencing" function with zero mark **only** is not supported.

8.2.10 Diagnosis and troubleshooting

NOTE

With regard to basic parameterization, please note the differences in diagnosis and troubleshooting to the types of unit with the older function classes FC (CU1), VC (CU2) and SC (CU3). These differences are described below.

In order to make these differences clear, the parameter numbers and other deviations are either printed in dark gray or have a dark-gray background.

8.2.10.1 Evaluating the possibilities of hardware diagnosis

LED displays

There are three LED displays on the front of the CBP:

- ◆ red: CBP operating
- ◆ yellow: Data exchange with the basic unit
- ◆ green: Transfer of useful data via the PROFIBUS

Diagnostic LEDs give the user rapid information on the status of the CBP at any particular instant.

More detailed diagnostic information can be read out directly from the diagnostics memory of the CBP by means of a diagnostic parameter.

NOTE

During normal operation, all three LEDs light up synchronously and for the same length of time (flashing)!

The stationary status of an LED (on or off) indicates an unusual operating status (parameterization phase or fault)!

LED	Status	Diagnostic information
Red	Flashing	CBP operating; voltage supply on
Yellow	Flashing	Fault-free data exchange with the basic unit
Green	Flashing	Fault-free cyclical useful data traffic with a master, class 1, via PROFIBUS

Table 8.2-20 LED display of the CBP

LED	Status	Diagnostic information
Red	Flashing	No cyclical useful data traffic with a master, class 1, via PROFIBUS –DP due to e.g. EMC interference, bus connector pulled out, polarity reversal of connections, node number not supplied with useful data by the master. Acyclical useful data traffic with a master, class 2 (DriveES, DriveMonitor, SIMATIC OP) does not affect the green LED.
Yellow	Flashing	
Green	Off	

Table 8.2-21 Online operation without useful data

LED	Status	Diagnostic information
Red Yellow Green	Off On On	Voltage supply for CBP cut off; replace CBP or basic unit
Red Yellow Green	On Off On	Data exchange with the basic unit not possible; replace CBP or basic unit
Red Yellow Green	On On Off	No cyclical useful data traffic with a master, class 1, via PROFIBUS is possible; PROFIBUS cable not connected or defective

Table 8.2-22 Fault display CBP

In the following, all exceptional operating conditions are listed which are displayed as such by the CBP.

LED	Status	Diagnostic information
Red Yellow Green	Flashing Off On	CBP is waiting for the basic unit to begin initialization
Red Yellow Green	On Off Flashing	CBP is waiting for the basic unit to complete initialization
Red Yellow Green	Flashing On Off	Checksum error in flash EPROM of the CBP (Download firmware again or replace CBP)
Red Yellow Green	Flashing On On	Error in RAM test of the CBP Replace CBP (external RAM, DPRAM or SPC3-RAM defective)
Red Yellow Green	Flashing Off Off	Only CBP2 DP slave software detects serious fault Note fault number in r732.8 and inform Customer Service

Table 8.2-23 Exceptional operating conditions

LED	Status	Diagnostic information
Red Yellow Green	Off Off Flashing	Only CBP2 USS protocol has been set

Table 8.2-24 USS

8.2.10.2 Fault and alarm display on the basic unit

If faults occur during communication between the PROFIBUS and the CBP, corresponding fault or alarm messages are displayed on the PMU or on the OP of the basic unit.

Alarms

Alarm number		Meaning
First CB/TB	Second CB	
A 081	A 089	The ID byte combinations sent by the DP master in the configuration telegram do not correspond with the permitted ID byte combinations (see table 8.2-12) Consequence: No connection established with the PROFIBUS-DP master; new configuration is necessary.
A 082	A 090	No valid PPO type can be established from the configuration telegram from the DP master. Consequence: No connection established with the PROFIBUS-DP master, new configuration is necessary.
A 083	A 091	No net data or invalid net data (e.g. complete control word STW1=0) are being received by the DP master. Consequence: The process data are not being transferred to the DPR. If parameter P722 (P695) is not equal to zero, this will result in fault message F 082 being tripped (see chapter "Process data monitoring").
A 084	A 092	Telegram traffic between DP master and CBP has been interrupted (e.g. cable break, bus connector disconnected or DP master switched off). Consequence: If parameter P722 (P695) is not equal to zero, this will result in fault message F 082 being tripped (see chapter "Process data monitoring").
A 086	A 094	Failure of heartbeat counter recognized by basic unit. Consequence: Interruption of communication to the automation system
A 087	A 095	DP slave software detects serious fault. Fault number in diagnostic parameter r732.8 Consequence: Communication no longer possible. Secondary fault F082

Alarm number		Meaning
First CB/TB	Second CB	
A 088	A 096	Only CBP2 At least one configured cross-traffic encoder is not yet active or has failed. For details, see CBP2 diagnostic parameters. Consequence: If a encoder is still not active, the relevant setpoints are set to null as a substitute. If a cross-traffic encoder fails, transmission of the setpoints to the basic unit may be interrupted, depending on the setting in P715. Secondary fault F082.

Table 8.2-25 Alarm display on the basic unit

Assignment

The alarm number for the first CB/TB applies to the following configurations:

- ◆ Exactly one CBP has been plugged into slots A to G in the electronics box and no T100/T400 technology board has been plugged in
- ◆ If two CBPs have been plugged in, the alarm number applies to the one which has been plugged into the slot with the lower slot letter.

The alarm number for the second CB applies to the following configurations:

- ◆ One T100/400 technology board has been plugged in and the CBP in the electronics box has been plugged into slots A to C.
- ◆ If two CBPs have been plugged in, the alarm number applies to the one which has been plugged into the slot with the higher letter.

NOTE

The alarm A 082 / A 090 can also be displayed on the basic unit the first time the CBP is started as long as telegrams are not being exchanged with a DP master, e.g. because the bus cable has not yet been connected.

Fault displays

Fault number		Meaning
First CB/TB	Second CB	
F080	F085	Fault in the dual-port RAM Remedy: CBP probably defective, i.e. replace CBP
F081 Fault value (r949) = 0	F081 Fault value (r949) = 2	Fault in the heartbeat counter. The heartbeat counter is no longer being incremented by the CBP due to an internal fault. The CBP is not plugged in correctly or is defective Remedy: Check the connection. If necessary, replace CBP
F082 Fault value (r949) = 1	F082 Fault value (r949) = 2	Telegram failure in the dual-port-RAM (DPR). The telegram failure monitoring time set by means of parameter P722 (P695) has expired (see chapter "Process data monitoring"). The bus has been interrupted or all net data are transferred with 0 (see also A083) Remedy: Check bus cable incl. connecting plug. In the DP-master, assign values not equal to zero to control word STW1.

Table 8.2-26 Fault display on the basic unit

Assignment

The fault number for the first CB/TB applies to the following configurations:

- ◆ Exactly one CBP has been plugged into slots A to G in the electronics box and no T100/T300/T400 technology board has been plugged in.
- ◆ If two CBPs have been plugged in, the fault number applies to the one which has been plugged into the slot with the lower slot letter.

The fault number for the second CB applies to the following configurations:

- ◆ One T100/T300/T400 technology board has been plugged in and the CBP in the electronics box has been plugged into slots A to C
- ◆ If two CBPs have been plugged in, the fault number applies to the one which has been plugged into the slot with the higher letter.

8.2.10.3 Evaluating CBP diagnostic parameters

(For CBP2 diagnostic parameters, see section 8.2.10.6)

NOTE

Please note that, for types of unit with the older function classes FC (CU1), VC (CU2) and SC (CU3), indexed parameter **r731.i** is to be used appropriately instead of **r732.i**.

In order to support start-up and for service purposes, the CBP stores diagnostic information in a diagnostics buffer. The diagnostic information can be read out with the indexed parameter **r732.i** (CB/TB diagnosis).

If two CBPs are inserted in the electronics box, the diagnostic area for the second CBP begins in parameter **r732** from index 33, i.e. an offset of 32 has to be added to the required index number for reading out the diagnostic information of the second CBP.

CBP diagnostic parameters r732

Meaning	Index number	
	1st CBP	2nd CBP
CBP_Status	.1	.33
DP Ctrlr_Status	.2	.34
Global_Controls	.3	.35
Counter: telegrams received without faults (only DP standard)	.4 (Low)	.36 (Low)
Reserved	.4 (High)	.36 (High)
Counter "TIMEOUT"	.5 (Low)	.37 (Low)
Reserved	.5 (High)	.37 (High)
Counter "CLEAR DATA"	.6 (Low)	.38 (Low)
Reserved	.6 (High)	.38 (High)
NOTICE! The following indices have a different meaning if "Extended telegram diagnosis" is selected via P711 P711 / P696 (CB parameter 1).		
Counter: Heartbeat-counter fault	.7 (Low)	.39 (Low)
Reserved	.7 (High)	.39 (High)
Number of bytes for special diagnosis	.8 (Low)	.40 (Low)
Reserved	.8 (High)	.40 (High)
Mirroring slot Identifier 2	.9 (Low)	.41 (Low)
Mirroring slot Identifier 3	.9 (High)	.41 (High)
Mirroring P918 (CB bus address), only low part	.10 (Low)	.42 (Low)
Reserved	.10 (High)	.42 (High)
Counter re-configuration by CU	.11 (Low)	.43 (Low)
Counter initializations	.11 (High)	.43 (High)
Fault detection DPS manager fault (8 bits)	.12 (Low)	.44 (Low)

Meaning	Index number	
	1st CBP	2nd CBP
Reserved	.12 (High)	.44 (High)
Determined PPO type (8 bits)	.13 (Low)	.45 (Low)
Reserved	.13 (High)	.45 (High)
Mirroring "DWORD-Specifier-ref"	.14	.46
Mirroring "DWORD-Specifier-act"	.15	.47
Counter DPV1:DS_WRITE, positive acknowledgement	.16 (Low)	.48 (Low)
Reserved	.16 (High)	.48 (High)
Counter DPV1: DS_WRITE, negative acknowledgement	.17 (Low)	.49 (Low)
Reserved	.17 (High)	.49 (High)
Counter DPV1:DS_READ, positive acknowledgement	.18 (Low)	.50 (Low)
Reserved	.18 (High)	.50 (High)
Counter DPV1:DS_READ, negative acknowledgement	.19 (Low)	.51 (Low)
Reserved	.19 (High)	.51 (High)
Counter DP/T: GET DB99, positive acknowledgement	.20 (Low)	.52 (Low)
Counter DP/T: PUT DB99, positive acknowledgement	.20 (High)	.52 (High)
Counter DP/T: GET DB100, positive acknowledgement	.21 (Low)	.53 (Low)
Counter DP/T: PUT DB100, positive acknowledgement	.21 (High)	.53 (High)
Counter DP/T: GET DB101, positive acknowledgement	.22 (Low)	.54 (Low)
Counter DP/T: PUT DB101, positive acknowledgement	.22 (High)	.54 (High)
Counter DP/T-service negative acknowledgement	.23 (Low)	.55 (Low)
Counter DP/T: application relation, positive acknowledgement	.23 (High)	.55 (High)
Reserved	.24	.56
Generating date: day, month	.25	.57
Generating date: year	.26	.58
Software version	.27	.59
Software version	.28	.60
Software version: flash EPROM checksum	.29	.61
Reserved	:	
Reserved	.32	.64

Table 8.2-27 CBP diagnostics buffer

8.2.10.4 Meaning of information in the CBP diagnosis parameter r723

(For CBP2 diagnosis, see section 8.2.10.6)

r732.1

(090H, CBP_Status)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

- ◆ Bit 0
"CBP Init": CBP is presently being initialized or is waiting for initialization from the basic board
(normal operation: not set)
- ◆ Bit 1
"CBP Online": CBP selected via board mounting position 2" (DPRAM Offset Address 0x54) or via board mounting position 3" (DPRAM Offset Address 0x55) by the basic board
(normal operation: set)
- ◆ Bit 2
"CBP Offline": CBP selected neither via board mounting position 2" (DPRAM Offset Address 0x54) nor via board mounting position 3" (DPRAM Offset Address 0x55) by the basic board
(normal operation: not set)
- ◆ Bit 3
Value range exceeded "CB bus address" (P918) (basic board).
(normal operation: not set)
- ◆ Bit 4
Extended diagnostic activated [CB parameter 1 (P711 / P696) <> 0].
(normal operation: not set)
- ◆ Bit 8
Incorrect identification byte transferred (faulty configuration telegram from the PROFIBUS DP master).
(normal operation: not set)
- ◆ Bit 9
Incorrect PPO type (faulty configuration telegram from the PROFIBUS DP master).
(normal operation: not set).
- ◆ Bit 10 (not used for CBP2)
Correct configuration received from the PROFIBUS DP master
(normal operation: set).
- ◆ Bit 12
Fatal error detected by the DPS manager SW
(normal operation: not set)
- ◆ Bit 13
Program on the CBP is processed cyclically (is only escaped from if reset is made)
(normal operation: set).
- ◆ Bit 15
Program on the CBP in "Communications online" loop (is only escaped from if initialization is carried out by the basic board)

**r732.2 (092H,
DP Ctrler_Status)**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

- ◆ Bit 0 Offline/Passive Idle
0 = DP Ctrler is offline (normal operation)
1 = DP Ctrler is in passive-idle
- ◆ Bit 1 Reserved
- ◆ Bit 2 Diag-Flag
0 = Diagnostics buffer collected by the master
1 = Diagnostics buffer not collected by master
- ◆ Bit 3 RAM Access Violation, memory accessed > 1.5kByte
0 = No address violation (normal operation)
1 = With addresses >1536 bytes, retreat made from
respective address 1024 and access is made
under this new address
- ◆ Bits 4,5 DP-State 1..0
00 = Status "Wait_Prm"
01 = Status "Wait_Cfg"
10 = Status "DATA_Exchg"
11 = Not possible
- ◆ Bits 6,7 WD-State 1..0
00 = Status "Baud_Search"
01 = Status "Baud_Control"
10 = Status "DP_Control"
11 = Not possible from PROFIBUS DP master
- ◆ Bits 8,9,10,11 Baud rate 3..0
0000 = 12 mbaud
0001 = 6 mbaud
0010 = 3 mbaud
0011 = 1.5 mbaud
0100 = 500 kbaud
0101 = 187.5 kbaud
0110 = 93.75 kbaud
0111 = 45.45 kbaud
1000 = 19.2 kbaud
1001 = 9.6 kbaud
Rest = Not possible
- ◆ Bits 12,13,14,15 SPC3-Release 3..0:
0000 = Release 0
Rest = Not possible
DPC31:
0000 = Step A
0001 = Step B
0010 = Step C

**r732.3 (094H,
Global_Controls)**

Bits remain set until the next DP global control.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

- ◆ Bit 0 Reserved
- ◆ Bit 1 1 = Clear_Data telegram received
- ◆ Bit 2 1 = Unfreeze telegram received
- ◆ Bit 3 1 = Freeze telegram received
- ◆ Bit 4 1 = Unsync telegram received
- ◆ Bit 5 1 = Sync telegram received
- ◆ Bits 6,7 Reserved

**r732.4
(Low-Byte), 096H**

Counter for telegrams received error-free (only DP standard)

Counter for received DP net telegrams

**r732.5
(Low-Byte), 098H**

Counter TIMEOUT

Counter is incremented if the "TIMEOUT" signal is identified. This occurs if, for example, the bus connector is pulled out when response-monitoring has been activated (at the DP master).

**r732.6
(Low-Byte), 09AH**

Counter CLEAR DATA

Is incremented if the global ctrl. signal "CLEAR DATA" is identified (see also r732.3). This occurs if, for example, the DP master is set in "STOP".

**r732.7
(Low-Byte), 09CH**

Counter Heartbeat-Counter Error

Is incremented if the heartbeat-counter is not changed by the basic board or the technology board within approx. 800 ms.

**r732.8
(Low-Byte), 09EH**

Number of bytes during special diagnosis

Number of bytes entered after r732.9 during special diagnosis selected by means of CB parameter 1.

**r732.9
(Low-Byte), 0A0H**

Mirroring slot Identifier 2

Read out of the DPRAM during run up: Offset Address 054H, with VC,FC and SC, corresponds to parameter P090.

**r732.9
(High-Byte), 0A1H**

Mirroring slot Identifier 3

Read out of the DPRAM during run up: Offset Address 055H, with VC,FC and SC, corresponds to parameter P091.

**r732.10
(Low-Byte), 0A2H**

Mirroring P918

Read out of the DPRAM during run up: "CB Bus address" (only Low-byte)

**r732.11
(Low-Byte), 0A4H**

Counter Re-configuration by CU

Re-configuration requested by the basic board in online mode

**r732.11
(High-Byte), 0A5H**

Counter Initialization

Is incremented during run through of the initialization routine

**r732.12
(Low Byte), 0A6H**

DPS Manager Error

Error detection in the event of a fatal DPS manager error

**r732.13
(Low-Byte), 0A8H**

PPO type

PPO type detected from configuration telegram

**r732.13
(High-Byte), 0A9H**

Reserved

**r732.14,
0AAH u. 0ABH**

Mirroring "DWORD-Specifier-ref"

Read out of the DPRAM during run up: updated cyclically

**r732.15,
0ACH u. 0ADH**

Mirroring "DWORD-Specifier-act"

Read out of the DPRAM during run up: updated cyclically

r732.16 (Low-Byte), 0AEH	Counter DS_WRITE acknowledgement negatively
r732.16 (High-Byte), 0AFH	Reserved
r732.17 (Low-Byte), 0B0H	Counter DS_WRITE acknowledged positively
r732.17 (High-Byte), 0B1H	Reserved
r732.18 (Low-Byte), 0B2H	Counter DS_READ acknowledged negatively
r732.18 (High-Byte), 0B3H	Reserved
r732.19 (Low-Byte), 0B4H	Counter DS_READ acknowledged positively
r732.19 (High-Byte), 0B5H	reserved
r732.20 (Low-Byte), 0B6H	Counter GET DB99 acknowledged positively
r732.20 (High-Byte), 0B7H	Counter PUT DB99 acknowledged positively
r732.21 (Low-Byte), 0B8H	Counter GET DB100 acknowledged positively
r732.21 (High-Byte), 0B9H	Counter PUT DB100 acknowledged positively
r732.22 (Low-Byte), 0BAH	Counter GET DB101 acknowledged positively
r732.22 (High-Byte), 0BBH	Counter PUT DB101 acknowledged positively
r732.23 (Low-Byte), 0BCH	Counter DPT-Service acknowledged negatively
r732.23 (High-Byte), 0BDH	Counter Applic. acknowledged positively Increment during DPT service "Set-up application relation"
r732.24 (Low-Byte), 0BEH	reserved
r732.24 (High-Byte), 0BFH	reserved
r732.25 0C0H and 0C1H	Creation date Day and month when CBP firmware created (Display: 0304 = 03.04.)
r732.26 0C2H and 0C3H	Creation data Year when CBP firmware created (Display = Year)
r732.27 0C4H and 0C5H	Software-Version Software version V X.YZ (Display X)
r732.28 0C6H and 0C7H	Software-Version Software version V X.YZ (Display YZ)
r732.29 0C8H and 0C9H	Flash-EPROM Checksum Is read out of the flash EPROM during run-up

8.2.10.5 Additional methods of diagnosis for start-up personnel

(See section 8.2.10.7 for extended CBP2 diagnosis)

NOTE

The CB parameters, P711 to P721, have two indices. The following convention applies to this:

Index 1 is valid for the first CBP

Index 2 is valid for the second CBP

In order to determine which CBP is the first and which the second, see Section 8.2.5 "Mounting methods / CBP slots".

CB parameter 1 Telegram diagnosis

With P711 / P696 (CB parameter 1), special diagnostic entries for the CBP diagnostics buffer can be selected. If P711 / P696 is set to a value not equal to zero during parameterization of the CBP by the converter, telegram contents of the PROFIBUS-DP telegram are cyclically entered into the CBP diagnostics buffer, depending on the set value.

The entries are made in rising sequence beginning with r732.9 (r732.10, r732.11 etc.) in the same way as the corresponding useful data are transferred via the PROFIBUS-DP, namely high-byte before low-byte, high-word before low-word. The original entries (i.e. when P711 / P696 = "0") are overwritten, beginning with r732.9.

Entries r732.1 to 732.8 retain their meaning.

Detailed knowledge of PROFIBUS-DP telegrams is needed in order to evaluate these diagnostic entries.

It is only possible to set parameter P711 / P696 when the "Hardware Configuration" function is being selected (P060 or. P052).

NOTE

Parameter P711 / P696 is only to be set to a value other than zero for diagnostic purposes because permanent transfer of diagnostic information to the DPRAM reduces the data throughput rate of the CBP!

The original entries in parameter r732 / r731 are overwritten, beginning with r732.9 / r731.9.

PMU:

P711 / P696 = 0 Telegram diagnosis = Off

P711 / P696 = 1 to 26 Telegram diagnosis = ON

Telegram entries

P711 P696	= 0	No supplementary diagnosis (default setting)		
The following entries apply to cyclical data transfer via MSZY-C1				
P711 P696	= 1	PPO useful data in the CBP receive buffer	Useful-data telegram (master → converter)	Length depends on PPO type
P711 P696	= 2	PPO useful data in the CBP transmit buffer	Useful-data telegram (converter → master)	Length depends on PPO type
P711 P696	= 3	Configuration buffer	Useful-data telegram (master → converter)	Length = 25 bytes
P711 P696	= 4	Parameterization buffer	Parameterization telegram (master → converter)	Length = 10 bytes
The following entries apply to cyclical data transfer via MSAC-C1				
P711 P696	= 10	Useful data of the DS100	Data unit in DS_WRITE to DS100	Max. 32 bytes
P711 P696	= 11	Useful data of the DS100	Data unit in DS_READ to DS100	Max. 32 bytes
The following entries apply to acyclical data transfer via MSAC-C2				
P711 P696	= 21	Useful data in the DB99	Data unit in PUT to the DB99	Max. 32 bytes
P711 P696	= 22	Useful data in the DB99	Data unit in GET to DB99	Max. 32 bytes
P711 P696	= 23	Useful data in the DB100	Data unit in PUT to DB100	Max. 32 bytes
P711 P696	= 24	Useful data in the DB100	Data unit in GET to DB100	Max. 32 bytes
P711 P696	= 25	Useful data in the DB101	Data unit in PUT to DB101	Max. 32 bytes
P711 P696	= 26	Useful data in the DB101	Data unit in GET to DB101	Max. 32 bytes

Table 8.2-28 Selection of PROFIBUS-DP telegram entries

Example 1

Parameter P711 / P696 = 1

The useful data (PPO) received from the DP master via the cyclical standard channel MSCY_C1 are entered in the diagnostics buffer.

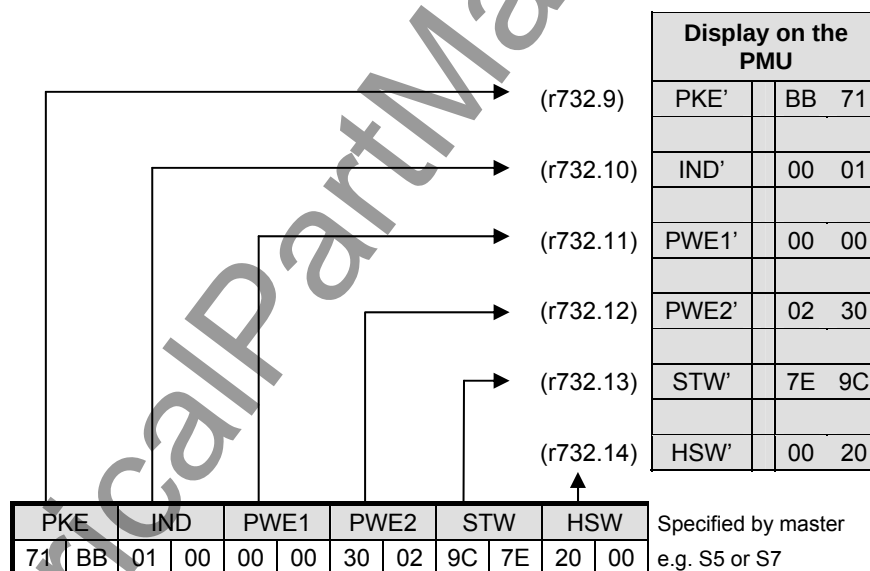
PPO type = 1

Four words, PKW part plus control word 1 (STW1) and the main setpoint (HSW), are received. The PKW part is placed, beginning with the PKE, in parameter r732.9; STW1 and also the HSW are placed from parameter r732.13 onwards (high part at the least significant address).

In the following example, a WRITE request from the DP master is shown with the value "3002" in parameter P443.

The control word is specified with 9C7E_{Hex} in the DP master and 2000_{Hex} is specified as the setpoint.

The values in r732 are displayed in Motorola format, i.e. high-byte and low-byte are shown interchanged in relation to what is displayed in the other parameters.

**Visualization parameter r733**

In order to visualize the received **process data** (PZD), parameter r733 can also be used. In parameter r733, all process data are displayed normally, i.e. in Intel format, in the same way as they are used in the MASTERDRIVES.

The PKW interface cannot be visualized by means of parameter r738 and r739.

The index ranges used in parameters r733, r738 and r739 are shown in the function diagrams in the appendix.

NOTE

In the examples and in the following tables, information with an apostrophe (e.g. PKE') means that, with these values, the high-byte and the low-byte are swapped round in relation to the original value, as for example in the programmable controller.

Example 2

Parameter P711 / P696 = 2

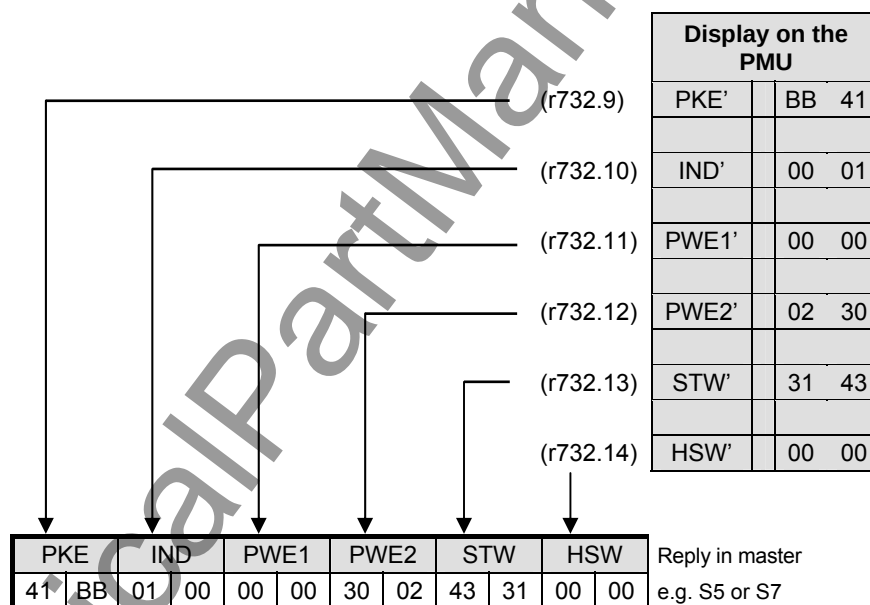
The useful data (PPO) sent to the DP master via the standard cyclical channel MSCY_C1 are entered into the diagnostics buffer.

PPO-type = 1

Four words are sent, PKW part plus status word 1 (ZSW1) and the main actual value (HIW). The PKW part is stored, beginning with the PKE, in parameter r732.9 and ZSW1 as well as the HIW from parameter r732.13 onwards (high part at the least significant address).

In the following example, the reply (to the DP master) to the WRITE request in example 1 is shown in parameter P443 with the value "3002". The status word is returned by the converter with 4331_{Hex}; 0000_{Hex} is given as the actual value.

The values in r732 are displayed in Motorola format, i.e. high-byte and low-byte are shown interchanged in relation to what is displayed in the other parameters.



**Telegram contents
(communication
with Master 1)**

Display in r732	When P711 = 1 or 2		When P711 = 3	When P711 = 4	When P711 = 10	When P711 = 11
	PPOs 1,2, or 5	PPOs 3 or 4	Different depending on PPO	Paramet- eriz. telegram		
ii 09	PKE'	PZD1'	00 04	Byte 2 u 1	PKE'	PKE'
ii 10	IND'	PZD2'	AD 00	Byte 4 u 3	IND'' 2)	IND'' 2)
ii 11	PWE1'	PZD3' *	04 C4	Ident-No.	PWE1'	PWE1'
ii 12	PWE2'	PZD4' *	00 00	Byte 8 u 7	PWE2'	PWE2'
ii 13	PZD1'	PZD5' *	40 BB	Byte 10 u 9	PWE3'	PWE3'
ii 14	PZD2'	PZD6' *	00 04	xxx	PWE4'	PWE4'
ii 15	PZD3' *	xxx	8F 00	xxx	PWE5'	PWE5'
ii 16	PZD4' *	xxx	C2 C0	xxx	PWE6'	PWE6'
ii 17	PZD5' *	xxx	per PPO	xxx	PWE7'	PWE7'
ii 18	PZD6' *	xxx	per PPO	xxx	PWE8'	PWE8'
ii 19	PZD7' **	xxx	per PPO	xxx	PWE9'	PWE9'
ii 20	PZD8' **	xxx	per PPO	xxx	PWE10'	PWE10'
ii 21	PZD9' **	xxx	per PPO	xxx	PWE11'	PWE11'
ii 22	PZD10' **	xxx	1)	xxx	PWE12'	PWE12'
ii 23	xxx	xxx	xxx	xxx	PWE13'	PWE13'
ii 24	xxx	xxx	xxx	xxx	PWE14'	PWE14'

1) The 25 bytes with slot-oriented S7 type identifications are always entered, even if the CBP is configured with identification bytes by an SIMATIC S5 or a non-Siemens master.

2) As regards IND', high-byte and low-byte are inverted in relation to the IND': this is based on a different definition of the useful data for PPOs and acyclically transferred sets of data.

* only for PPO2 and 4

** only for PPO5

Structure and content of the parameterization telegram									
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10
DP- Statu s	WD_ Fac 1	WD_ Fac 2	TSDR - min	PNO-Ident- No.		Grou- p- Ident	DPV1 - Statu s 1	DPV1 - Statu s 2	DPV1 - Statu s 3

Table 8.2-29 Telegram contents in parameter r732i09 which can be read out (communication with Master 1)

**Telegram contents
(communication
with DriveMonitor)**

Display in r732	When P711 = 21	When P711 = 22	When P711 = 23	When P711 = 24	When P711 = 25	When P711 = 26
ii 09	PZD rights	PZD rights	PKE'	PKE'	PZD1'	PZD1'
ii 10	xxx	xxx	IND''	IND''	PZD2'	PZD2'
ii 11	xxx	xxx	PWE1'	PWE1'	PZD3'	PZD3'
ii 12	xxx	xxx	PWE2'	PWE2'	PZD4'	PZD4'
ii 13	xxx	xxx	PWE3'	PWE3'	PZD5'	PZD5'
ii 14	xxx	xxx	PWE4'	PWE4'	PZD6'	PZD6'
ii 15	xxx	xxx	PWE5'	PWE5'	PZD7'	PZD7'
ii 16	xxx	xxx	PWE6'	PWE6'	PZD8'	PZD8'
ii 17	xxx	xxx	PWE7'	PWE7'	PZD9'	PZD9'
ii 18	xxx	xxx	PWE8'	PWE8'	PZD10'	PZD10'
ii 19	xxx	xxx	PWE9'	PWE9'	PZD11'	PZD11'
ii 20	xxx	xxx	PWE10'	PWE10'	PZD12'	PZD12'
ii 21	xxx	xxx	PWE11'	PWE11'	PZD13'	PZD13'
ii 22	xxx	xxx	PWE12'	PWE12'	PZD14'	PZD14'
ii 23	xxx	xxx	PWE13'	PWE13'	PZD15'	PZD15'
ii 24	xxx	xxx	PWE14'	PWE14'	PZD16'	PZD16'

Table 8.2-30 Telegram contents in parameter r732i09 which can be read out
(communication with DriveMonitor)

**CB parameter 3
(DPRAM monitor)**

By means of CB parameter 3, i.e. P713 / P698, a hex monitor can be activated with which addresses of the dual-port RAM can be read out on the CBP.

DANGER


Parameter P713 / P698 is to be reserved exclusively for suitably trained start-up personnel.

In order to use the hex monitor to best effect, appropriate detailed knowledge of the structure of the dual-port RAM is necessary. In P713 / P698, only the offset address (decimal) is entered.

If CB Parameter 3 is set to a value other than "0", 12 bytes are cyclically entered in diagnostic parameter r732 from r732.9 onwards. This is done from the absolute address set in CB parameter 3 (decimal) onwards. CB Parameter 3 has the highest priority and disables entries by CB parameter 1.

**Diagnosis with
PROFIBUS Class II
Master**

A Class II master (normally a PG programming unit) can be used for start-up and diagnosis. During start-up/testing, the Class II master assumes the function of the Class I master for the selected station. The exchange of useful data with the slave, however, is not cyclical.

8.2.10.6 CBP2 diagnostic parameters

Meaning of standard diagnosis with P711.x = 0

Parameter No.	Content (high byte)	Content (low byte)
r732.1	CBP2 status (same content as CBP)	
r732.2	DPC31 status (same content as CBP, SPC3 status)	
r732.3	Global control (same content as CBP)	
r732.4	Counter: CLEAR DATA (alteration if, e.g. DP-Master in "Stop")	Counter: fault-free cyclical telegrams
r732.5	Counter: Heartbeat counter fault from basic unit	Counter: Watchdog state changed (alteration during plugging/unplugging of connector or C1 master is coming/going)
r732.6	Mirroring: Slot identifier 3	Mirroring: Slot identifier 2
r732.7	PNO identification (0x8045)	
r732.8	Number of valid bytes in r732.9 to r732.24 when P711.x > 0 (special diagnosis) or: fault number DP slave software for alarm A087	
	NOTICE! The following indices have a different meaning if "Extended telegram diagnosis" is selected via P711 / P696 (CB parameter 1).	
r732.9	Cross traffic: address encoder 1	Encoder 2
r732.10	Encoder 3	Encoder 4
r732.11	Encoder 5	Encoder 6
r732.12	Encoder 7	Cross traffic: address encoder 8
r732.13	CBP2 itself works as a cross-traffic encoder	PPO type (0xFF: no PPO)
r732.14	Cross traffic: number of configured encoders	Cross traffic: Score Board, one bit per encoder (Bit 0 = Encoder 1, ... Bit 7=Encoder 8) 0: Encoder inactive 1: Encoder configured and active
r732.15	Counter: repeated cyclical PKW request	Counter: new cyclical PKW task
r732.16	Counter: C1 DS100 Write/Read negative	Counter: C1 DS100 Write/Read positive
r732.17	Counter: DriveES Write/Read negative	Counter: DriveES Write/Read positive
r732.18	Counter: DriveES Control negative	Counter: DriveES Control positive
r732.19	Counter: DriveES Setpoints negative	Counter: DriveES Setpoints positive
r732.20	Counter: S7 Protocol negative	Counter: S7 Protocol positive
r732.21	Counter: Abort C2 master	Counter: Initiate C2 master
r732.22	S7 protocol access fault: For fault number, see following table	
r732.23	S7 protocol access fault: Data block number or parameter number	
r732.24	S7 protocol access fault: Data block offset or index word	
r732.25	Generating date: Day	Generating date: Month
r732.26	Generating date: Year	
r732.27	Software version	
r732.28	Software version	
r732.29	Software version: Flash-EPROM checksum	

Fault S7 protocol (r732.22), fault numbers < 150 correspond to PKW fault numbers:

No.	Cause	Remedy (e.g. in ProTool)
	No. 0 .. 199: S7 task has been changed into a parameter task. Fault detection in the basic board or technology board. Additional info in r732.23, r732.24: parameter number, index word.	
0	There is no parameter number	Check data block number
1	Parameter value cannot be altered	-
2	Top or bottom limit exceeded	-
3	There is no subindex	Check data block offset
4	Access to single value with array identifier	Set data block offset = 0
5	Access to word with double word task or vice versa	Use correct type of data (e.g. INT for word, DINT for double word)
6	Setting not allowed (can only be reset)	-
7	Description element cannot be altered	(should not occur here)
11	No parameter change rights	-
12	Keyword missing	-
15	There is no text array	-
17	Task cannot be executed due to operating status	-
101	Parameter number deactivated at the moment	-
102	Channel width too small	(should not occur here)
103	PKW number incorrect	(should not occur here)
104	Parameter value not permissible	-
105	Access to array parameter with single identifier	Set data block offset > 0
106	Task not implemented	-
	No. 200-209: S7 task is formally defective. Error detection in the COM BOARD. Additional info in r732.23, r732.24: data block number, data block offset	
200	Error in variables address (no additional info)	Permissible: range of "Data block"
201	Data block number not permissible	Permissible: 1...31999
202	Data block offset not permissible	Permissible: 0...116, 10001...10116, 20000...20010
203	Non-permissible "Type" during access to parameter value	Permissible: CHAR, BYTE, INT, WORD, DINT, DWORD, REAL
204	Non-permissible "Number of elements" during access to parameter value	Permissible: effective 2 or 4 byte
205	Non-permissible "Type" during access to text	Permissible: CHAR, BYTE
206	Non-permissible "Type" during access to description	Permissible: CHAR, BYTE, INT, WORD, DINT, DWORD, REAL
207	Non-permissible odd "Number of elements" in the case of type CHAR or BYTE	Correct the "Number of elements"
208	Non-permissible change of text/description	-
209	Inconsistency in the write task: "Type" and "Number of elements" does not match "Type of data" and "Length of data"	(Defective communications partner)

No.	Cause	Remedy (e.g. in ProTool)
	No. 220: S7 task has been changed into a parameter task. Reply from the basic board or the technology board is faulty. Fault detection in the CBP. Additional info in r732.23, r732.24: data block number, data block offset.	
220	Parameter reply does not match task	(Defective basic board or technology board)
	No. 240: Fault detection in the CBP; without additional info	
240	Reply too long for reply telegram	(Defective communications partner)

Diagnosis of clock synchronization with "SIMOLINK" diagnostic parameter r748 (MASTERDRIVES MC only):

r748.x	(Content of SIMOLINK SLB)	Content of PROFIBUS CBP2
r748.1	Number of error-free synchronizing telegrams	
r748.2	CRC error	Internal
r748.3	Number of timeout errors	Internal
r748.4	Last bus address signaled	Internal
r748.5	Address of the node which sends the special telegram "Timeout"	Internal
r748.6	Active SYNC-interrupt delay	Internal
r748.7	Position of the node in the ring	Internal (deviation of pulse period, configured on CU and set via PROFIBUS)
r748.8	Number of nodes in the ring	Maximum permissible deviation of the pulse period
r748.9	Synchronism deviation (65535: Synchronization not active) should fluctuate between 65515 and 20	
r748.10	Corrected pulse period in units of 100 ns	
r748.11	T0 counter (0 if synchronization active)	Internal
r748.12	Internal	Internal
r748.13	Internal	Internal
r748.14	Timer	Internal
r748.15	Bus cycle time implemented	
r748.16	Internal	Internal

8.2.10.7 Extended CBP2 diagnosis for start-up personnel

Activation of the extended diagnosis with P711.x > 0

Image of the C1 master telegrams

P711.x	Display in r732.9..24 (32 bytes)	
1	Output: PKW and setpoints from the master	Maximum 32 bytes
2	Input: PKW and actual values to the master	Maximum 32 bytes
3	Configuring telegram from the master	Byte 0 – 31
50	End identifier: 0x5A, 0xA5	Byte 32 - 63
51		Byte 64 - 95
52		Byte 96 - 127
53		Byte 128 - 159
54		Byte 160 - 191
55		Byte 192 - 223
56		Byte 224 - 244
4	Parameterizing telegram from the master	Byte 0 – 31
60	End identifier: 0x5A, 0xA5	Byte 32 - 63
61		Byte 64 - 95
62		Byte 96 - 127
63		Byte 128 - 159
64		Byte 160 - 191
65		Byte 192 - 223
66		Byte 224 - 244

Diagnosis of configuration and parameterization

P711.x	r732.x	
30	r732.9	Result of parameterizing telegram evaluation (see table)
	r732.10	Result of evaluating cross-traffic parameterization (see table)
	r732.11	Result of configuring telegram evaluation (see table)
	r732.12	PPO type 1-5; if free configuration, then 0xff
	r732.13	Length of the input data to the master (without PKW) in bytes
	r732.14	Length of the output data from the master (without PKW) in bytes
	r732.15	Double-word specifier setpoints
	r732.16	Double-word specifier actual values
	r732.17	Free memory in the multi-port RAM of the DPC31 in bytes

The value output in parameter P732.9 (P711.x = 30) arises due to bit-by-bit OR linking of the following parameters. In the case of errors in the block for cross-traffic parameterization, the detailed fault codes are to be entered in parameter P732.10. Only if P732.10 contains the value 0 can the clear causes of the fault be read out of P732.9. If P732.10 <> 0, the content of P732.9 is falsified and the errors leading to abort cannot be clearly determined!

Value	Meaning
0x0000	Parameterizing telegram is error free
0x0001	Unknown master, length of para. telegram <10 and >7
0x0002	Unknown para. block. The following are supported: 0xE1 – Equidistance, 0xE2 – Cross traffic ¹⁾
0x0004	It was not possible to fully identify the para. telegram
0x0008	It was not possible to set up the parameter buffer in the DPC31. (Memory size insufficient!)
0x0010	The block for equidistance parameterization has an incorrect length (24 + 4 = 28 bytes)
0x0020	The CU has not opened the RCC channel (no CU SW-version with equidistance capability) or cannot process the RCC channel
0x0040	Non-permissible parameter (e.g. bus cycle time and pulse frequency do not correlate)
0x0080	Tbase-dp is larger than 16 bits after de-normalization
0x0100	Tdp is larger than 16 bits
0x0200	Tdx is larger than Tdp
0x0400	The free computing time is not sufficient. (Tdp-Tdx is too small)
0x0800	The para. telegram contains an invalid value for Isochronous Mode Supported (permissible values 0xE1 ¹⁾)
0x1000	Unknown equidistance mode set by the basic board

¹⁾ From CBP2 V2.21, 0xE1, 0xE2 with DriveES Slave OM; 0x04 with GSD R4
are applicable

Table 8.2-31 Parameter-telegram evaluation r732.9 / P711 = 30

Value	Meaning
0x0000	Parameterizing block "cross-traffic" error-free
0x1001	Default return value
0x1002	The version of the filter table is not supported. Identifier 0xE2 is supported.
0x1004	The data area of the CBP2 (16 word PZD) is exceeded.
0x1008	The pick-off has an odd number of bytes. Only word-by-word pick-offs are permitted.
0x1010	The maximum number of pick-offs has been exceeded. (A maximum of 8 pick-offs are allowed, including pick-off of own data)
0x1020	No links have been configured in the "cross-traffic" parameterizing block
0x1040	A pick-off does not indicate the beginning of a process data word
0x1080	The permissible telegram length which is to be read has been exceeded (maximum 244 bytes).
0x1100	The reserved memory area in the multi-Port RAM has been exceeded.
0x1200	Non-permissible publisher address 1-125
0x1400	Several links to a publisher are not permissible.

Table 8.2-32 Parameter-telegram evaluation, cross-traffic, r732.10 / P711 = 30

Diagnosis of the setpoint source (especially during cross traffic)

P711.x	r732.x	Content	High byte	Low byte
31	r732.9	Setpoint source:	Setpoint 2	Setpoint 1
	P732.10	0: Master	Setpoint 4	Setpoint 3
	P732.11	1 to 8: cross-traffic encoder	Setpoint 6	Setpoint 5
	P732.12	9: -	Setpoint 8	Setpoint 7
	P732.13		Setpoint 10	Setpoint 9
	P732.14		Setpoint 12	Setpoint 11
	P732.15		Setpoint 14	Setpoint 13
	P732.16		Setpoint 16	Setpoint 15
	P732.17	Byte offset of the setpoint within the setpoint source (value range 0 to 30)	Setpoint 2	Setpoint 1
	P732.18		Setpoint 4	Setpoint 3
	P732.19		Setpoint 6	Setpoint 5
	P732.20		Setpoint 8	Setpoint 7
	P732.21		Setpoint 10	Setpoint 9
	P732.22		Setpoint 12	Setpoint 11
	P732.23		Setpoint 14	Setpoint 13
	P732.24		Setpoint 16	Setpoint 15

Diagnosis of clock synchronization

P711.x	r732.x	Content
32	r732.9	Interrupt enable by the basic board
	r732.10	RCC parameter 1
	r732.11	RCC parameter 2
	r732.12	Synchronization mode from the basic board

8.2.11 Appendix

Technical data

Order number	CBP: 6SE7090-0XX84-0FF0 CBP2: 6SE7090-0XX84-0FF5
Size (length x width)	90 mm x 83 mm
Degree of pollution	Degree of pollution 2 acc. to IEC 664-1 (DIN VDE 0110/T1), Moisture condensation during operation is not permissible
Mechanical strength	To DIN IEC 68-2-6 (if board is correctly mounted)
In stationary use	
• displacement	0.15 mm in the frequency range 10 Hz to 58 Hz
• acceleration	19.6 m/s ² in the frequency range > 58 Hz to 500 Hz
During transport	
• displacement	3.5 mm in the frequency range 5 Hz to 9 Hz
• acceleration	9.8 m/s ² in the frequency range > 9 Hz to 500 Hz
Climatic class	Class 3K3 to DIN IEC 721-3-3 (during operation)
Method of cooling	Natural air cooling
Permissible ambient or coolant temperature	
• during operation	0° C to +70° C (32° F to 158° F)
• during storage	-25° C to +70° C (-13° F to 158° F)
• during transport	-25° C to +70° C (-13° F to 158° F)
Permissible moisture stress	Relative humidity ≤ 95 % during transport and storage ≤ 85 % during operation (condensation not permissible)
Supply voltage	5 V ± 5 %, max. 600 mA, internally from the basic unit
Output voltage	5 V ± 10 %, max. 100 mA, electrically isolated supply (X448/Pin 6)
	• for bus termination of the serial interface or
	• for supplying an OLP (Optical Link Plug)
Data transfer rate	max. 12 MBaud

Table 8.2-33 Technical data

Block diagram of the CBP

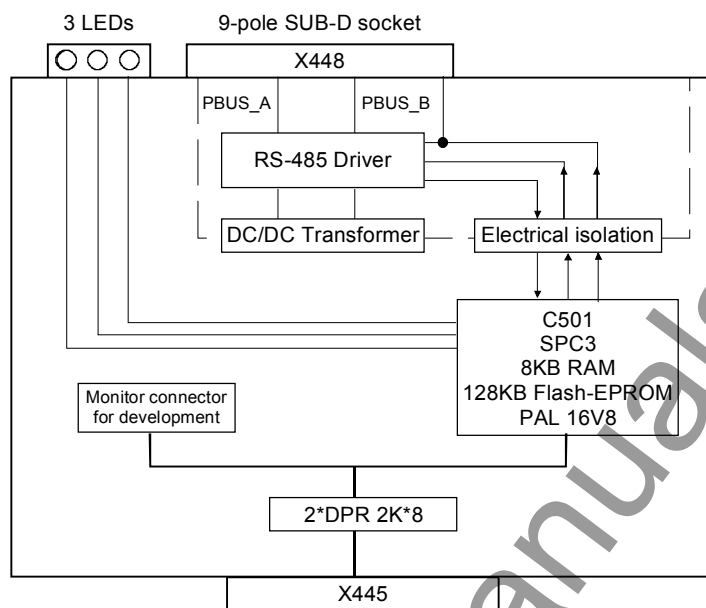


Fig. 8.2-40 Block diagram of the CBP

8.3 SIMOLINK

8.3.1 General principles

Definition

SIMOLINK (Siemens Motion Link) is a digital, serial data transfer protocol with a fiber-optic cable as its transfer medium. The SIMOLINK drive link has been developed for extremely fast and strictly cyclical transfer of process data (control information, setpoints, actual values and status information) between individual MASTERDRIVES MC/VC units or between MASTERDRIVES MC/VC units and a higher-level control system with synchronization of all connected nodes to a common system clock.

Application

SIMOLINK enables highly dynamic and accurate synchronism of all connected MASTERDRIVES MC units to be realized on account of its extremely fast data transfer by transmitting a strictly time-equidistant and jitter-free SYNC telegram in each cycle. Typical areas of use are, for example, all applications requiring a high degree of synchronism (angular synchronism) of individual MASTERDRIVES MC units to each other. A typical area of application is, for example, the replacement of previously mechanically coupled moving axes by individual electric drives, e.g. for printing machines. SIMOLINK can further be used in highly dynamic coordination tasks of individual MASTERDRIVES MC/VC units, such as in the motion control of individual axes on packing machines.

Components

SIMOLINK consists of the following components:

- ◆ SIMOLINK master
Interface for higher-level automation systems,
e.g. SIMATIC FM458 or SIMADYN (see Chapter 8.3.8)
- ◆ SIMOLINK board (SLB)
Interface for drives (see Chapter 8.3.4)
- ◆ SIMOLINK switch (see following section)
- ◆ Fiber-optic cable
Connecting medium of nodes on the SIMOLINK ring (see Chapter 8.3.4)

The SIMOLINK master and the SIMOLINK board are active nodes on SIMOLINK. The SIMOLINK switch is a passive node.

- ◆ Active nodes can receive and send telegrams and can read or write the contained information.
- ◆ Passive nodes can only pass on received telegrams. It is not possible for them to process the information contained therein.

SIMOLINK switch

The SIMOLINK switch is a passive node which has a "switching" function between two SIMOLINK rings.

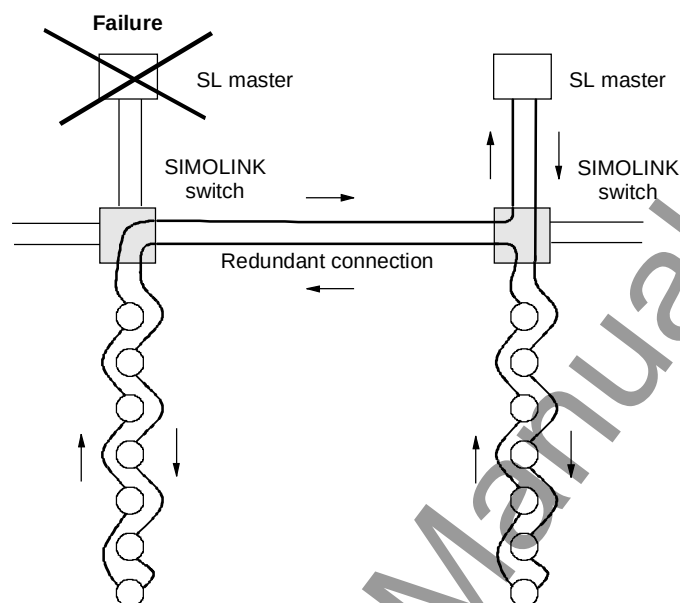


Fig. 8.3-1 Example of an application for the SIMOLINK switch

SIMOLINK features

- ◆ The transfer medium is a fiber-optic cable. Either glass or plastic fiber-optic cables can be used.
- ◆ SIMOLINK has the structure of a ring of fiber-optic cables where each node acts as a signal amplifier.
- ◆ Thus, the following distances can be realized, depending on the selected medium:
 - max. 40 m between each node on a plastic fiber-optic cable or
 - max. 300 m between each node on a glass-fiber-optic cable.
- ◆ Up to 201 active nodes ¹⁾ can be interlinked on SIMOLINK.

¹⁾ From now on, the active nodes are only referred to in the text as nodes

- ◆ MASTERDRIVES MC only:
Synchronization of the nodes is effected through a SYNC telegram which is generated by a node with a special function, the dispatcher function, and is received simultaneously by all other nodes. The SYNC telegram is generated absolutely time-equidistantly and jitter-free. The time between two SYNC telegrams is the bus cycle time of SIMOLINK and, at the same time, it corresponds to the common clock time for synchronization of all connected nodes.
- ◆ Data transfer between nodes is effected strictly cyclically in the bus cycle clock time. This means that all data written or read by the nodes is transferred between two SYNC telegrams. Upon receipt of the SYNC telegram, the previously received data in every MASTERDRIVES MC/VC unit is passed on to the control system of the converter as being the currently applicable data. This ensures that the latest applicable data is available to all nodes on the bus at the same time.

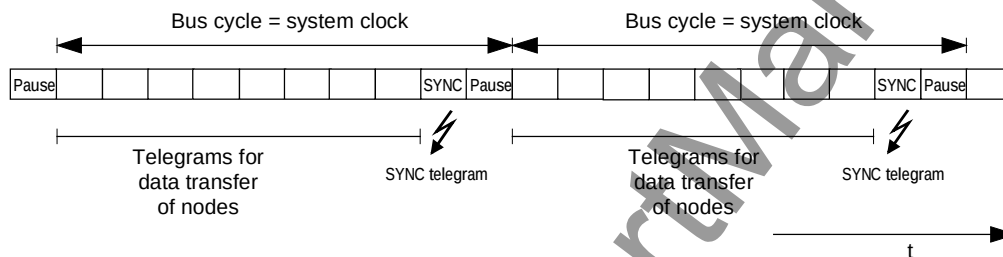


Fig. 8.3-2 SIMOLINK telegram traffic

- ◆ The transfer rate is a fixed 11 MBit/s
- ◆ A 32 bit word can be transferred in each telegram. The total length of each telegram is 70 bit, including the 32 bit net information. Thus, at a transfer rate of 11 Mbit/sec, a telegram has a transfer time of $6.36 \mu\text{s}$.
- ◆ SIMOLINK has a very high data throughput. This means that all the telegrams are sent without an interval directly one after the other. For example, with a selected bus cycle time of 1 ms, 155 telegrams with data contents (value of 32 bit per telegram) can be transferred via SIMOLINK.
- ◆ The functionality of the SIMOLINK application defines the assignment of telegrams to nodes. There are two possible applications:
 - the peer-to-peer functionality and
 - the master/slave functionality.

**Peer-to-peer
functionality**

This field of application describes all applications for which there is no dedicated logical master for distributing information via SIMOLINK. A typical application example here today is the "Continuous material throughput" which is implemented with the peer-to-peer protocol, in which drives have equal rights in a logical sense (peer-to-peer) in their exchange of information with each other. In accordance with the definition of the term "peer-to-peer", (communication between equals), this function is described as the "Peer-to-peer" functionality on SIMOLINK. This functionality enables extremely fast, synchronized and absolutely freely selectable transfer of data (no restrictions imposed by the physical bus configuration as in the peer-to-peer protocol) between MASTERDRIVES MC/VC units. The system needs to be designed with a "timing generator" for generating the telegram traffic and which keeps the bus system fully functional. The SIMOLINK dispatcher provides the interface to this function in the converter. The term "Dispatcher" is used to describe the principle characteristic of this interface: independent, constant dispatching of telegrams. The interfaces in the other MASTERDRIVES MC/VC units on SIMOLINK operate as "Transceivers".

The term "Transceiver" is made up of the words "Transmitter" and "Receiver". It means that a transceiver can receive and then send telegrams, but it cannot initiate telegram traffic itself (main difference to the dispatcher).

**Master/slave
functionality**

In this case, a central station (logical master) supplies all the other nodes (logical slaves) on the bus system with information (control bits, setpoints, etc.) This function is referred to hereafter as the "Master/slave" functionality. It refers to the logics of data transfer between the nodes on SIMOLINK. The system needs to be configured with a SIMOLINK interface in the central station (master) in this application field. This interface is both the logical master for data transfer and the initiator and monitor for telegram traffic on SIMOLINK (= dispatcher function). This interface, including its functions contained in an automation system, is referred to as the "SIMOLINK master".

The interfaces in the other nodes, e.g. in the converters, are "SIMOLINK transceivers".

NOTE

There is always only one node with a dispatcher function in the SIMOLINK ring. This is either a SIMOLINK board with dispatcher parameterization or a SIMOLINK master.

8.3.2 Peer-to-peer functionality

Each node on SIMOLINK has an active function either as a transceiver or as a dispatcher. There is always only one node with a dispatcher function in the SIMOLINK ring. All the other nodes are transceivers.

Bus topology

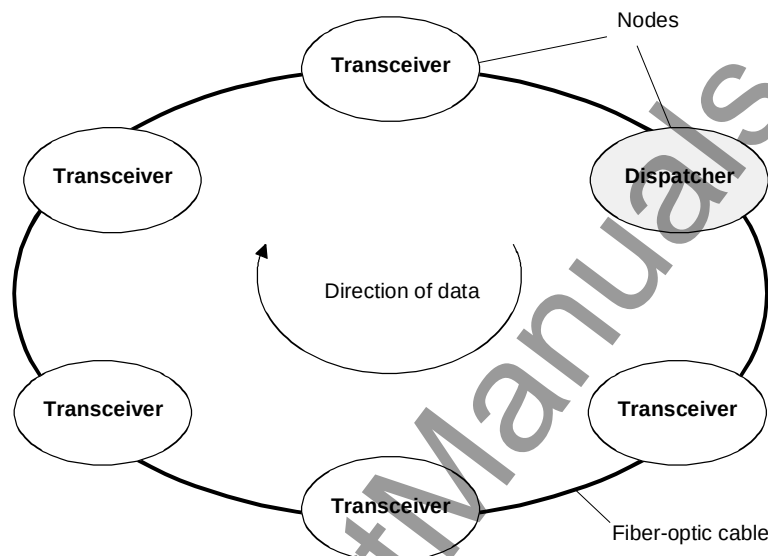


Fig. 8.3-3 SIMOLINK with dispatcher

Dispatcher

A table (= task table) is defined in the SIMOLINK dispatcher in which all telegrams are entered in the order in which they are sent. Each telegram has an address section (= node address) and a subaddress section (= channel number) in the telegram header. The telegrams are entered in the task table with ascending address and subaddress sections. The SIMOLINK dispatcher initiates telegram traffic by dispatching all the telegrams one after the other, beginning with the telegram with the lowest address and subaddress section according to the entry in the task table. As soon as the SIMOLINK dispatcher has dispatched all the telegrams, it sends a synchronization telegram (SYNC telegram) and a pause telegram. After this, it dispatches the first telegram from the task table again without any delay.

NOTE

The dispatcher can upread or overwrite the data contents of telegrams, as can every transceiver.

Transceiver

Each transceiver receives the telegrams (all of them) initiated by the dispatcher and can upread their data contents (value of 32 bit per telegram) or overwrite them with their own data, in accordance with a determined rule. The received telegrams are passed on to the next node in the ring, irrespective of whether the data contents have been read, overwritten or revised. Nodes with a transceiver function cannot maintain data traffic in the ring on their own.

8.3.3 Application with peer-to-peer functionality

Principle

The peer-to-peer functionality with SIMOLINK corresponds in principle to the peer-to-peer link with which you may already be familiar from MASTERDRIVES and SIMOREG, i.e. exchange of process data between MASTERDRIVES MC/VC units with the following additional advantages:

- ◆ Very fast (11 Mbit/s; one hundred and fifty 32-bit data in 1 ms)
- ◆ Freely selectable, i.e. every MASTERDRIVES MC/VC can send process data to every other MASTERDRIVES MC/VC, or receive data from it.
- ◆ Maximum of sixteen 32-bit data per MASTERDRIVES MC/VC possible via SIMOLINK; i.e. every MASTERDRIVES MC/VC can receive up to 8 32-bit data via SIMOLINK, and send up to 8 32-bit data to other MASTERDRIVES MC/VC units.

Basic principle of addressing

The telegram address is not interpreted as a "destination address" (which determines to whom the information is to be sent), but is understood to be a "source address". This indicates where the information is coming from.

Dispatchers and transceivers write their information (= data) in the telegrams assigned to them (node address = address in telegram) on the bus. Dispatchers and transceivers can read every telegram on the bus. For this purpose, the nodes have separate storage areas for receive data and transmit data.

Addressing mechanism - writing

The dispatcher and transceiver nodes only transmit information (= write data) in the telegrams which are assigned to them via the address. A maximum of 8 x 32-bit data can be transferred in 8 telegrams (same address and channel number from 0 to 7). A channel number is assigned to each 32-bit value and thus clearly also a telegram on the bus.

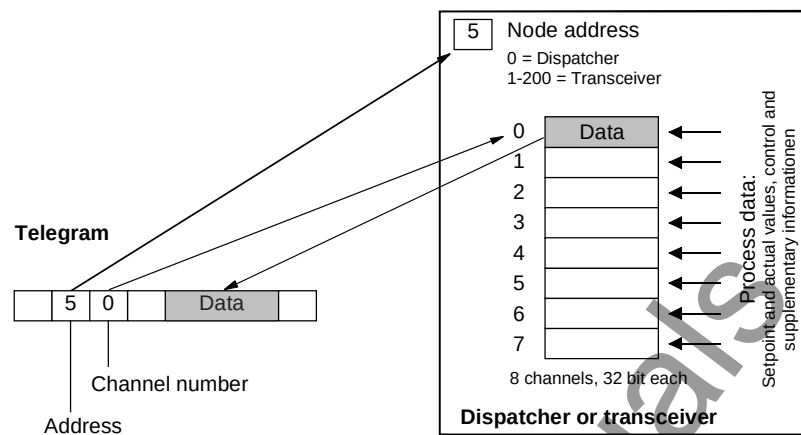


Fig. 8.3-4 Writing data

Addressing mechanism - reading

The active nodes (dispatcher and transceivers) can read the data of any telegram on the bus (also their own telegrams; separate storage areas for transmit data and receive data). A maximum of 8 different telegrams (8 x 32-bit data) can be read. For this purpose, **those** addresses and channel numbers whose data are to be read are parameterized as receive telegrams in the dispatcher or in the transceivers. This parameterization is carried out before data traffic is started up; in the case of MASTERDRIVES, for example, via the parameters of the converter.

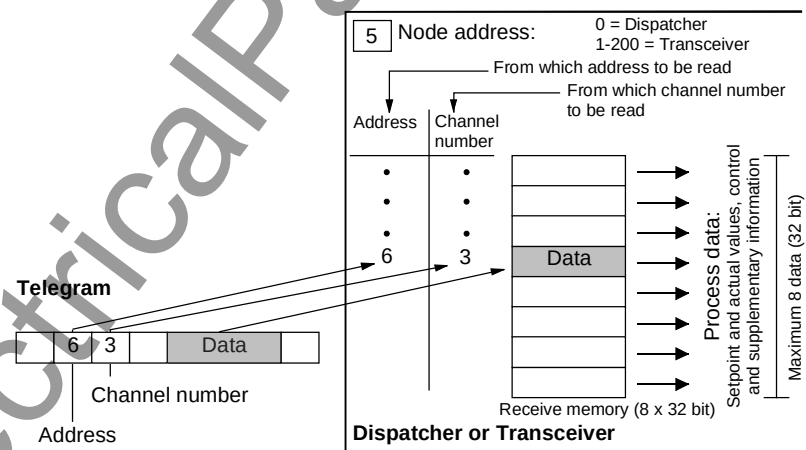


Fig. 8.3-5 Reading data

Example

The node with the address 5 (= transceiver interface) can "deposit" a maximum of 8 x 32 bit data on the bus. This means that the transceiver writes its data (32 bit in each case) in telegrams with address 5 and channel numbers 0 to 7. All the active nodes on SIMOLINK (the dispatcher as well as the transceivers) can decide whether they want to read this data. If, for example, a node wants to read the data of node 5 (= address 5) with channel number 2, this has to be configured accordingly. In this case, the address 5 and the channel number 2 have to be configured as the "Reading address".

Data transfer

In the "Peer-to-peer" application with the dispatcher, only process data (control and status words, setpoints and actual values) are transferred. When using a data area in the telegram, in the case of process data with word size (= 16 bit), two process data per telegram can also be transferred or read.

NOTE

All usable telegrams must be entered in the task table of the dispatcher.

Applications

Typical applications for SIMOLINK are the implementation of digital setpoint cascades in which one or more setpoints can be given to the slave drives by a MASTERDRIVES MC/VC unit acting as master drive.

8.3.4 Components of the peer-to-peer functionality

SLB optional board

The SLB optional board (*SIMOLINK board*) is used for linking drives to SIMOLINK.

Each SLB optional board is a node on SIMOLINK.

The optional board is provided with three LED displays which supply information on the current operating status.

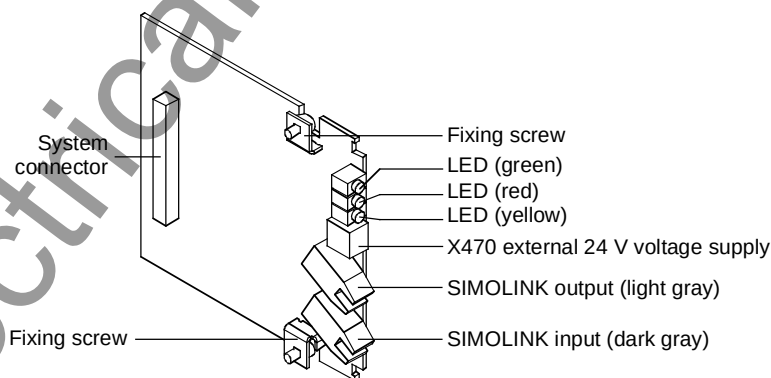


Fig. 8.3-6 SLB optional board (SIMOLINK board)

The SLB optional board links the converters/inverters to SIMOLINK. It can be used as the SIMOLINK dispatcher or as a SIMOLINK transceiver. The functionality is determined by parameterization.

Fiber-optic cable medium

A fiber-optic cable is used as the transfer medium in SIMOLINK. Plastic or glass-fiber optic cables can be used.

For cable lengths (the distance between two nodes) up to a maximum of 40 m, plastic cables are used.

NOTE

Recommendation:
Plastic fiber-optic cables from Siemens; CA-1V2YP980/1000,200A

For cable lengths (distance between two nodes) up to max. 300 m, fiber-optic cables with a glass core and a plastic sheath can be used.

NOTE

Recommendation:
Fiber-optic cables with glass core from Siemens; CLY-1V01S200/230,10A

The above-mentioned fiber-optic cables do not have an outer sheath. When using them for wiring outside switch cabinets, the cables must either be laid in cable ducts or conduits or suitable cables with an outer sheath must be used. On cables with an additional outer sheath, this must be removed before fixing the connector at the end of the cable as the connectors cannot accommodate the sheath. Therefore, when selecting the cable, one must make sure that the then remaining outer fiber diameter of 2.2 mm for attaching the connector is maintained.

24 V voltage supply

The SLB optional board has a 24 V voltage input for the external voltage supply of the board. This ensures that data transfer is maintained in SIMOLINK even with powered-down converter/inverter. Changeover between internal voltage supply from the converter/inverter and external voltage supply is carried out automatically, with priority being given to the external voltage supply.

NOTICE

A changeover must not be performed during bus operation because it generates a reset signal on the option board as a result of which bus operation is interfered with.

NOTE

If the SLB option board is continuously supplied with external voltage, switching the internal voltage supply on and off (on/off switching of the CU) after the internal voltage supply is switched on will result in a bus operation fault. Fault-free operation after switching in the internal voltage supply is only possible under the following conditions:

- ◆ SLB in transceiver operation and with part number 477 458 9000 15
- ◆ CU has firmware version MASTERDRIVES MC from V1.66 or MASTERDRIVES MC Performance 2 from V2.32, MASTERDRIVES VC from V3.42
- ◆ Before switching off, the SLB option board was actively participating in bus operation (i.e. the SLB was correctly parameterized by the CU)

8.3.5 Parameterization of the peer-to-peer functionality

The data traffic is determined by the parameterization of the dispatcher and the transceivers.

The configuration for enabling process data to be sent from a MASTERDRIVES MC/VC unit is determined by the BICO technique. The BICO technique is also used to determine the position in the control system at which the received process data are to act.

NOTE

Setting is carried out exclusively by means of the parameters of the MASTERDRIVES MC/VC unit. No additional configuration tool is required.

Parameterization of the SLB is carried out via the PMU, the OP1S or a PC with the DriveMonitor start-up tool.

The following parameterizations are necessary for configuring the SLB:

◆ **P740: SLB node address**

- 0: simultaneous selection of dispatcher function
- 1 - 200: simultaneous selection of transceiver function

◆ **P741: SLB telegram failure time (dispatcher and transceiver)**

The telegram failure time is a parameterizable failure time which is stored in every node. The telegram failure time determines the maximum time between two HW interrupts. The HW interrupt is generated by the interface after receipt of a SYNC telegram.

If a node does not receive a SYNC telegram within this time (→ no HW interrupt), the "TlgOFF" diagnostic bit is set in every node in which the telegram failure time is running.

The telegram failure time is activated after receipt of the first SYNC telegram.

The telegram failure time should be at least twice as long as the SIMOLINK cycle time.

If you use the SIMOLINK, telegram failure monitoring should be activated! $P741 = 4 \times P746$ (SLB bus cycle time) is recommended for the SLB telegram failure time. See also the function diagram [140].

- ◆ **P742: SLB transmit power** (dispatcher and transceiver)
The power of the fiber-optic transmit block for every node can be set by a parameter.
The transmit power can be set in the stages 3 = 40 m, 2 = 25 m and 1 = 15 m cable length. This scaling means, for example, that in stage "2" a transmit power is set for bridging a distance of up to 25 m plastic fiber-optic cable.
 - Localization of fault sources in the medium upon start-up:
Hidden fault sources on the transfer medium which may not be possible to detect with full power strength can be better localized by reducing the transmit power. Possible causes of the faults may, for example, be that the bending radii are too small or that the contacts of the fiber-optic cable fibers in the connector are poor.
 - Ageing of the fiber-optic cable components:
By reducing the transmit power, the ageing process of the fiber-optic cable components can be slowed down.
- ◆ **P743: Number of nodes** (dispatcher and transceivers)
With this function, each node can compensate for its individual time delay t_{delay} for compensation of runtime delays caused by the signal conversion in each node.
Formula for transceivers at the n-th position in the ring:

$$t_{\text{delay},n} = [\text{number of nodes} - n] \times 3 \text{ bit times};$$
 The "Number of nodes" value is specified to the nodes as a parameter.

NOTE

The position n at which the node is situated in the ring is calculated automatically in the SIMOLINK starting cycle.

The SL master or dispatcher sends a special telegram with the address 253 "Count nodes" and the starting value 1. Each transceiver which receives this telegram remembers this number (= Count number) and then increments the data contents by the value 1. In this way, the node has the count number 1 directly after the SL master or dispatcher while the SL master or dispatcher has the maximum count number, which also corresponds to the number of node. The result of this procedure can be checked in parameters r748 Index 7 (position of the node in the ring) and r748 Index 8 (number of nodes in the ring).

NOTE

The formula stated above neglects the throughput delay of the SIMOLINK switch. Generally, this is permissible as the switch, for example, is usually situated at the beginning of the ring and thus does not cause any delay between transceivers.

The transceiver n waits $t_{\text{delay},n}$ before it can give an HW interrupt to the unit application after receipt of the SYNC telegram. This ensures that the interrupts to the unit applications of all nodes are effected as synchronously as possible.

Normally, this parameter does not have to be altered. The dispatcher passes on the determined number of nodes to the slaves automatically. The latter deduce the necessary delay time from this if the parameter has been set to 0 (= automatic calculation). Only in the case of high accuracy requirements and special influences (SIMOLINK switch, long leads) might it be necessary to manually alter this parameter.

The calculated delay time $t_{VZ,n}$ (normalized to 3 bit times) can be checked in parameter r748 Index 6.

- ◆ **P744: SLB selection** (dispatcher and transceiver)
Only MASTERDRIVES MC: Is for selecting source of synchronization and data when there are two SIMOLINK boards or CBPs in a MASTERDRIVES unit.
- ◆ **P745: SLB channel number** (dispatcher)
This parameter is used to set the number of used channels (max. 8). The selected value is firmly applicable for all nodes on the bus.
- ◆ **P746: SLB cycle time** (dispatcher)
This is used to set the bus cycle time. The bus cycle time can be set from 0.20 ms to 6.50 ms in a 10 μ s grid.

NOTE

The dispatcher determines the task table from the SLB channel number and the SLB cycle time (consecutive numbering, starting with node address 0 and channel number 0, at first incrementing the channel number) in accordance with the following formula:

$$n = \left(\frac{P746 + 3.18 \mu s}{6.36 \mu s} - 2 \right) \times \frac{1}{P745}$$

n: Number of addressable nodes (checked at r748 Index 4)

Task table example:

P746 = 0.20 ms; P745 = 2; $\rightarrow n = 15$

Address	0	0	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	8

Channel	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Address	9	9	10	10	11	11	12	12	13	13	14	14	255	255

Channel	0	1	0	1	0	1	0	1	0	1	0	1	0	0

Only those addresses and channels listed in the task table are processed.

- ◆ **P 749: SLB read address** (dispatcher and transceiver)
Is for setting the channels to be read. Input is in the notation address.channel. Up to 8 channels can be defined by the 8 parameter indices. The data in these channels are transferred via connectors K7001 - K7016 or KK7031-KK7045.

◆ **P 751: Source SLB transmit data**

Used to select the connectors to be transmitted via SLB channels 1 to 8 (subdivided into low-word and high-word). Double connectors must be entered in two consecutive indices, so that they are transmitted with the full resolution.

◆ **P 755: SIMOLINK configuration (dispatcher)**

When data are transferred from one slave to another, the problem arises that the dead time on the bus depends on the node address of the transceiver. Specifically, this means that data transfer from slave 2 to slave 1 via the dispatcher takes one cycle time longer than data transfer between slave 1 and slave 2. The reason for this is that the data are collected by the dispatcher and are not transmitted onward until the next cycle. This problem can be eliminated by addressing each transceiver twice in one SLB cycle, a first time to obtain the current data of the transceiver which are then available in the dispatcher, and a second time to transmit that data onward, although the number of addressable nodes are thereby reduced by half.

Parameter values (only dispatcher):

- xxx0: No dead time compensation
- xxx1: Dead time compensation activated → Number of addressable nodes = $n / 2$

When 2 SIMOLINKs are operated in a converter, data adoption and synchronization can be changed over from one to the other (cf. P 744). If this changeover is also to be possible during operation (converter status °014), this is to be enabled by the user. This function is only provided in the case of MASTERDRIVES MC units. Parameter values:

- xx0x: No changeover during operation (converter status °014)
- xx1x: Changeover of synchronization and data transfer allowed during operation

In the case of operation in a ring with a master which triggers the bus cycle externally (e.g. SIMADYN D), the MASTERDRIVES slaves are to be configured for exact adherence to the bus cycle time. Otherwise, it is assumed internally that the bus cycle time is determined by the particular number of telegrams. The actual bus cycle time does not then correspond exactly to the time which has been set. This function is only provided in the case of units which can be synchronized (MASTERDRIVES MC).

Parameter values:

- x0xx: Bus cycle time corresponding to the calculated number of telegrams (normal operation)
- x1xx: Exact adherence to the set bus cycle time

8.3.6 Diagnostics of the peer-to-peer functionality

The following diagnostics information is available to the user:

LED displays

Three LED displays are provided on the front section of the SLB optional board which supply information on the current operating status.

Operating display

LED	Status	Diagnostic information
Green	Flashing	Fault-free net data transfer via SIMOLINK
Red	Flashing	SLB operating
Yellow	Flashing	Data exchange with basic unit is okay

Table 8.3-1 SLB operating display

Fault display

LED	Status	Diagnostic information
Green	off/on	No net data exchange possible via SIMOLINK; bus cable is not connected or is defective
Red	off/on	Voltage supply for SLB cut off; replace SLB or basic unit
Yellow	off/on	No data exchange with the basic unit; bus cable is not connected or is defective; replace SLB or basic unit

Table 8.3-2 SLB fault display

Binectors

- ◆ **B0041: Time out:**
Bit = 1 indicates that an interruption has occurred in cyclic data transfer. This status remains active until cyclic data transfer has been resumed.

NOTE

The reaction time is permanently stored in the SLB and cannot be changed.

Every time "Time out" occurs, the SLB diagnostics parameter (r748, Index 3) is incremented by the value 1 (→ statistics).

At the same time, the address of the node that has first noticed the interruption in the ring can be upread in r748, Index 5.

- ◆ **B0040: SLB telegram failure**
Bit = 1 indicates that the telegram failure time set in the "SLB TlgOFF" parameter (P741) has run out in this node, without a valid SYNC signal having been received.
- ◆ **B0042: Start alarm**
Bit = 1 indicates that the SIMOLINK ring is physically open and that a start cannot be carried out. This status is also signaled by alarm A002.
Bit = 0 indicates that the SIMOLINK ring is physically closed.
- ◆ **B0043: Drive synchr. (only MC)**
Bit = 1 indicates whether the CU is synchronized to the SIMOLINK BUS. Corresponds to the inverse of alarm A003.

- ◆ **B0047: SLB2 timeout** (only MC)
Bit = 1 indicates that a timeout has been detected on the passive SIMOLINK bus.
- ◆ **B0048: SLB2 start** (only MC)
Bit = 1 indicates that the passive SIMOLINK ring is physically open and a start cannot be carried out. This binector corresponds to alarm A004.
- ◆ **r748: SLB diagnostics**
The diagnostic parameter is used to retrieve various status data of the SIMOLINK bus. The following information can be read from the various indices:
 - r748.1: Number of error-free SYNC telegrams (corresponds to the bus cycles that have elapsed without error).
 - r748.2: Number of CRC errors (telegrams with errors).
 - r748.3: Number of timeout errors (bus interrupt). Note: On bus initialization, data traffic is interrupted several times, causing some timeout errors.
 - r748.4: (Dispatcher only) last addressable address; on initialization the last address addressable in the selected configuration is entered here.
 - r748.5: Address of the station that has signaled timeout.
 - r748.6: Here, the hardware interrupt delay is stored that was calculated from the number of stations set (P743), or from the number of stations transferred during initialization (with automatic parameterization P743 = 0), and the position of the station in the SLB ring.
 - r748.7: Position of the station in the SLB ring (result of the count during initialization).
 - r748.8: Number of stations in the SLB ring (result of the count during initialization).
 - r748.9: (MASTERDRIVES MC) deviation from the synchronization point. Cannot be synchronized, the value is set to NO_SYNCHRONIZATION (= 65535). Should not fluctuate outside 65515 (-20) and 20.
 - r748.10: Pulse period adapted to the bus cycle time in 100 ns (e.g. pulse frequency 5kHz → display value 2000). If no synchronization is possible, the value NO_SYNCHRONIZATION (= 65535) is entered.
 - r748.11: Current state of the T0 counter. Should be 0 for active synchronization (MASTERDRIVES MC only).
 - r748.14: Current state of the time slice counter. Should be 0 for active synchronization (MASTERDRIVES MC only).
 - r748.15: Bus cycle time implemented in 10 μs.
 - r748.16: Bus cycle time transmitted during initialization from the master/dispatcher in 10μs.

◆ **r750: SLB receive data**

In indices 1 to 16, the received data word 1 to 16 are displayed.

◆ **r752: SLB transmit data**

In indices 1 to 16, the received data word 1 to 16 (corresponds to channel 1 to 8) are displayed.

8.3.7 Synchronization of the control circuits by means of the bus cycle time (MC only)

The bus cycle time must be in a defined proportion to the time slots of the individual closed-loop control units in order to synchronize the decentralized lower-level control loops in the converters. The following applies to the time slots in the case of MASTERDRIVES MC:

- ◆ Current control in time slot T_0
- ◆ Speed control from V1.30 in time slot $T_1 = 2 T_0$
from V2.00 in time slot T_0
- ◆ Position control in time slot $T_3 = 8 T_0$
- ◆ Synchronism $T_3 = 8 T_0$ or $T_4 = 16 T_0$
- ◆ The time slot $T_0 = 1/\text{pulse frequency}$ is set on the MASTERDRIVES MC by selecting the pulse frequency (P340).

Thus the following applies to the selection of the bus cycle time:

Bus cycle time $P746 = 1 / P340 * 2^n$

n = slowest time slot to be synchronized T_n ;

where $n \in \mathbb{N} = \{2, 3, \dots\}$

T_2 can be synchronized as a minimum. Individual synchronization of T_0 or T_1 cannot be implemented.

- ◆ **Example:**
If the position control loops of the various converters have to be synchronized to each other, the selected bus cycle time has to be a 2^n -fold quantity of $4 T_0$. At a pulse frequency of $P340 = 5.0 \text{ kHz}$ the resulting bus cycle time $P746$ is at least 0.80 ms ($4 * 200 \mu\text{s}$).

Standard parameterization

Synchronization of the slow time slots at a low bus cycle time

In a number of applications it is necessary to set a low bus cycle time and at the same time to synchronize the slower time slots. For this purpose, it is necessary to transfer additional time slot information from the dispatcher over the SIMOLINK to the transceivers. This information is generated in the dispatcher at connector K260. It must be transferred via the SIMOLINK and input to the transceivers at parameter P753. In parameter P754, the slowest time slot to be synchronized is set.

Example:

The bus cycle time should be as short as possible while at the same time the synchronization control is synchronized in T_4 for all drives. At a pulse frequency of 5 kHz (P340), the shortest bus cycle time is 0.80 ms (P746). The dispatcher sets connector K260 to SIMOLINK word 3 (P751 Index 3 = 260) for all transceivers (P753 = 7003). Parameter P754 is set to 4 (for T_4) at the dispatcher and at the transceivers.

Synchronization parameter assignment

Parameters:

- ◆ **P 746: SLB cycle time** (dispatcher)
Serves for setting the bus cycle time. The bus cycle time can be set from 0.20 ms to 6.50 ms in increments of 10 μ s. The bus cycle time of the dispatcher is transferred automatically to the slaves. The bus cycle time in effect can be upread from parameter r748 Index 15.
- ◆ **P753: Sync. time counter** (transceiver)
Input parameter for additional time slot information from the dispatcher. This parameter must be connected to the SIMOLINK-connector (K7001 - K7016), which contains the time slot information.
- ◆ **P754: Max. sync. time slot** (dispatcher and transceiver)
The slowest time slot n to be synchronized is entered here. This function will not work unless parameter P753 is connected correctly.

Connectors:

K260: Time counter (dispatcher only)

This connector contains additional time slot information from the dispatcher.

8.3.8 Synchronization diagnostics (MC only)

The following diagnostics information is available to the user:

Binectors

◆ **B0043: Drive synchronism**

Bit = 1 indicates that the drive is running in synchronism.

Bit = 0 indicates that the drive is not yet running in synchronism or cannot be synchronized. This status is also signaled by alert A003.

Parameters

◆ **r748 Index 9: Synchronism deviation**

The value should vary between -20 (= 65515) and 20, if synchronization is functioning. A stable value of 65535 indicates that synchronization is turned off because the pulse frequency (P340) and the SLB cycle time do not go together.

◆ **r748 Index 11: T0 counter**

The value should always be 0 when synchronization is functioning.

8.3.9 Switchover of the synchronization source (MC only)

MASTERDRIVES MC devices provide the option of plugging in and parameterizing two SIMOLINK modules and two CBP2s. Because of the physical situation, synchronization on only one of the communication modules and data transfer from only one of the two SIMOLINK modules is possible. Connecting up a second SIMOLINK ring would not therefore enable transfer of more data. The only possible applications are installations in which different machine configurations with different SIMOLINK-ring nodes are desired or necessary or where redundancy of the SIMOLINK rings is desired or necessary.

Parameter

◆ **P744: SLB selection** (dispatcher and transceiver)

BICO parameter, Index 1, is for selecting a source (binector) by means of which the active SIMOLINK (synchronization and data source) is defined when two SIMOLINK boards are present in a MASTERDRIVES unit.

By means of Index 2, the Profibus can be selected as the synchronization source. A SIMOLINK, if present, can no longer be used to transfer data; it only works as a transmitter in order to maintain telegram traffic in the SLB ring.

The synchronization source is selected according to the following scheme:

	744.1	744.2
SLB1 (lower slot) active	0	0
SLB2 (higher slot) active	1	0
CBP active	x	1

◆ **P755: SIMOLINK configuration**

If a 1 is set at the second position of the configuration parameter, changeover between the two SIMOLINK boards can be enabled during operation. This is only possible if the bus cycle time is the same even if changeover is enabled during operation.

- xx0x: No changeover during operation (converter status °014)
- xx1x: Changeover of synchronization and data transfer allowed during operation

Description of functioning

When two SIMOLINK boards are being operated in one unit, the active board is used for data transfer (same as when only one board is present). The passive board is initialized (SIMOLINK ring starts) and sends the parameterized transmit data. Synchronization and data transfer by the passive board is not possible. Transmit and read data are the same for the active and passive SIMOLINK. Different parameterizations of the two SIMOLINK boards are only possible in the case of the following parameters:

- ◆ Node address (P740)
- ◆ Number of nodes (P743)
- ◆ Number of channels (P745)
- ◆ Bus cycle time (P746)

The 1st index is allocated to SLB1 (lower slot) and the 2nd index is allocated to SLB2 (higher slot). Which of the two SLBs is the active one is determined by the selection (P744).

The diagnostic parameter (P748) always indicates the data of the active SIMOLINK.

If it has not been ensured by a master (e.g. SYMADYN D) that the two SIMOLINK rings are working synchronously, it can be assumed that, when a changeover is made to the passive SIMOLINK, there is no synchronization at first. The drives are synchronous with the bus again only after the synchronization time (at 5 kHz pulse frequency and 3.2 ms bus cycle time, maximum 7 sec.). In the case of applications where synchronicity is an essential component of functioning, changeover during operation should not be carried out.

Changeover during operation must be explicitly enabled by the user (P755). In addition, changeover during operation is prevented if synchronization to the previously passive SIMOLINK is not possible because different bus cycle times (P746) have been selected.

8.3.10 Special data and application flags

	For special functions, further options for data transmission are available via the SIMOLINK bus.
Application flags	With application flags it is possible to transmit an additional four binary items of information. These are not explicitly assigned to any station, i.e. every station can read and set the application flags. Resetting is only possible via the dispatcher/master. Parameterization: P747 Src.SLB Appl.Flags: Used to specify the binectors to be used as application flags. B7010 to B7013: These binectors indicate the applications flags received.
Special data	In addition to the 8 telegrams per station, a total of four special telegrams with 32 bits of useful data are available for data transmission in the SIMOLINK bus. The special telegrams can be read by any station but only written by the dispatcher (currently only MASTERDRIVES MC) / master. Parameterization: P756 Src.SLB special data: (dispatcher only) Used to specify the double connectors to be transmitted as special data. KK7131 to KK7137: These connectors indicate the special data received.

8.3.11 Configuration (example of peer-to-peer functionality)

Technology

Angular synchronism with 3 MASTERDRIVES MC units.

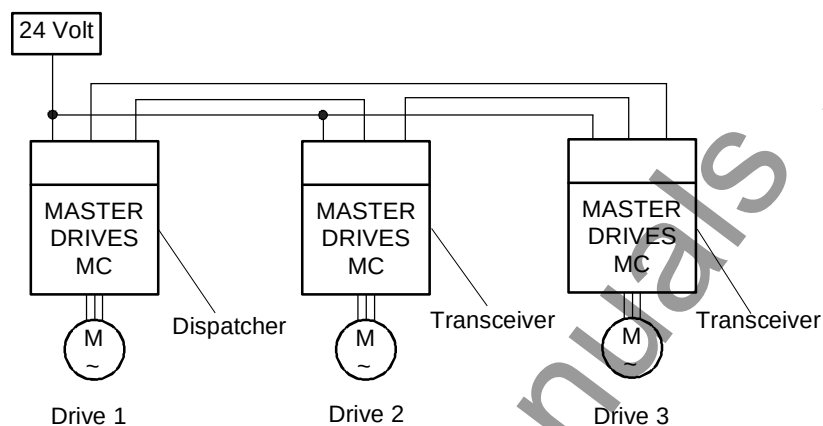


Fig. 8.3-7 Configuration example of peer-to-peer functionality

- ◆ Drive 1, master drive with integrated virtual master axis
The master speed setpoint for the drive group is specified via an analog input or via the PROFIBUS DP.
The integrated virtual master axis function generates a position, speed and acceleration setpoint for slave drives 2 and 3. In addition, the slave drives are powered up/down by the master drive (control word). This means that every slave drive is given its individual control word.
Vice versa the slave drives send their individual status word to the master drive. This results in the following table:

		Receive		
		Master drive 1	Slave drive 2	Slave drive 3
Transmit	Master drive 1		STW_2 Sset nset aset	STW_3 Sset nset aset
	Slave drive 2	ZW_2		
	Slave drive 3	ZW_3		

Table 8.3-3 Transmitting and receiving control/status words between master and slave drives

- ◆ Drive 2 and 3, slave drives with integrated position control

Communication

The 3 SIMOLINK interfaces must be parameterized as follows for transmitting the process data:

- ◆ **SLB in master drive 1 (dispatcher)**
The following 5 process data have to be transferred (written):
 - STW_2 = control word for drive 2
 - STW_3 = control word for drive 3
 - s_{set} = position setpoint
 - n_{act} = speed setpoint
 - a_{act} = acceleration setpoint
 5 telegrams (= 5 channels) are required for this.
- ◆ **SLB in slave drive 2 (transceiver)**
One item of process data is transferred in ZW_2 (written).
For this, one telegram (= 1 channel) is required.
ZW_2 = status word of drive 2
- ◆ **SLB in slave drive 3 (transceiver)**
One item of process data is transferred in ZW_3 (written).
For this, one telegram (= 1 channel) is required.
ZW_3 = status word of drive 3

Parameterization of the dispatcher

The following parameter settings are of significance for the dispatcher as the master drive:

- ◆ **P740 = 0** (Dispatcher function)
- ◆ **P745 = 5** (SLB channel number)
This means that each node is provided with five telegrams for writing.

NOTE

The setting always depends on the requirements of the node with the largest required number of channels. In this example, this is the dispatcher (master drive 1) with five telegrams.

- ◆ **P746 = 1 ms** (SLB cycle time)
A sufficient number of additional telegrams are automatically added to non-addressed nodes as is required to achieve this cycle time.
Synchronization of the control loops in the converter via the bus cycle time: The bus cycle time must be in a defined relation to the time slots of the individual controls for synchronization of the decentralized lower-level control loops in the converters. The following is applicable for the time slots on MASTERDRIVES units:
 - Current control in time slot T_0
 - Speed control from V1.30 in time slot $T_1 = 2 T_0$
from V2.00 in time slot T_0
 - Position control in time slot $4 T_0$

- The time slot $T_0 = 1/\text{pulse frequency}$ is set on MASTERDRIVES units by selecting the pulse frequency (P340). Thus the following applies for the selection of the bus cycle time:

Bus cycle time = $2^n \times \text{slowest time slot to be synchronized}$;
where $n \in N = \{2, 3, \dots\}$

Example:

If the position control loops of the various converters have to be synchronized to each other, the selected bus cycle time has to be an n-fold quantity of $4 T_0$.

Parameterization of the transceivers

Transceiver (slave drive 2) is given the node address 1 and transceiver (slave drive 3) is given the node address 2.

Parameterization of process data monitoring

The following diagrams show the assignment of the process data to be read or written using the example of master drive 1 and slave drive 2.

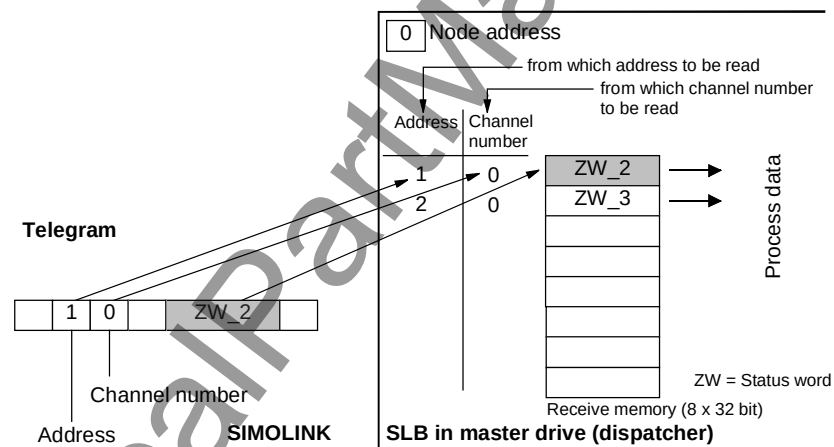


Fig. 8.3-8 Master drive 1, reading data

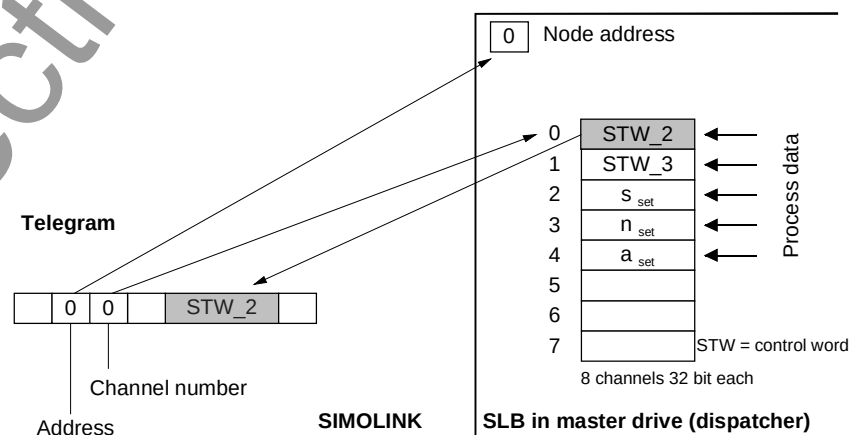


Fig. 8.3-9 Master drive 1, writing data

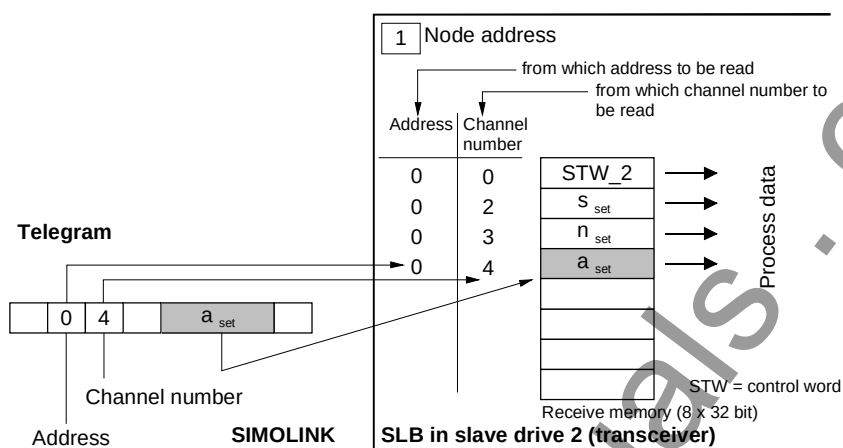


Fig. 8.3-10 Slave drive 2, reading data

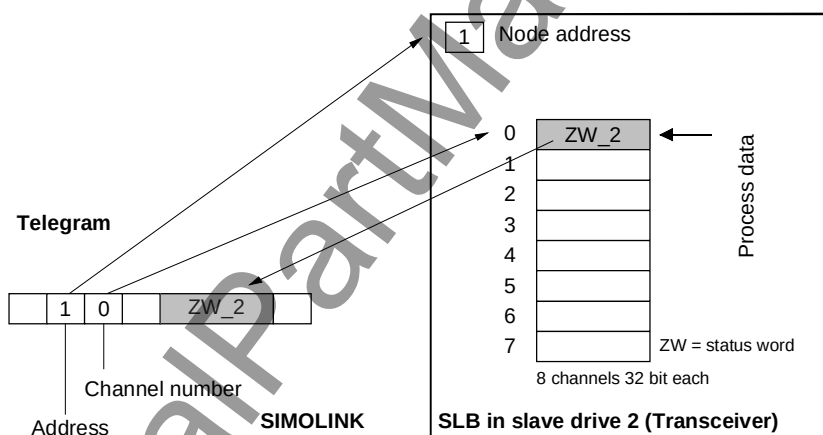


Fig. 8.3-11 Slave drive 2, writing data

8.3.12 Master/slave functionality

In the master-slave functionality, an SL master (SIMOLINK interface) operates in an automation system instead of the dispatcher (peer-to-peer).

There is always only one SL master in the SIMOLINK ring. All the other nodes are transceivers.

Bus topology

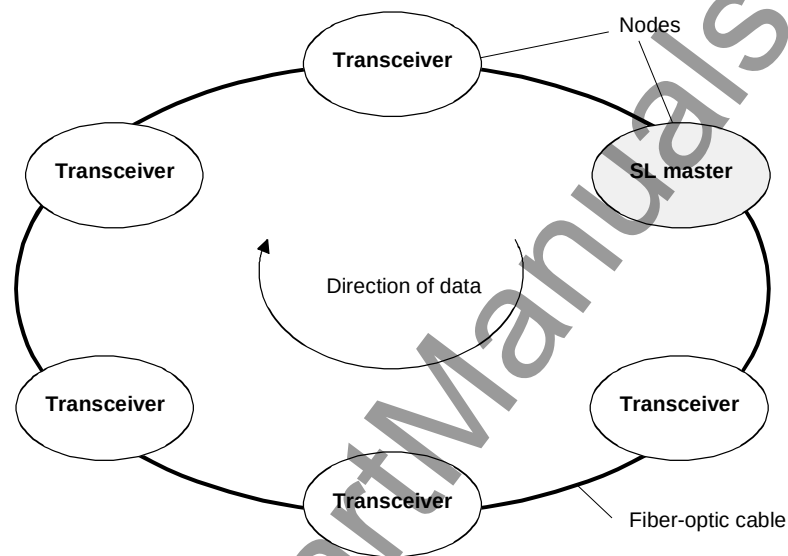


Fig. 8.3-12 SIMOLINK ring with SL master

SL master

The SL master is the SIMOLINK interface in "higher-level" open-loop and closed-loop control systems or industrial PCs. As far as the central control of telegram traffic is concerned, there is no difference between the dispatcher and the SL master. The task table also specifies in the case of the SL master which and how many telegrams the SL master shall send via the bus in one bus cycle.

Differences to the dispatcher:

- ◆ The applications of the "Master/slave" functionality require a different mechanism for data transfer than used in the "Peer-to-peer" functionality.
- ◆ Flexible address list (address gaps are possible), i.e. the task table can be configured a lot more freely.
- ◆ The number of channels used per transceiver can be individually determined and does not have to be identical. The maximum number of channels per transceiver is generally restricted to 8.
- ◆ The SL master itself has 8 channels for data transfer, just as in the case of the dispatcher or transceiver, however, at the same time it can use the telegrams with the address and channel number code of the transceivers for its data transfer.

NOTE

The SL master uses the "intelligence" and the possibilities offered by the open/closed-loop control system or the PC for configuring the task table. The following SL masters are currently available:

- SIMOLINK module in SIMATIC FM458
- Expansion board ITSL in SIMADYN D

Transceiver

In accordance with the peer-to-peer functionality

8.3.13 Application with master/slave functionality

Principle

This configuration is not based on the principle of freely selectable data transfer between MASTERDRIVES MC/VC units because control is effected from a higher-level automation system.

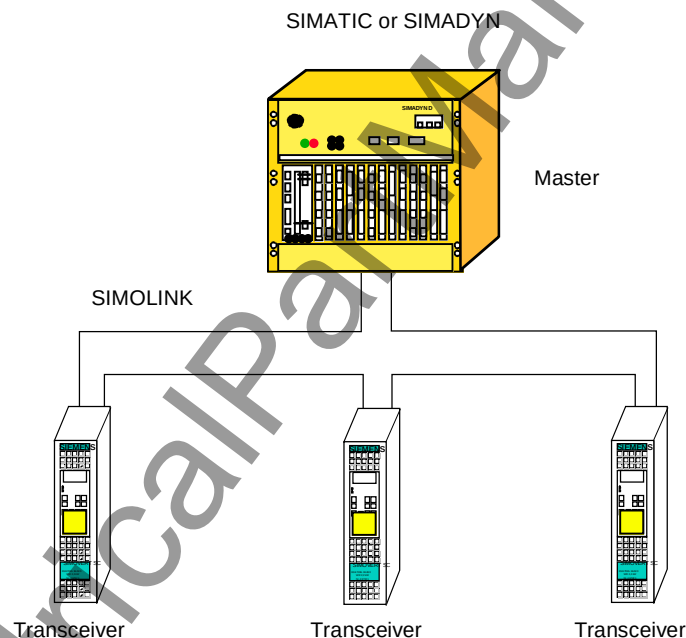


Fig. 8.3-13 Application example of master/slave functionality

There is a SIMOLINK interface in the automation system which also operates as a logical master in addition to the dispatcher function. This means that the automation system dispatches a maximum of eight 32 bit data back to the master by overwriting received telegrams with the dispatch information. This is the typical structure of data exchange according to the master/slave principle.

Rules for the exchange of data

- ◆ Each transceiver can read a maximum of 8 telegrams, however, the difference to the peer-to-peer functionality is that only telegrams which have an address corresponding to the address of the node or the master address 0 are read.

Note: These telegrams must, of course, be entered in the task table of the master.

- ◆ As in the case of the peer-to-peer functionality, each transceiver can only write data on telegrams whose telegrams have the address of the transceiver.
- ◆ The master can read and write on all telegrams.

The master can implement data exchange between two transceivers by transferring the received data of one transceiver to the telegrams (= address) of the other.

NOTE

Every transceiver can also read the telegrams of any other node. However, whether the read data are receive or transmit data, depends on where the respective nodes are situated in the SIMOLINK ring (definite data traffic in the SIMOLINK ring).

NOTICE

The SIMADYN-D master can be operated in different SIMOLINK operating modes. Modes 3 to 5 are suitable for error-free data traffic with MASTERDRIVES. Especially when using the asynchronous mode (= 1) problems can arise on the MASTERDRIVES MC/VC because the hardware interrupt generated by the bus cycle might not be equidistant and hardware interrupts triggered in too quick succession will cause a computation time overflow in the MASTERDRIVES MC/VC basic unit.

8.4 CBC Communications Board

8.4.1 Product description

The optional CBC board (Communication Board CAN) is used for connecting drives to higher-level automation units and other field units by means of the CAN (Controller Area Network) protocol.

View

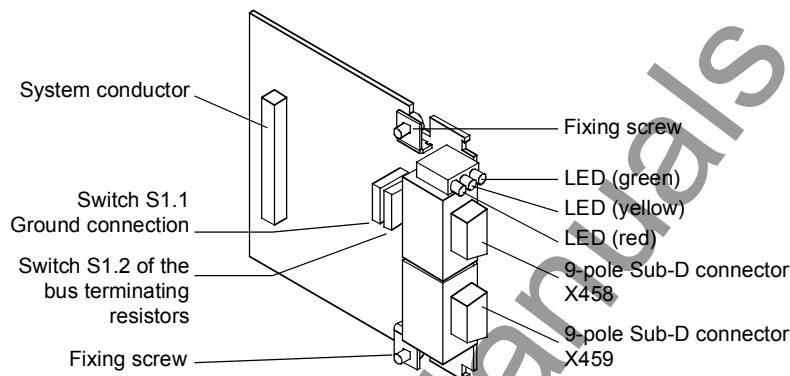


Fig. 8.4-1 View of the optional CBC board

Technical information

The optional board has three LEDs (green, yellow and red) for providing information on the current operating status.

It is supplied with voltage via the basic unit.

The CBC can be simply plugged into the electronics box of the converter and works with all software and hardware output-states of the MASTERDRIVES converters.

The CBC has a 9-pole Sub-D connector (X458) and a 9-pole Sub-D socket (X459) for connecting it to the CAN bus. The pins of these connecting elements are identically assigned and connected through internally. They are also short-circuit proof and floating.

Functions

The CAN (Controller Area Network) protocol is permanently specified in the international standards recommendation, ISO-DIS 11898. Here, however, only the electrical part of the physical and the data link layers are specified (layer 1 and layer 2 in the ISO-OSI-7 layer reference model). The CiA, with its DS 102-1 recommendation, defines the bus interface and the bus medium for use as an industrial field bus.

The CBC complies with the specifications in ISO-DIS 11898 and in DS 102-1.

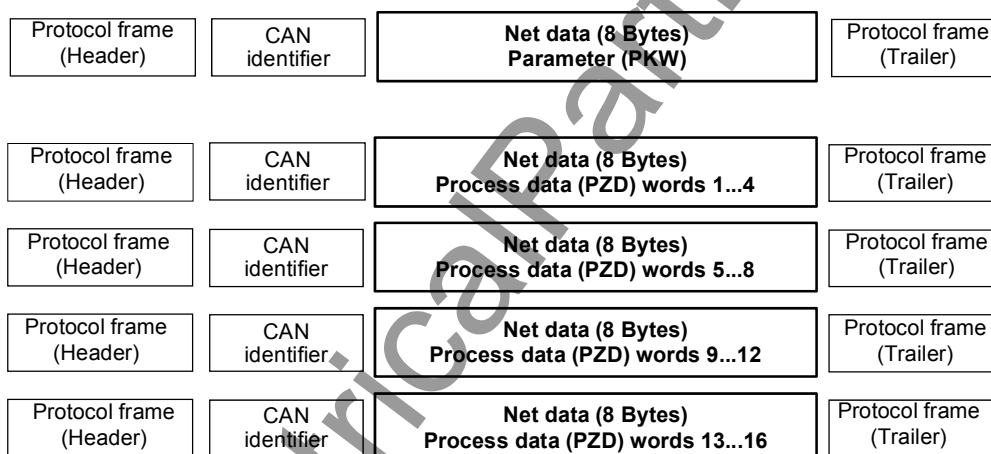
A data profile for variable-speed drives similar to the VDI/VDE guideline 3689 "PROFIBUS profile for variable speed drives" has not yet been defined. The specifications of the "PROFIBUS profile for variable speed drives" are therefore used for the net data.

For the drives, VDI/VDE guideline 3689 specifies the net-data structure with which a communications partner can access the drive slaves. The net-data structure is divided into two areas:

- ◆ The process-data area, i.e. control words and setpoints or status information and actual values
- ◆ The parameter area for reading/writing parameter values, e.g. reading out faults and reading out information on the properties of a parameter such as reading out min./max. limit values etc.

The number of process data (maximum 16) and activation of the parameter interface is parameterized on the unit. The parameterization of the net-data structure depends on the function of the drive within the overall automation system. The process data are processed with the highest priority and in the shortest time slices. The process data are for controlling the drive within the overall automation system, e.g. power-on/power-off, stipulation of setpoints, etc.

With the help of the parameter area, the user has free access to all parameters in the converter (CU and, if necessary, the TB) via the bus system. This facility can be used, for example, for reading out detailed diagnostic information, fault messages and so on. Information for visualizing the drive can thus be called using a higher-level system, e.g. a PC, without affecting the transfer of process-data.



PKW: Parameter ID word
PZD: Process data

Fig. 8.4-2 Structure of the net data in the telegrams of the CAN protocol

Controlling and operating the MASTERDRIVES converters via the CAN bus

In the process-data area (see Fig. 8.4-2), all the information is transferred which is necessary for controlling a speed-controlled drive in an integrated technical process. Control information (control words) and setpoints are given to the converter by the CAN-bus master. In the reverse direction, information on the status of the converter (status words) and actual values are transferred.

The CBC communications board stores the received process data in the dual-port RAM in the sequence in which they are transferred in the telegram.

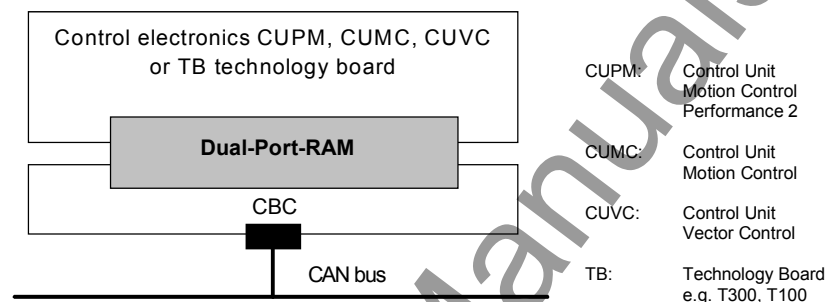


Fig. 8.4-3 Coupling of the CBC to the converter via the dual-port RAM interface

An address is assigned to each word in the dual-port RAM. The contents of the dual-port RAM in the converter (CU + if necessary the TB) can be freely routed by means of parameters, e.g. the second word in the process-data area of the telegram as a speed setpoint sent to the ramp-function generator connected downstream. The same mechanism applies to other setpoints and to each individual bit of the control word. The procedure is also used in the reverse direction for transferring actual values and the status words.

Besides supporting the normal exchange of process data, the CBC communications board also supports broadcasting (same process data for all drives on the bus), multicasting (same process data for a group of drives on the bus) and cross traffic (data exchange between the individual drives without participation of a CAN-bus master).

Diagnostic LEDs quickly provide the user with information on the current status of the CBC. Detailed diagnostic information can be directly read out of the diagnostics memory of the CBC by means of a diagnostic parameter.

8.4.2 Mounting methods / CBC slots

NOTE

The CBC can be directly mounted into Compact PLUS units. In all other types of unit in this series, it is mounted on the CUPM, CUMC or CUVC or connected in the electronics box with an adapter board.

8.4.2.1 Mounting positions of the CBC in MC Compact PLUS units

NOTE

In principle, the optional CBC board (Communications Board CAN) can be mounted in any slot. Please bear in mind, however, that an encoder board always requires Slot C.

Position of the slots

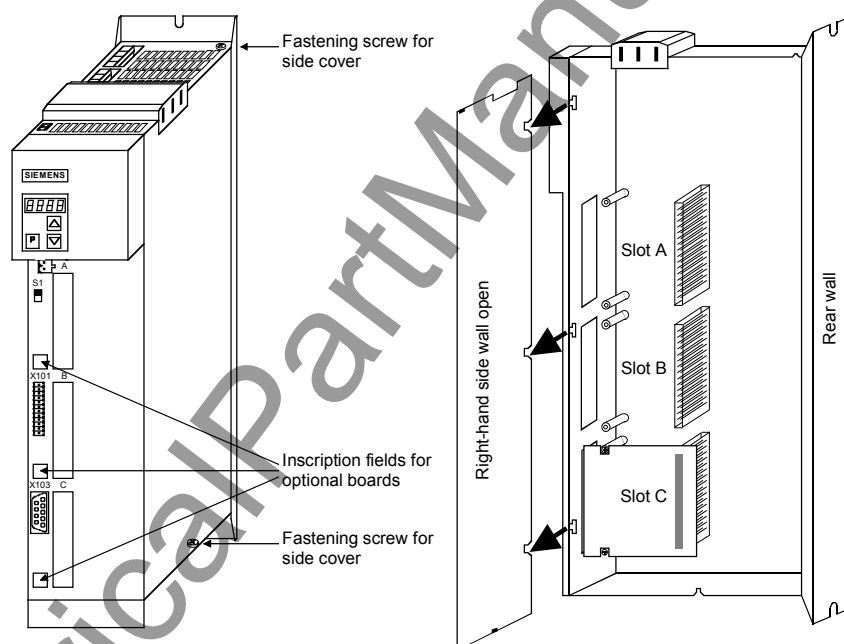


Fig. 8.4-4 Position of the slots (with side wall on the right removed)

DANGER



Due to the DC link capacitors, hazardous voltages are still present in the converter up to 5 minutes after it has been disconnected from the power supply. Before opening the converter, wait until the capacitors have completely discharged.

8.4.2.2 Mounting positions of the CBC in Compact and chassis units of function classes MC (CUPM, CUMC) and VC (CUVC)

Slots

In the electronics box of the compact-type and chassis-type converters and inverters, there are up to six slots available for installing an optional board. The slots are marked with the letters A to G. Slot B is not present in these types of unit; it is used in Compact PLUS units.

If you wish to use Slots D to G, you must first mount the LBA (Local Bus Adapter) and the corresponding adapter board (6SX7010-0KA00).

NOTE

In principle, you can operate the optional CBC board (Communication Board CAN) in any slot. Please bear in mind, however, that an encoder board always needs Slot C and that the LBA requires the slots to be used in a particular sequence.

The CBC can be mounted on the adapter board in both slots, i.e. TOP and/or BOTTOM.

Position of the slots

The slots are located at the following positions:

◆ Slot A	CU board	Top
◆ Slot C	CU board	Bottom
◆ Slot D	Adaptation board in mount. pos. 2	Top
◆ Slot E	Adaptation board in mount. pos. 2	Bottom
◆ Slot F	Adaptation board in mount. pos. 3	Top
◆ Slot G	Adaptation board in mount. pos. 3	Bottom

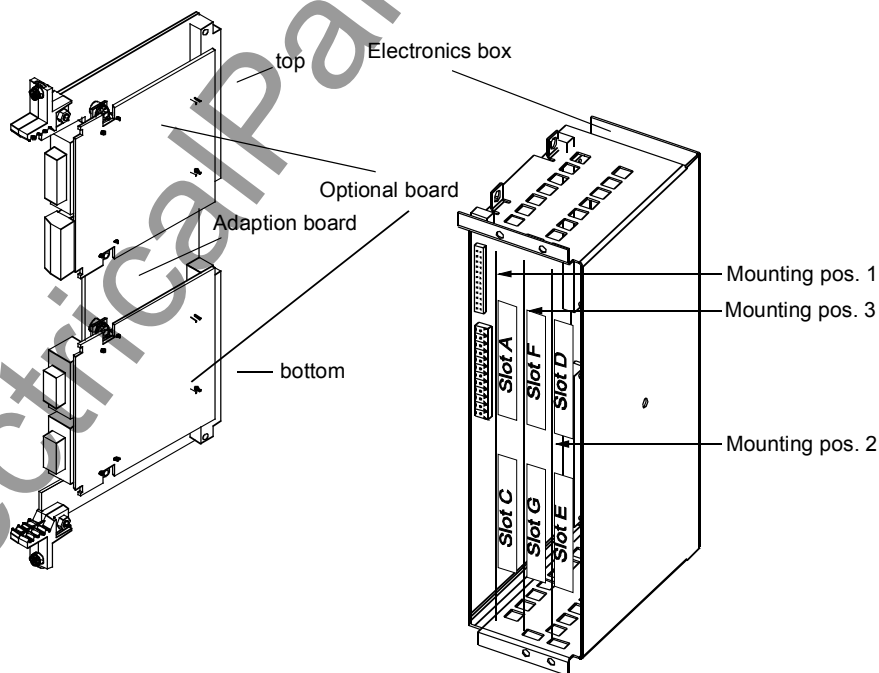


Fig. 8.4-5 Adaptation board with optional boards and position of the slots for Compact units and chassis-type units

DANGER

Due to the DC link capacitors, hazardous voltages are still present in the converter up to 5 minutes after it has been disconnected from the power supply. Before opening the converter, wait until the capacitors have completely discharged.

For technical reasons, certain sequences for using the slots are stipulated for the LBA.

If only one adaptation board with optional boards is inserted in the electronics box, it must always be inserted in slot +1.B2 (ON THE RIGHT), i.e. mounting position 2.

If a T100 / T300 or T400 technology board is plugged into the electronics box in addition to the adapter board with CBC, the technology board must be plugged into position +1.B2. In this case, the CBC is plugged into position +1.B3.

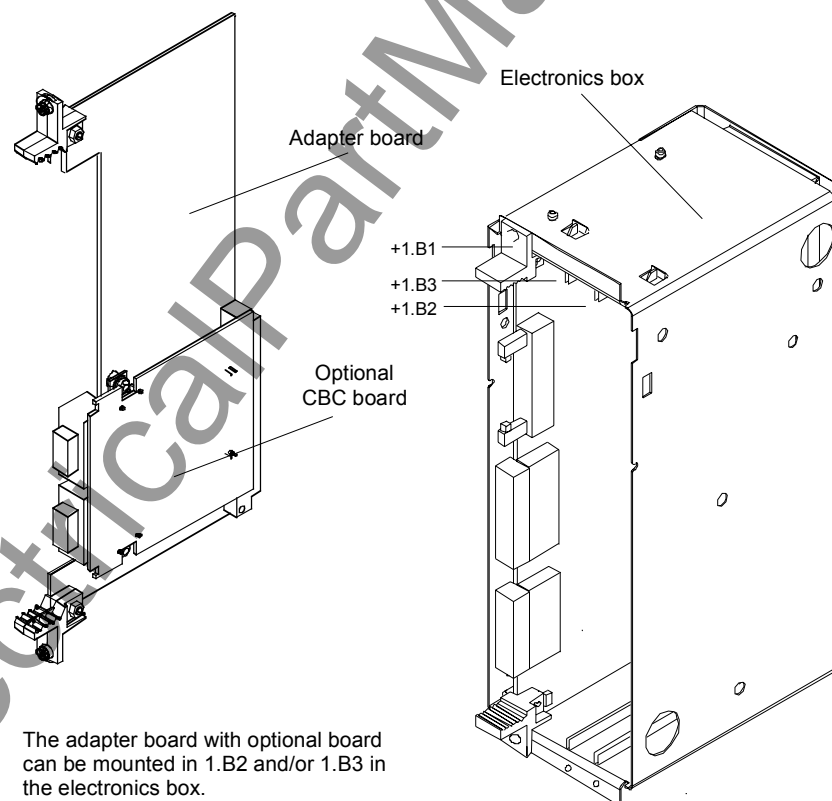
8.4.2.3
Mounting positions of the CBC in Compact type and chassis type units with the CU of the function classes FC (CU1), VC (CU2) or SC (CU3)


Fig. 8.4-6 Electronics box with free slots (+1.B2 and +1.B3) and adapter board with CBC

On the adapter board, **only one** CBC may be mounted in position X 198, i.e. BOTTOM.

In order to mount the CBC with adapter board, the LBA (Local Bus Adapter) backplane adapter must first be mounted.

NOTE

If only one optional board is used, it must always be plugged in position +1.B2 (RIGHT) in the electronics box.

If, in addition to the CBC, a technology board (T100 / T300 or T400) is plugged into the electronics box, the technology board must be plugged into position +1.B2. In this case, the CBC is plugged into position +1.B3.

8.4.2.4 Mounting positions of the CBC in VC Compact PLUS units

NOTE

In principle, the optional CBC board (Communications Board CAN) can be mounted in any slot.

Position of the slots

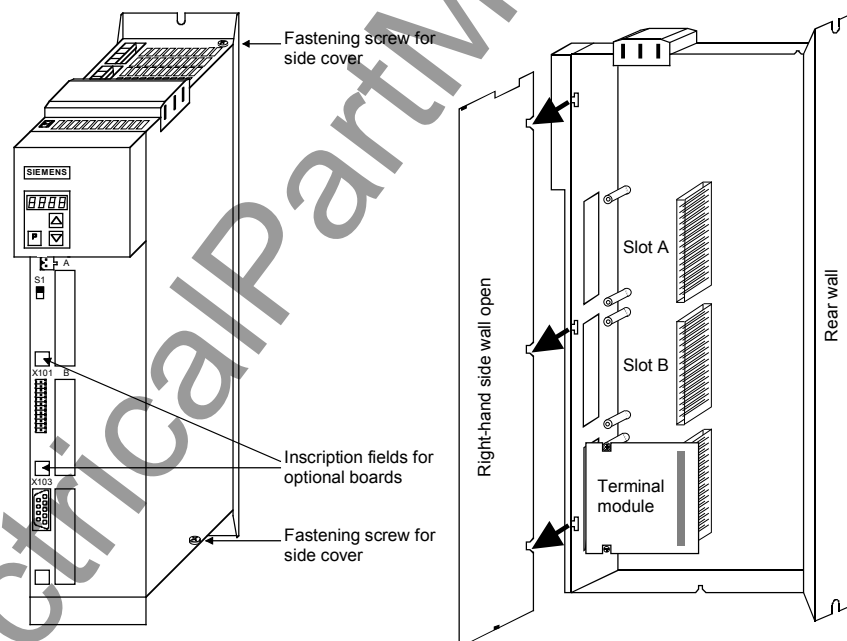


Fig. 8.4-7 Position of the slots (with side wall on the right removed)

DANGER



Due to the DC link capacitors, hazardous voltages are still present in the converter up to 5 minutes after it has been disconnected from the power supply. Before opening the converter, wait until the capacitors have completely discharged.

8.4.3 Connecting

DANGER

The SIMOVERT MASTERDRIVES are operated with high voltages. Any work on the unit may only be carried out by qualified personnel. If this warning is ignored, serious bodily injury or considerable damage to property can occur as a consequence.

Because of the DC link capacitors, there continues to be dangerous voltage in the unit until up to 5 minutes after disconnection. The unit must not therefore be opened until at least this length of time has expired.

Even when the motor is at a standstill, the power terminals and the control terminals can carry voltage. During work on the converter, it is to be disconnected from supply.

When handling the opened converter, it must be kept in mind that live components are exposed.

NOTICE

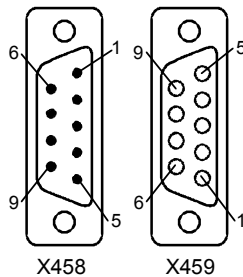
The CBC contains electrostatically sensitive components. These components can very easily be destroyed by improper handling.

8.4.3.1 Connection of the bus cable

The optional CBC board has a 9-pole Sub-D connector (X458) and a 9-pole Sub-D socket (X459) which are provided for connection to the CAN bus.

Both terminals are identically assigned and connected through internally. They are also short-circuit proof and floating.

X458, X459



Pin	Designation	Significance
1	-	Not assigned
2	CAN_L	CAN_L bus cable
3	CAN_GND	CAN ground (ground M5)
4	-	Not assigned
5	-	Not assigned
6	CAN_GND	CAN ground (ground M5)
7	CAN_H	CAN_H bus line
8	-	Not assigned
9	-	Not assigned

Table 8.4-1 Terminals X458 (pins) and X459 (socket)

The two Sub-D connectors X458 and X459 are identically assigned and all conductors are connected through internally.

The bus cable must have at least four cores, stranded in pairs, with a wave resistance of 120 ohms, e.g. the PYCYM wiring cable from SIEMENS.

Order No.: 5DV5 002 PYCYM 2 x 2 x 0.6

As a plug, the Sub-D connector SBM 383 from SIEMENS is recommended:

Connector components	Order No.
9-pole male connector	V42254-A1115-A209
9-pole female connector	V42254-A1115-B209
Housing (shielded)	V42254-A6000-G109
Knurled-head screw for screw interlocking	V42254-A112-V009

Mounting the bus cable

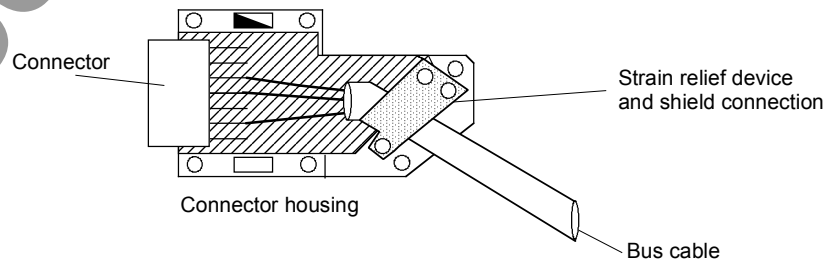


Fig. 8.4-8 Connecting the bus cables

- ◆ When stripping the insulation off the shield, make sure that the shield is not damaged!
- ◆ When stripping the insulation off the core ends, make sure that the copper core is not damaged!

Data transfer rate	Max. cable length (in m)
10 kBit/s	1000
20 kbit/s	1000
50 kBit/s	1000
100 kBit/s	750
125 kBit/s	530
250 kBit/s	270
500 kBit/s	100
800 kBit/s	20
1 Mbit/s	9

Table 8.4-2 Cable length in relation to the baud rate

8.4.3.2 EMC measures

For fault-free CAN bus operation, the following measures are necessary:

Shielding

NOTICE

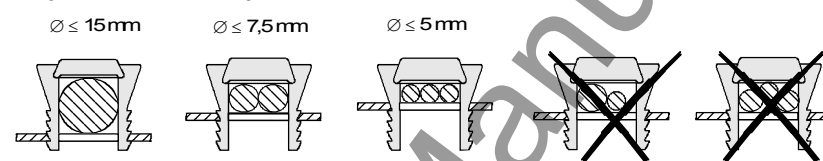
The bus cables must be twisted and shielded and are to be routed separately from power cables, the minimum clearance being 20 cm. The shield must be connected through the largest possible surface area on both sides, i.e. the shield of the bus cable between 2 converters must be connected to the converter housing or the connector housing at **both** ends. The same applies to the shield of the bus cable between the CAN bus master and the converter.

If bus and power cables intersect, they must do so at an angle of 90 °.

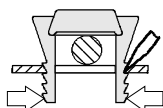
With regard to the CAN bus, there are two ways of attaching the shield:

1. Attaching the shield with the help of shield clamps:
The shield of the bus cable can be attached to the converter housing with the help of shield clamps (Compact units) or shield clamps and cable ties (chassis units). How to use the shield clamps is shown in Fig. 8.4-8 and Fig. 8.4-9. In this case, the shield must not be exposed in the bus connector at the CBC but at the converter housing (see Fig. 8.4-10).
2. Attaching the shield in the connector housing:
The shield of the bus cable can be connected to the shield of the connector housing and is then connected to the CBC board via the connector and to ground as well (see Fig. 8.4-7).

Snap in the shield clamp



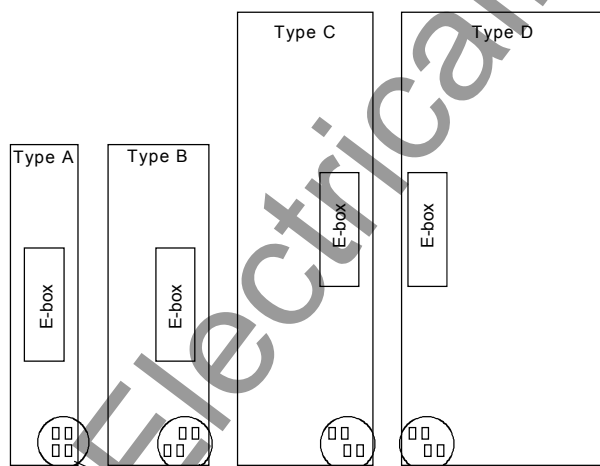
Release the shield clamp



Squeeze the shield clamp together with your hand or a screwdriver and pull upwards.

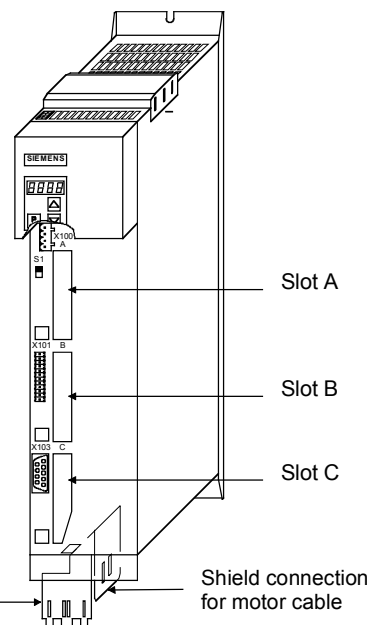
Fig. 8.4-9 Using the shield clamps

Compact type and chassis type units



Shield connecting points

Compact PLUS



Shield connection for control cables

Shield connection for motor cable

Fig. 8.4-10 Position of the shield connecting points

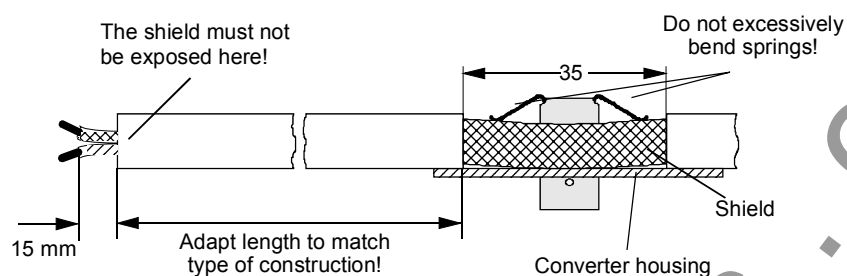


Fig. 8.4-11 Removing insulation from the cable when shield clamps are used

Potential equalization

- ◆ Please avoid differences in potential (e.g. as a result of different power supply levels) between the converters and the PROFIBUS-DP master.
- ◆ Use equipotential bonding cables:
 - 16 mm² Cu equipotential bonding cables up to 200 m
 - 25 mm² Cu equipotential bonding cables over 200 m
- ◆ Route the equipotential bonding cables so that there is the smallest possible surface between the equipotential bonding cables and signal cables.
- ◆ Connect equipotential bonding cables to the ground/protective conductor through the largest possible surface area.

Laying cables

Please comply with the following instructions when laying cables:

- ◆ Do not lay bus cables (signal cables) directly parallel to power cables.
- ◆ Lay signal cables and the associated equipotential bonding cables with the lowest possible distance between them and on the shortest routes.
- ◆ Lay power cables and signal cables in separate cable ducts.
- ◆ Attach shields through a large surface area.

8.4.3.3 Bus termination of the CAN bus (jumper S1.2)

For fault-free operation of the CAN bus, the bus cable must be terminated with bus terminating resistors at both ends (see Fig. 8.4-11). The bus cable from the first CAN bus node up to the last CAN bus node is to be regarded as **one** bus cable so that the CAN bus must only be terminated twice.

The bus terminating resistors must be connected into the circuit at the first bus node (e.g. the master) and the last bus node (e.g. the slave). If the bus-terminating node is a CBC, please close jumper S1.2 of the DIP-FIX switch, S1, on the CBC board!

NOTE

Please ensure that the bus termination is only connected in the circuit at the first bus node and the last bus node (e.g. CBC)!

Jumper	Function	As supplied
S1.2	Bus termination X458/459	Open (no bus termination)

Table 8.4-3 Bus termination with switch S1

8.4.3.4 Ground connection (jumper S1.1)

Jumper S1.1 normally remains open. If the CAN bus interface of the master is operated as a ground-free interface, you can close jumper S1.1 on one converter in order to connect the bus to ground.

Jumper	Function	As supplied
S1.1	Ground connection, interface ground (X458/459)	Open (no bus termination)

Table 8.4-4 Ground connection with switch S1

NOTE

For fault-free operation of the CAN bus, the bus cable must be terminated with bus terminating resistors at both ends. The bus cable from the first CAN bus node up to the last CAN bus node is to be regarded as one bus cable so that the CAN bus must only be terminated twice.

Switch S1.2 of the bus terminating resistors is located on the optional board behind connector X458.

NOTE

If the CAN bus interface of the master is to be operated ground-free, you can close switch S1.1 at one node in order to connect the bus to ground.

The switch for ground connection is located on the optional board behind connector X458.

8.4.3.5 Interface X458 / X459 with jumper strip S1

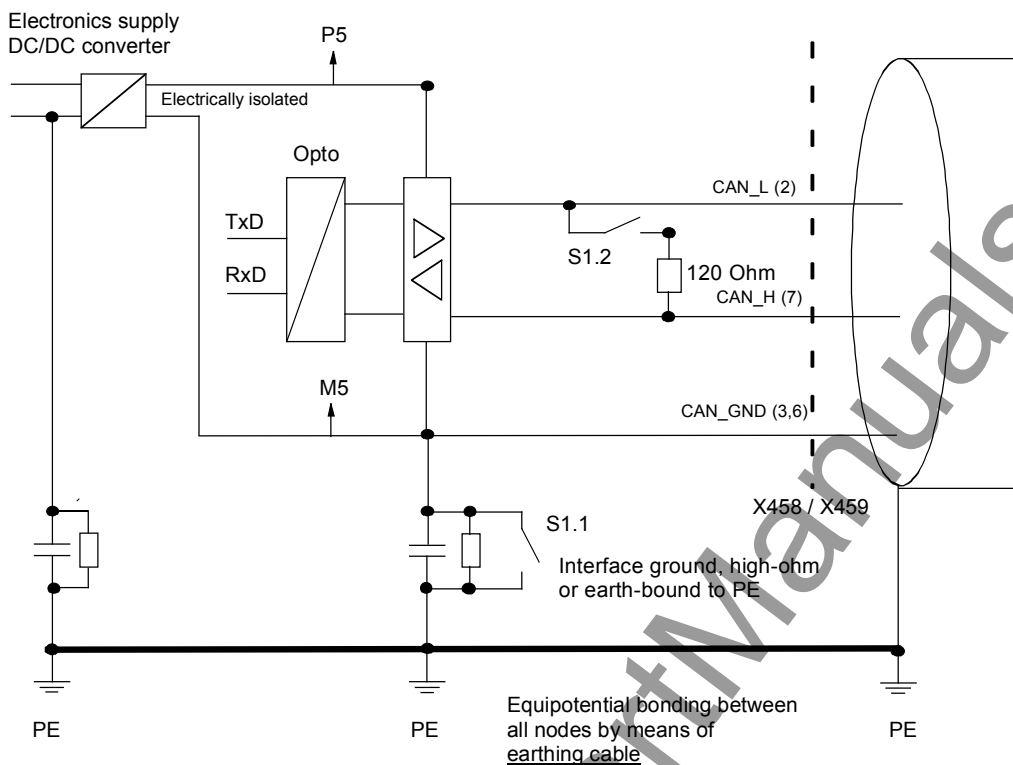


Fig. 8.4-12 Function of jumper strip S1

8.4.3.6 Recommended circuits

Replacing the CBC with bus interruption

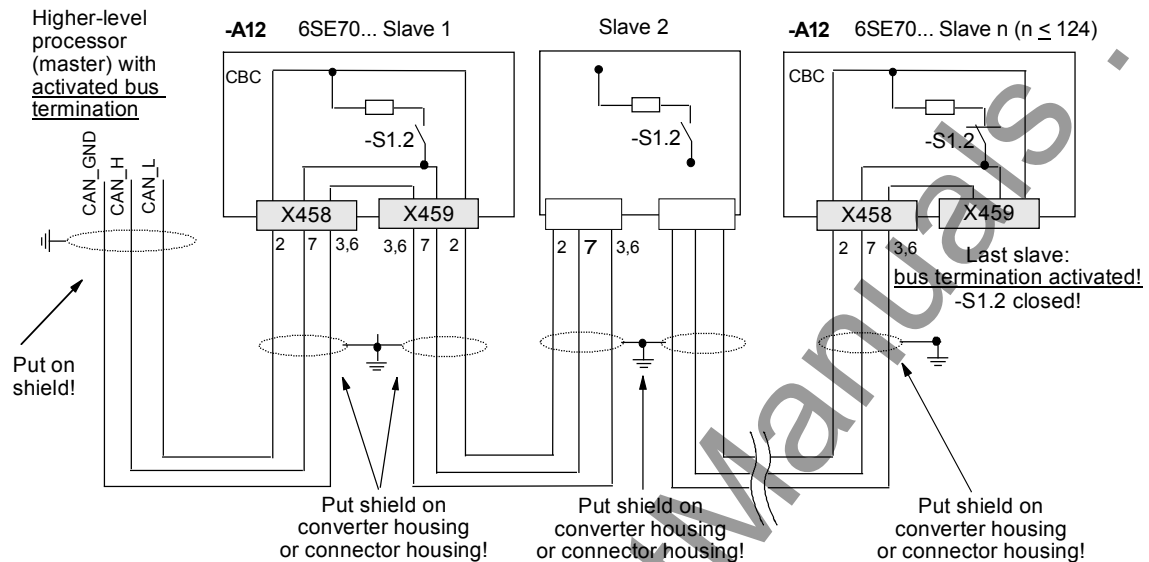


Fig. 8.4-13 Bus connection interrupted when connector X458 or X459 is pulled out

Replacing the CBC without bus interruption

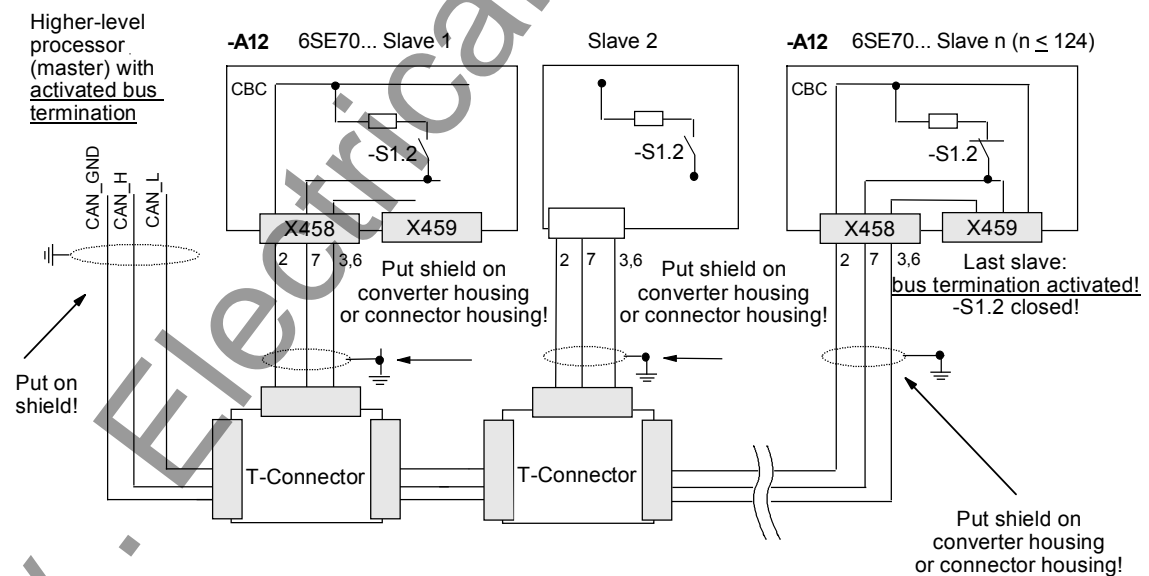


Fig. 8.4-14 Bus connection not interrupted when connector X458 is pulled out

8.4.4 Data transfer via the CAN bus

8.4.4.1 General

With regard to the transfer of net (useful) data, a distinction is made between parameter data (PKW data) and process data (PZD data) (see also Section 8.4.1 "Product description").

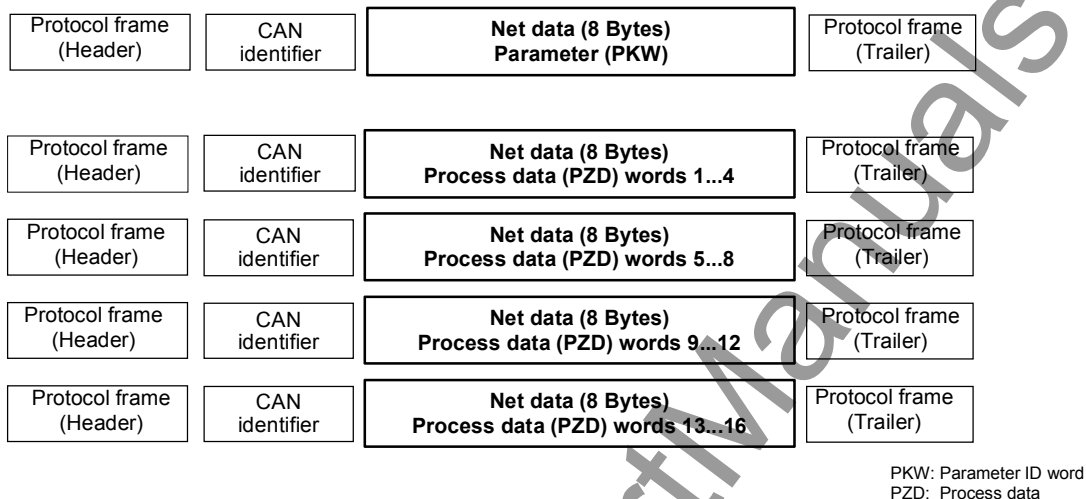


Fig. 8.4-15 Structure of the net data in the telegrams of the CAN protocol

A CAN data telegram consists of the protocol header, the CAN identifier (ID), up to 8 bytes of net data and the protocol trailer.

The CAN identifier is used for unambiguously identifying the data telegram. In the standard message format, a total of 2048 different CAN identifiers are possible and, in the extended message format, 2^{29} CAN identifiers. The extended message format is tolerated by the CBC but not evaluated.

The CAN identifier also specifies the priority of the data telegrams. The lower the number of the CAN identifier, the higher its priority. If two or more bus nodes want to send data telegrams at the same time, the CAN telegram with the lowest CAN identifier and thus the highest priority is accepted.

A maximum of 8 bytes of net data can be transferred in a CAN data telegram. The PKW area always consists of 4 words or 8 bytes, i.e. the data can be transferred in a single data telegram.

In contrast to this, the process data area for MASTERDRIVES consists of 16 words, i.e. a total of 4 data telegrams are needed to transfer all possible process data.

8.4.4.2 Parameter area (PKW)

With the PKW mechanism, you can perform the following tasks:

- ◆ reading parameters
- ◆ writing parameters
- ◆ reading the parameter description
(parameter type, max./min. value, etc.)

The parameter area is always composed of 4 words.

1st word:

Parameter ID (PKE)				
Byte 1			Byte 0	
15	12	11	10	0
AK		SPM	PNU	

2nd word:

Parameter Index (IND)				
Byte 3			Byte 2	
15	8		7	0
Bit 15 = PARA PAGE SEL			Index	

3rd word:

Parameter value (PWE)	
Byte 5	Byte 4
Parameter value Low (PWE1)	
Byte 7	Byte 6
Parameter value High (PWE2)	

4th word:

AK: Task or reply ID

SPM: Toggle bit for processing the parameter change report (not supported by the CBC)

PNU: Parameter number

Parameter ID (PKE) The parameter ID (PKE) is **always** a 16-bit value.

Bits 0 to 10 (PNU) contain the number of the required parameter. The meaning of the parameters can be found in the section, "Parameter list", of the converter operating instructions.

Bit 11 (SPM) is the toggle bit for parameter change reports.

NOTE

Parameter change reports are not supported by the CBC.

Bits 12 to 15 (AK) contain the task reply ID.

The meaning of the task ID for the task telegram (master → converter) is shown in Table 8.4-5. It corresponds to the specifications in the "PROFIBUS profile for variable-speed drives". Task IDs 10 to 15 are specific to SIMOVERT MASTERDRIVES and are not defined in the PROFIBUS profile.

The meaning of the reply ID for the reply telegram (converter → master) is shown in Table 8.4-6. This also corresponds to the specifications in the "PROFIBUS profile for variable-speed drives". Reply IDs 11 to 15 are specific to SIMOVERT MASTERDRIVES and are not defined in the PROFIBUS profile. If the reply ID has the value 7 (task cannot be executed), an error number is placed in parameter value 1 (PWE1).

Task ID	Meaning	Reply ID	
		positive	negative
0	No task	0	7 or 8
1	Request parameter value	1 or 2	↑
2	Change parameter value (word) for non-indexed parameters	1	
3	Change parameter value (double word) for non-indexed parameters	2	
4	Request descriptive element ¹	3	
5	Change descriptive element (not with the CBC)	3	
6	Request parameter value (array) ¹	4 or 5	
7	Change parameter value (array, word) for indexed parameters ²	4	
8	Change parameter value (array, double word) for indexed parameters ²	5	
9	Request number of array elements	6	
10	Reserved	-	
11	Change parameter value (array, double word) and store in the EEPROM ²	5	
12	Change parameter value (array, word) and store in the EEPROM ²	4	
13	Change parameter value (double word) and store in the EEPROM	2	
14	Change parameter value (word) and store in the EEPROM	1	↓
15	Read or change text (not with the CBC)	15	7 or 8

¹ The required element of the parameter description is specified in IND(2nd word)

² The required element of the indexed parameter is specified in IND(2nd word)

Table 8.4-5 Task ID (master → converter)

Reply ID	Meaning
0	No reply
1	Transfer parameter value in the case of non-indexed parameters (word)
2	Transfer parameter value in the case of non-indexed parameters (double word)
3	Transfer descriptive element ¹
4	Transfer parameter value (array, word) in the case of indexed parameters ²
5	Transfer parameter value (array, double word) in the case of indexed parameters ²
6	Transfer number of array elements
7	Task cannot be executed (with error number)
8	No operator change rights for the PKW interface
9	Parameter change report (word) (not with the CBC)
10	Parameter change report (double word) (not with the CBC)
11	Parameter change report (array, word) ² (not with the CBC)
12	Parameter change report (array, double word) ² (not with the CBC)
13	Reserved
14	Reserved
15	Transfer text (not with the CBC)

¹ The required element of the parameter description is specified in IND (2nd word)

² The required element of the indexed parameter is specified in IND (2nd word)

Table 8.4-6 Reply ID (converter -> master)

Example of parameter identifier

Source for the ON/OFF command (control word 1, bit 0): P554 (=22A Hex)

Change parameter value (array, word) and store in the EEPROM.

1st word

Bit No.:

Binary value

HEX value

Parameter ID (PKE)															
15				12		11		10				0			
AK				SPM				PNU							
Byte 1								Byte 0							
1				1		0		0		0		0		1	
C				2				2				A			

Bits 12 to 15: Value = 12 (= "C" Hex); change parameter value (array, word) and store in the EEPROM

Bits 0 to 11: Value = 554 (= "22A" Hex); parameter number without a set parameter change report bit

Error numbers in the case of reply "Task cannot be executed" Error numbers in the case of reply "Task cannot be executed" (converter parameters).
The error numbers are transferred in the 3rd word (PWE1) of the reply.

No.	Meaning	
0	Non-permissible parameter number (PNU)	If there is no PNU
1	Parameter value cannot be changed	If the parameter is a visualization parameter
2	Upper or lower limit exceeded	–
3	Erroneous subindex	–
4	No array	In the case of tasks for indexed parameters, to a non-indexed parameter e.g. Task: 'Change parameter value (word, array)' for non-indexed parameter
5	Incorrect data type	–
6	Setting not allowed (can only be reset)	–
7	Descriptive element cannot be altered	Task never possible with MASTERDRIVES
11	No operator control rights	–
12	Key word missing	Converter parameter: 'Access key' and/or 'Parameter special access' not correctly set
15	No text array present	
17	Task cannot be executed because of operating status	Converter status does not permit the set task at the moment
101	Parameter number deactivated at present	
102	Channel width too small	Parameter reply too long for the CAN telegram
103	PKW: number incorrect	<i>Cannot occur with the CBC</i>
104	Parameter value not admissible	–
105	The parameter is indexed	In the case of tasks for non-indexed parameters, to an indexed parameter e.g. Task: 'PWE, change word' for indexed parameter
106	Task not implemented	–

Comment on error number 102:

This error number is transferred if the parameter reply to a parameter task is longer than the available 8 bytes of the CAN data telegram and therefore cannot be transferred. The data are not divided up to create several telegrams.

Comment on error number 104:

This error number is transferred if, in the converter, no function has been assigned to the parameter value which is to be adopted or if the value cannot be accepted at the time of the change for internal reasons (even though it is within the limits).

Table 8.4-7 Error numbers in the case of reply "Task cannot be executed" (converter parameter)

Example

The parameter 'PKW number' for the G-SST1 (number of net data in the PKW channel):

Minimum value: 0 (0 words)

Maximum value: 127 (corresponds to variable length)

Permissible values for USS: 0, 3, 4 and 127

If a change task with a PWE other than 0, 3, 4 or 127 is sent to the converter, the reply is: 'Task cannot be executed' with error value 104.

Parameter index (IND) 2nd word

The index is an 8-bit word and is always transferred over the CAN bus in the low byte (bits 0 to 7) of the parameter index (IND). The high byte (bits 8 to 15) of the parameter index (IND) contains the parameter page selection bit (bit 15).

The parameter page selection bit acts as follows:

If this bit = 1, the parameter number (PNU) transferred in the PKW request is given an offset of 2000 in the CBP and then passed on.

Parameter designation (as per parameter list)	Serial parameter number	Parameter addresses via PROFIBUS		
		PNU [decimal]	PNU [hex.]	Bit *)
P000 - P999 (r000 - r999)	0 - 999	0 - 999	0 - 3E7	= 0
H000 - H999 (d000 - d999)	1000 - 1999	1000 - 1999	3E8 - 7CF	= 0
U000 - U999 (n000 - n999)	2000 - 2999	0 - 999	0 - 3E7	= 1
L000 - L999 (c000 - c999)	3000 - 3999	1000 - 1999	3E8 - 7CF	= 1

*) Parameter page selection

In the case of an indexed parameter, the required index is transferred. The meaning of the indices can be found in the section, "Parameter list", of the instruction manual for the converter.

In the case of a descriptive element, the number of the required element is transferred. The meaning of the descriptive elements can be found in the "PROFIBUS profile for variable-speed drives" (VDI/VDE 3689).

**Example
Parameter index**

Source for the ON/OFF1 command (control word 1, bit 0):

P554 (=22A Hex)

Change parameter value of index 1.

2nd word	Parameter index (IND)			
	15		8 7	
Bit No.:				
	Byte 3		Byte 2	
Binary value	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 1
HEX value	0	0	0	1

Bits 8 to 15: Bit 15 parameter page selection bit

Bits 0 to 7: Index or number of the descriptive element

Parameter value (PWE) 3rd and 4th word

The parameter value (PWE) is **always** transferred as a double word (32 bits). **Only one** parameter value can ever be transferred in a telegram.

A 32-bit parameter value is composed of PWE1 (least significant word, 3rd word) and PWE2 (most significant word, 4th word).

A 16 bit parameter value is transferred in PWE1 (least significant word, 3rd word). In this case, you must set PWE2 (most significant word, 4th word) to the value 0.

Example Parameter value

Source for the ON/OFF1 command (control word 1, bit 0):

P554 (=22A Hex)

Change parameter value of index 1 to the value 3100.

Parameter value (PWE)	
3rd word (PWE1)	Byte 5 Byte 4
Bit No.:	15 8 7 0
HEX value	3 1 0 0

4th word (PWE2)	Byte 7 Byte 6
Bit No.:	31 24 23 16
HEX value	0 0 0 0

Bits 8 to 15: Parameter value in the case of 16-bit parameter or low component in the case of 32-bit parameter

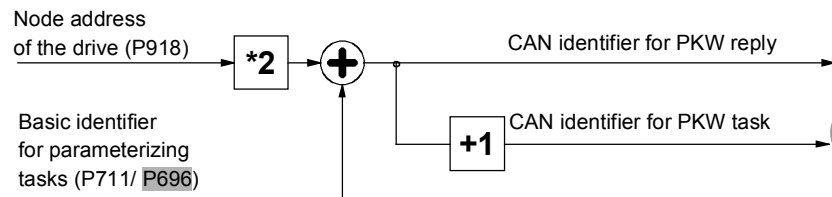
Bits 16 to 31: Value = 0 in the case of 16-bit parameter or high component in the case of 32-bit parameter

CAN identifiers for parameter processing

Two unambiguous CAN identifiers are needed for parameter processing, one for the PKW task and one for the PKW reply. In contrast to other protocols, the CAN protocol only recognizes identifiers and not node addresses. Practical experience shows, however, that it is useful to define node addresses here as well for reasons of clarity. For parameter processing, the individual CAN identifiers of the drive can thus be generated from the node address (P918 "CB bus address") and the basic identifier value (P711 / P696 "CB parameter 1").

NOTE

A parameter on a gray background is only valid for MASTERDRIVES with CU1, CU2 or CU3.



- ◆ CAN identifier for the parameter task (PKW task):
(value in P711 / P696) + (value in P918)*2
- ◆ CAN identifier for the parameter reply (PKW reply):
(value in P711 / P696) + (value in P918)*2 + 1

In addition to the PKW task, a PKW task broadcast is possible, i.e. a parameter task is simultaneously processed by all bus nodes. The CAN identifier for this is set in parameter P719 / P704 "CB parameter 9". The node address does not go in here as the task is to be processed by all slaves. The associated parameter reply is made with the regular CAN identifier for the PKW reply as described above.

Example

Parameter-value processing, i.e. the reading and writing of parameter values of the drives, is to take place in the whole CAN network, from identifier 1000 onwards.

Specification of the identifiers for PKW task and PKW reply:

Drive with node address 0:

1. P711 / P696 = 1000 (PKW basic identifier)
 2. P918 = 0 (node address)
- PKW task ID = 1000 PKW reply ID = 1001

Drive with node address 1:

1. P711 / P696 = 1000 (PKW basic identifier)
 2. P918 = 1 (node address)
- PKW task ID = 1002 PKW reply ID = 1003
and so on.

Rules for task/reply processing

- ◆ The length of the task or reply is always 4 words.
- ◆ The least significant byte (in the case of words) or the least significant word (in the case of double words) is always sent first.
- ◆ **One** task or **one** reply can only relate to **one** parameter value.
- ◆ The slave does not send the reply to a parameter task until the data are received from the MASTERDRIVES unit.
During normal operation, this lasts 20 to 150 ms, depending on the type of MASTERDRIVES unit.
- ◆ In certain states of the converters (especially in initialization states), parameter processing is not carried out at all or only with a long delay. Here, a delay of up to 40 seconds can be expected for the reply.
- ◆ The master can only issue a new parameter task after receiving the reply to a previously issued task.
- ◆ The master identifies the reply to a task which has been set:
 - By evaluating the reply ID
 - By evaluating the parameter number, PNU
 - If necessary, by evaluating the parameter index, IND
 - If necessary, by evaluating the parameter value, PWE.
- ◆ The task must be sent complete in one telegram; telegrams with split tasks are not permissible. The same applies to the reply.

8.4.4.3 Process data area (PZD)

Control words and setpoints (tasks: master → converter) or status words and actual values (replies: converter → master) can be transferred by means of the process data.

The transferred process data only come into effect if the used bits of the control words, setpoints, status words and actual values have been routed (softwired) to the dual-port RAM interface.

For softwiring of the PZD, the number *i* of the process data (PZDi, *i* = 1 to 16) is entered in the connection value.

NOTE

The process-data connection as described here does not apply if a technology board has been mounted.

If a technology board (e.g. T300, T100) is used, the process-data connection is indicated the technology board manual.

Telegram: master → converter (Setpoint channel)	PZD receive															
	PZD 1 STW1	PZD 2 HSW	PZD 3	PZD 4	PZD 5	PZD 6	PZD 7	PZD 8	PZD 9	PZD 10	PZD 11	PZD 12	PZD 13	PZD 14	PZD 15	PZD 16
	1st word	2nd word	3rd word	4th word	5th word	6th word	7th word	8th word	9th word	10th word	11th word	12th word	13th word	14th word	15th word	16th word
Connectors for:																
16-bit process data	3001	3002	3003	3004	3005	3006	3007	3008	3009	3010	3011	3012	3013	3014	3015	3016
16-/32-bit PZDs	3001	3032	3034	3006	3037	3039	3041	3043	3045							
(Example)	3001	3032	3004	3005	3036	3038	3040	3042	3044	3016						
See Section 8.4.5.2	3001	3002	3033	3035	3007	3038	3010	3041	3013	3044	3016					

Telegram: converter → master (actual-value channel)	PZD send															
	PZD 1 ZSW	PZD 2 HIW	PZD 3	PZD 4	PZD 5	PZD 6	PZD 7	PZD 8	PZD 9	PZD 10	PZD 11	PZD 12	PZD 13	PZD 14	PZD 15	PZD 16
	1st word	2nd word	3rd word	4th word	5th word	6th word	7th word	8th word	9th word	10th word	11th word	12th word	13th word	14th word	15th word	16th word
Connectors for:																
Assignment of actual-value parameters in the case of 16-bit process data	P734 P694 i001	P734 P694 i002	P734 P694 i003	P734 P694 i004	P734 P694 i005	P734 P694 i006	P734 P694 i007	P734 P694 i008	P734 P694 i009	P734 P694 i010	P734 P694 i011	P734 P694 i012	P734 P694 i013	P734 P694 i014	P734 P694 i015	P734 P694 i016
16-/32-bit process data (examples)	P734 P694 i001	P734 P694 i002 = i003	P734 P694 i004 = i005	P734 P694 i006	P734 P694 i007 = i008	P734 P694 i009 = i010	P734 P694 i011 = i012	P734 P694 i013 = i014	P734 P694 i015							
See also Section 8.4.5.2	P734 P694 i001	P734 P694 i002	P734 P694 i003 = i004	P734 P694 i005 = i006	P734 P694 i007	P734 P694 i008 = i009	P734 P694 i010	P734 P694 i011 = i012	P734 P694 i013	P734 P694 i014 = i015	P734 P694 i016					

PZD: Process data
STW: Control word
ZSW: Status word

HSW: Main setpoint
HIW: Main actual value

Table 8.4-8 Permanently specified assignments and connectors

NOTE

A parameter on a gray background is only valid for MASTERDRIVES with CU1, CU2 or CU3.

CAN identifiers for process-data processing

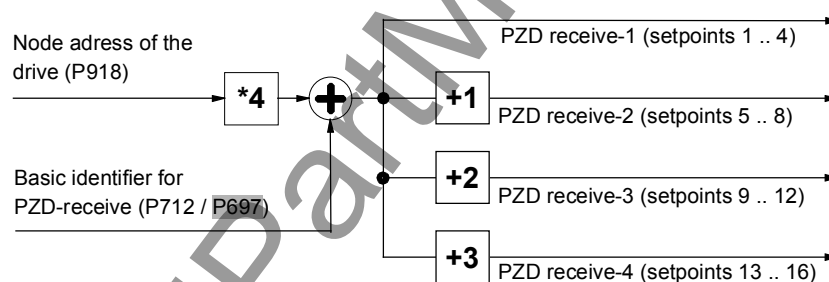
Basic process-data processing consists of the two functions, "Receiving process-data" (PZD receive) and "Sending process-data" (PZD send). A total of 16 process-data words are possible from the MASTERDRIVES units, both in the receive and in the send direction. For each direction, therefore, a total of 4 CAN messages are needed because each individual CAN telegram can only transfer 4 process-data words. This means that 4 unambiguous CAN identifiers are needed both for PZD send and PZD receive. As in parameter processing, node addresses and a basic identifier are also defined in order to achieve better communication.

NOTE

A parameter on a **gray background** is only valid for MASTERDRIVES with CU1, CU2 or CU3.

PZD receive

For the PZD receive function, the same PZD-receive basic identifier is set for all units on the bus by means of CB parameter P712 / P697, "CB parameter 2". Unique identification is achieved by means of the node address in parameter P918, "CB bus address", which must be different for each bus node. A total of 4 CAN identifiers are assigned.



CAN identifier for the 1st PZD-receive CAN telegram (words 1 to 4):
(value in 712 / P697) + (value in P918)*4

CAN identifier for the 2nd PZD-receive CAN telegram (words 5 to 8):
(value in 712 / P697) + (value in P918)*4 + 1

CAN identifier for the 3rd PZD-receive CAN telegram (words 9 to 12):
(value in 712 / P697) + (value in P918)*4 + 2

CAN identifier for the 4th PZD-receive CAN telegram (words 13 to 16):
(value in 712 / P697) + (value in P918)*4 + 3

Example

PZD-receive processing, i.e. the receiving of control words and setpoints in the whole CAN network, is to take place from identifier 200 onwards. Control word 1 is received in the 1st word, a 32-bit main setpoint in the 2nd and 3rd words, control word 2 in the 4th word and an additional setpoint in the 5th word.

Specification of the identifiers for PZD receive:

Drive with node address 0:

1. $P712 / P697 = 200$ (PZD-receive basic identifier)
2. $P918 = 0$ (node address)
- PZD-receive 1 = 200 PZD-receive 2 = 201
- PZD-receive 3 = 202 PZD-receive 4 = 203

Drive with node address 1:

1. $P712 / P697 = 200$ (PZD-receive basic identifier)
2. $P918 = 1$ (node address)
- PZD-receive 1 = 204 PZD-receive 2 = 205
- PZD-receive 3 = 206 PZD-receive 4 = 207

and so on.

Connecting the setpoints in the drive:

P443.01 (Source of main setpoint) = 3032

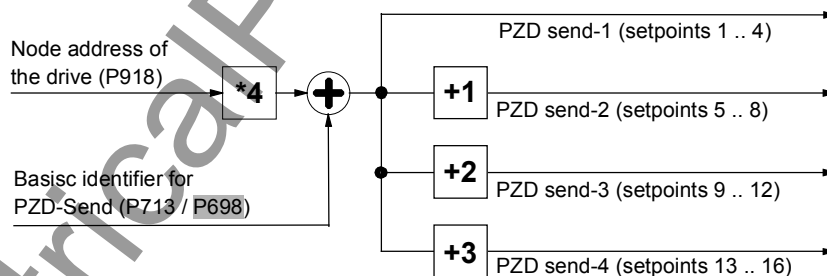
P554.01 (Source of ON/OFF1) = 3100 / 3001 (use of control word 1)

P433.01 (Source of additional setpoint) = 3005

PZD-send

For PZD-send, the same PZD-send basic identifier is set for all units on the bus by means of CB parameter P713 / P698, "CB parameter 3".

The number of CAN identifiers actually assigned and CAN telegrams sent depends on P714 / P699, "CB parameter 4", where the number of words to be sent (between 1 and 16) is specified.



CAN identifier for the 1st PZD-send CAN telegram (words 1 to 4):

(value in P713 / P698) + (value in P918)*4

CAN identifier for the 2nd PZD-send CAN telegram (words 5 to 8):

(value in P713 / P698) + (value in P918)*4 + 1

CAN identifier for the 3rd PZD-send CAN telegram (words 9 to 12):

(value in P713 / P698) + (value in P918)*4 + 2

CAN identifier for the 4th PZD-send CAN telegram (words 13 to 16):

(value in P713 / P698) + (value in P918)*4 + 3

Example

PZD-send processing, i.e. the sending of status words and actual values, is to take place in the whole CAN network from identifier 100 onwards. Control word 1 is sent in the 1st word, the actual speed as a 32-bit value in the 2nd and 3rd words, status word 2 in the 4th word, the output voltage in the 5th word, the output current in the 6th word and the current torque in the 7th word.

Specification of the identifiers for PZD-send:

Drive with bus address 0:

1. P713 / P698 = 100 (PZD-send basic identifier)
 2. P714 / P699 = 7 (number of actual values)
 3. P918 = 0 (node address)
- PZD-send 1 = 100 PZD-send 2 = 101
 (PZD-send 3 = 102 PZD-send 4 = 103)

Drive with node address 1:

1. P713 / P698 = 100 (PZD-send basic identifier)
 2. P714 / P699 = 7 (number of actual values)
 3. P918 = 1 (node address)
- PZD-send 1 = 104 PZD-send 2 = 105
 (PZD-send 3 = 106 PZD-send 4 = 107)

and so on (PZD-send 3 and PZD-send 4 are not sent because the number of actual values (P714 / P699) is only 7)

Connection of the actual values in the drive:

- P734.01 = 32 / P694.01 = 968 (status word 1)
 P734.02 = 151 / P694.02 = 218 (main actual value as a 32-bit value ->)
 P734.03 = 151 / P694.03 = 218 (same connector-/parameter numbers in 2 consecutive indices)
- P734.04 = 33 / P694.04 = 553 (status word 2)
 P734.05 = 189 / P694.05 = 3 (output voltage)
 P734.06 = 168 / P694.06 = 4 (output current)
 P734.07 = 241 / P694.07 = 5 (torque)

**CAN identifiers for
addition process-
data functions**

The PZD-recv-broadcast function is for simultaneously sending setpoints and control information from the master to all slaves on the bus. The CAN identifier must be the same for all slaves which use this function. This CAN identifier is entered by means of P716 / P701, "CB parameter 6".

The CAN identifier for the first PZD-recv-broadcast CAN telegram (words 1 to 4) then corresponds to the contents of P716 / P701.

- ◆ CAN identifier for the 1st PZD-recv-broadcast CAN telegram (words 1 to 4): **(value in P716 / P701)**
- ◆ CAN identifier for the 2nd PZD-recv-broadcast CAN telegram (words 5 to 8): **(value in P716 / P701) + 1**
- ◆ CAN identifier for the 3rd PZD-recv-broadcast CAN telegram (words 9 to 12): **(value in P716 / P701) + 2**
- ◆ CAN identifier for the 4th PZD-recv-broadcast CAN telegram (words 13 to 16): **(value in P716 / P701) + 3**

PZD-receive multicast

The PZD-receive-multicast function is for simultaneously sending setpoints and control information from the master to a group of slaves on the bus. The CAN identifier must be the same for all slaves within this group which use this function. This CAN identifier is entered by means of P717 / P702, "CB parameter 7". The CAN identifier for the first PZD-receive-multicast CAN telegram (words 1 to 4) then corresponds to the contents of P717 / P702.

- ◆ CAN identifier for the 1st PZD-receive-multicast CAN telegram (words 1 to 4): **(value in P717 / P702)**
- ◆ CAN identifier for the 2nd PZD-receive-multicast CAN telegram (words 5 to 8): **(value in P717 / P702) + 1**
- ◆ CAN identifier for the 3rd PZD-receive-multicast CAN telegram (words 9 to 12): **(value in P717 / P702) + 2**
- ◆ CAN identifier for the 4th PZD-receive-multicast CAN telegram (words 13 to 16): **(value in P717 / P702) + 3**

PZD-receive cross

The PZD-receive-cross function is for receiving setpoints and control information from another slave. With this function, process data can be exchanged between the drives without a CAN-bus master being present. The CAN identifier of PZD-receive cross for the receiving slave must be matched to the CAN identifier of PZD-send of the slave which is sending. This CAN identifier is entered by means of P718 / P703, "CB parameter 8". The CAN identifier for the first PZD-receive-cross telegram (words 1 to 4) then corresponds to the contents of P718 / P703.

- ◆ CAN identifier for the 1st PZD-receive-cross CAN telegram (words 1 to 4): **(value in P718 / P703)**
- ◆ CAN identifier for the 2nd PZD-receive-cross CAN telegram (words 5 to 8): **(value in P718 / P703) + 1**
- ◆ CAN identifier for the 3rd PZD-receive-cross CAN telegram (words 9 to 12): **(value in P718 / P703) + 2**
- ◆ CAN identifier for the 4th PZD-receive-cross CAN telegram (words 13 to 16): **(value in P718 / P703) + 3**

Notes and rules for process-data processing

- ◆ The least significant byte (in the case of words) or the least significant word (in the case of double words) is always sent first.
- ◆ **Control word 1** must always be contained in the 1st word of the received setpoints. If control word 2 is needed, this must be in the 4th word.
- ◆ **Bit 10 "Control of drive unit"** must always be set in control word 1, otherwise the new setpoints and control words are not accepted by the converter.
- ◆ The **consistency of the process data** is only ensured within the data of a CAN telegram. If more than four words are needed, they must be split up among several CAN telegrams because only four words can be transferred in a CAN telegram. Because the converter scans the setpoints asynchronously to telegram transfer, it may happen that the first CAN telegram is accepted by the current transfer cycle whereas the second CAN telegram still originates from the old transfer cycle. Related setpoints, therefore, should always be transferred in the same CAN telegram. If this is not possible due to the peculiarities of the installation, consistency can still be ensured by means of bit 10 "Control of drive unit". To do this, a CAN telegram is first sent in which bit 10 of the control word has been deleted. As a result, the setpoints are no longer accepted by the converter. All the CAN telegrams still needed are then sent. Finally, another CAN telegram is sent in which bit 10 of the control word has been set. As a result, all setpoints and control words are accepted in the converter at the same time.
- ◆ The described process-data functions for receiving setpoints and control words (PZD receive, PZD-receive broadcast, PZD-receive multicast and PZD-receive cross) can be used simultaneously. The transferred data overlap each other in the converter, i.e. the 1st word in the CAN telegrams PZD-receive 1, PZD-receive broadcast 1, PZD-receive multicast 1 and PZD-receive cross 1 is always interpreted in the converter as the same control word 1. The best way of combining these possibilities depends on the concrete application.

DANGER

When you change the initialization function of software version V1.3x to V1.40 and higher, or VC firmware from 3.22 to 3.23 and higher, the behavior of the converter also changes (reverting to the behavior of software versions V1.2x and lower again) as follows:

If the electronics supply is switched off on a converter that is in state "READY" and is connected to an automation system via a field bus (PROFIBUS, CAN, DEVICE-NET, or CC-Link), this causes a fault message for this converter in the automation system.

If the automation system nevertheless sends a control word STW1 with valid authorization (bit 10 = 1) and a pending ON command (bit 0 = 1) to this converter, this can cause the converter to switch on and go straight into "OPERATION" state when the electronics supply is connected at the converter.

8.4.5 Start-up of the CBC

NOTE

Please note the basic parameter differences (described below) to units with the old function classes FC (CU1), VC (CU2) and SC (CU3). These parameter numbers are printed on a dark gray background for purposes of distinction.

8.4.5.1 Basic parameterization of the units

Basic parameterization for MASTERDRIVES with CUPM, CUMC or CUVC

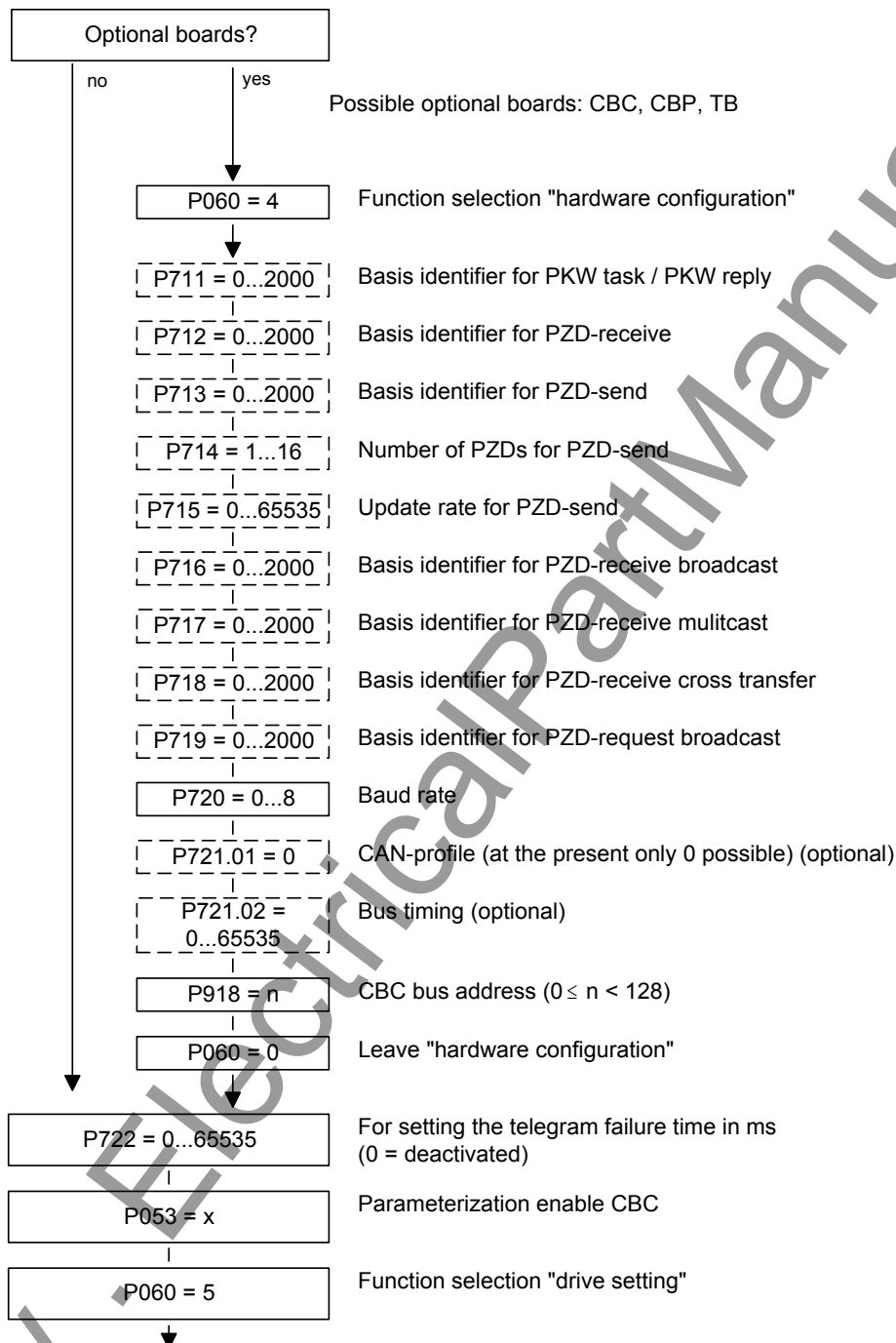


Fig. 8.4-16 Parameterization of the "hardware configuration" for MASTERDRIVES with CUPM, CUMC or CUVC

Basic parameterization for MASTERDRIVES with CU1, CU2 or CU3

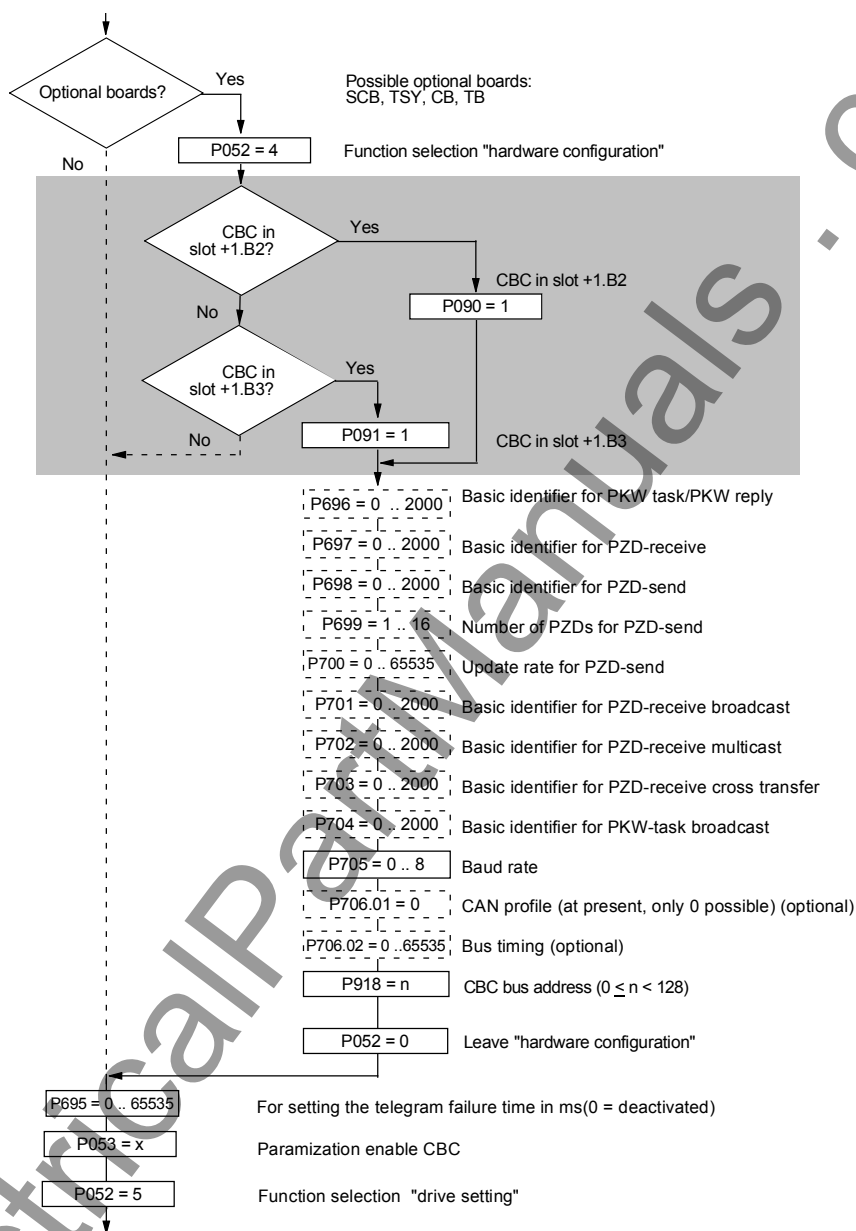


Fig. 8.4-17

Parameterization of the "hardware configuration" for MASTERDRIVES with CU1, CU2 or CU3

P053 (parameterizing enable)

This parameter is significant for the CBC if you wish to set or change parameters of the converter (incl. technology) by means of parameterizing tasks (PKW task or PKW-request broadcast).

In this case, set parameter P053 (see also the parameter list in the instruction manual of the converter) to an odd value (e.g. 1, 3, 7 etc.). With parameter P053, you specify the positions (PMU, CBC etc.) from which it is permissible to change parameters.

E.g.: P053 = 1: Parameterizing enable only CBC
 = 3: Parameterizing enable CBC+PMU
 = 7: Parameterizing enable CBC+PMU+SCom1 (OP)

If the parameter change (= parameterizing enable) is enabled via the CBC (P053 = 1, 3 etc.), all further parameters can be set from the CAN-bus master via the bus.

For further setting of parameters which concern data transfer via the CAN bus (e.g. process-data connection (softwiring)), you must know the number of process-data words received from the slave.

P060**P052**

Function selection "Hardware setting"

P090 (board slot 2) or P091 (board slot 3)

You can alter these parameters even when the CBC is exchanging net data via the CAN bus. You can thus parameterize the CAN-bus interface away from the converter. In this case, the CBC ceases communication via the bus and neither receives nor sends CAN data telegrams.

P711 (CB parameter 1)	P696 (CB parameter 1)
Basic identifier for PKW task (parameter task) With this parameter, the basic identifier can be set for a PKW task (parameter task). The actual CAN identifier for a PKW task is calculated from this parameter and the node address (P918) according to the following equation: $(\text{Parameter value of P711/ P696}) + (\text{Parameter value of P918}) \cdot 2$ The CAN identifier for a PKW reply (parameter reply) is the number subsequent to this, namely $(\text{Parameter value of P711/ P696}) + (\text{Parameter value of P918}) \cdot 2 + 1$ With the value 0 (pre-assigned) in this parameter, parameterization via the CAN bus is deactivated. If the calculated CAN identifier for the PKW task or PKW reply is outside the valid range (1 to 2000) or if it overlaps another CAN identifier, error F080 appears when status 4 "hardware configuration" is left. After acknowledgement of the error, you are in the "hardware configuration" status again and can correct the incorrect parameterization. Example: The basic identifier for parameterization in P711 / P696 is set to 1500. The node address in P918 is 50. The CAN identifier is thus $1500 + 50 \cdot 2 = 1600$ for a PKW task and 1601 for a PKW reply.	

P712 (CB parameter 2)	P697 (CB parameter 2)										
Basic identifier for PZD receive (receiving process data) With this parameter, the basic identifier for PZD receive (receive process data = setpoints / control words) can be set. The actual CAN identifier for PZD receive is calculated from this parameter and the node address (P918) according to the following equation: $(\text{Parameter value of P712} / \text{P697}) + (\text{Parameter value of P918}) \cdot 4$ Because only four setpoints (= 8 bytes) can be transferred with a CAN data telegram but sixteen setpoints are supported by MASTERDRIVES units, a total of four CAN data telegrams with four CAN identifiers are needed for transferring setpoints. The following three CAN identifiers are therefore also provided for PZD receive. The following table applies:											
<table border="1"> <thead> <tr> <th>Contents</th><th>CAN identifier</th></tr> </thead> <tbody> <tr> <td>Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2</td><td>$P712/P697 + P918 \cdot 4$</td></tr> <tr> <td>Setpoint 5 to Setpoint 8</td><td>$P712/P697 + P918 \cdot 4 + 1$</td></tr> <tr> <td>Setpoint 9 to Setpoint 12</td><td>$P712/P697 + P918 \cdot 4 + 2$</td></tr> <tr> <td>Setpoint 13 to Setpoint 16</td><td>$P712/P697 + P918 \cdot 4 + 3$</td></tr> </tbody> </table>		Contents	CAN identifier	Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2	$P712/P697 + P918 \cdot 4$	Setpoint 5 to Setpoint 8	$P712/P697 + P918 \cdot 4 + 1$	Setpoint 9 to Setpoint 12	$P712/P697 + P918 \cdot 4 + 2$	Setpoint 13 to Setpoint 16	$P712/P697 + P918 \cdot 4 + 3$
Contents	CAN identifier										
Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2	$P712/P697 + P918 \cdot 4$										
Setpoint 5 to Setpoint 8	$P712/P697 + P918 \cdot 4 + 1$										
Setpoint 9 to Setpoint 12	$P712/P697 + P918 \cdot 4 + 2$										
Setpoint 13 to Setpoint 16	$P712/P697 + P918 \cdot 4 + 3$										
With the value 0 (pre-assigned) in this parameter, PZD receive is deactivated. If the calculated CAN identifier for PZD receive is outside the valid range (1 to 2000) or if it overlaps another CAN identifier, error F080 appears when status 4 (hardware configuration) is left. After acknowledgement of the error, you are in the "hardware configuration" status again and can correct the incorrect parameterization.											
Example: The basic identifier for PZD receive in P712 / P697 is set to 500. The node address in P918 is 50. This results in a CAN identifier of $500 + 50 \cdot 4 = 700$ for the first CAN data telegram of PZD receive. The further CAN data telegrams for PZD receive have CAN identifiers 701 to 703.											

P713 (CB parameter 3)	P698 (CB parameter 3)										
Basic identifier for PZD-send (sending process data) With this parameter, the basic identifier for PZD-send (sending process data = status words / actual values) can be set. The actual CAN identifier for PZD-send is calculated from this parameter and the node address (P918) according to the following equation: $(\text{Parameter value of P713} / \text{P698}) + (\text{Parameter value of P918}) \cdot 4$ Because only four actual values (= 8 bytes) can be transferred with a CAN data telegram but sixteen actual values are supported by MASTERDRIVES units, a total of four CAN data telegrams with four CAN identifiers are needed for transferring the actual values. The following table applies:											
<table border="1"> <thead> <tr> <th>Contents</th><th>CAN identifier</th></tr> </thead> <tbody> <tr> <td>Status word 1 / Actual value 2 / Actual value 3 / Actual value 4 or Status word 2</td><td>$\text{P713/P698} + \text{P918} \cdot 4$</td></tr> <tr> <td>Actual value 5 to Actual value 8</td><td>$\text{P713/P698} + \text{P918} \cdot 4 + 1$</td></tr> <tr> <td>Actual value 9 to Actual value 12</td><td>$\text{P713/P698} + \text{P918} \cdot 4 + 2$</td></tr> <tr> <td>Actual value 13 to Actual value 16</td><td>$\text{P713/P698} + \text{P918} \cdot 4 + 3$</td></tr> </tbody> </table>		Contents	CAN identifier	Status word 1 / Actual value 2 / Actual value 3 / Actual value 4 or Status word 2	$\text{P713/P698} + \text{P918} \cdot 4$	Actual value 5 to Actual value 8	$\text{P713/P698} + \text{P918} \cdot 4 + 1$	Actual value 9 to Actual value 12	$\text{P713/P698} + \text{P918} \cdot 4 + 2$	Actual value 13 to Actual value 16	$\text{P713/P698} + \text{P918} \cdot 4 + 3$
Contents	CAN identifier										
Status word 1 / Actual value 2 / Actual value 3 / Actual value 4 or Status word 2	$\text{P713/P698} + \text{P918} \cdot 4$										
Actual value 5 to Actual value 8	$\text{P713/P698} + \text{P918} \cdot 4 + 1$										
Actual value 9 to Actual value 12	$\text{P713/P698} + \text{P918} \cdot 4 + 2$										
Actual value 13 to Actual value 16	$\text{P713/P698} + \text{P918} \cdot 4 + 3$										
With the value 0 (pre-assigned) in this parameter, PZD-send is deactivated. If the calculated CAN identifier for PZD-send is outside the valid range (1 to 2000) or if it overlaps another CAN identifier, the error F080 appears when status 4 "hardware configuration" is left. After acknowledgement of the error, you are in the "hardware configuration" status again and can correct the incorrect parameterization. Which values are sent is specified in parameters P713.01 / P694.01 to P713.16 / P694.16 by entering the relevant parameter numbers. Example: The basic identifier for PZD-send in P713 / P698 is set to 200. The node address in P918 is 50. This results in a CAN identifier of $200 + 50 \cdot 4 = 400$ for the first CAN data telegram of PZD-send. The further CAN data telegrams for PZD-send have CAN identifiers 401 to 403.											

P714 (CB parameter 4)	P699 (CB parameter 4)
Number of process data to be sent in the case of PZD-send With this parameter, the number of process data to be sent in the case of PZD-send is specified. Valid values are 1 to 16 words. From this information, the actual number and the length of the CAN data telegrams are determined. If the number of process data is outside the valid range (1 to 16), error F080 appears when status 4 "hardware configuration" is left. After acknowledgement of the error, you are in the "hardware configuration" status again and can correct the incorrect parameterization. Example: The basic identifier for PZD-send in P713 / P698 is set to 200. The node address in P918 is 50. This results in a CAN identifier of $200 + 50 \cdot 4 = 400$ for the first CAN data telegram of PZD-send. If the number of process data (P714 / P699) is now 10, a CAN data telegram with four words with CAN identifier 400 and a telegram with CAN identifier 401 is sent as is a CAN data telegram with two words and CAN identifier 402. These are the entered 10 words of process data. CAN identifier 403 is unused and is not sent.	

P715 (CB parameter 5)	P700 (CB parameter 5)
Up-date rate for PZD-send With this parameter, the up-date rate is set in milliseconds for PZD-send, i.e. the time base in which new actual values are to be sent from the unit. Meaning of the parameter values: • 0: Actual values are only sent on request (remote transmission requests). • 1 to 65534: Actual values are sent according to the time set in ms or on request (Remote Transmission Requests). 65535: Actual values are sent if the values have changed (event) or on request (remote transmission requests). This function should only be used if the values to be transferred only rarely change because, otherwise, the bus load becomes very high.	

P716 (CB parameter 6)	P701 (CB parameter 6)										
CAN identifier for PZD-receive broadcast With this parameter, the CAN identifier for PZD-receive broadcast (receiving process data = setpoints / control words) can be set. A broadcast telegram is to be received by all slaves on the bus. This parameter must be set the same for all slaves. Because only four setpoints (= 8 bytes) can be sent with a CAN data telegram but 16 setpoints are supported by MASTERDRIVES units, a total of four CAN data telegrams with four CAN identifiers are needed for transferring the setpoints. The following three CAN identifiers are therefore also provided for PZD-receive broadcast. The following table applies:											
<table border="1"> <thead> <tr> <th>Contents</th><th>CAN identifier</th></tr> </thead> <tbody> <tr> <td>Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2</td><td>P716/P701</td></tr> <tr> <td>Setpoint 5 to Setpoint 8</td><td>P716/P701 + 1</td></tr> <tr> <td>Setpoint 9 to Setpoint 12</td><td>P716/P701 + 2</td></tr> <tr> <td>Setpoint 13 to Setpoint 16</td><td>P716/P701 + 3</td></tr> </tbody> </table>		Contents	CAN identifier	Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2	P716/P701	Setpoint 5 to Setpoint 8	P716/P701 + 1	Setpoint 9 to Setpoint 12	P716/P701 + 2	Setpoint 13 to Setpoint 16	P716/P701 + 3
Contents	CAN identifier										
Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2	P716/P701										
Setpoint 5 to Setpoint 8	P716/P701 + 1										
Setpoint 9 to Setpoint 12	P716/P701 + 2										
Setpoint 13 to Setpoint 16	P716/P701 + 3										
With the value 0 (pre-assigned) in this parameter, PZD-receive broadcast is deactivated. If the calculated CAN identifier for PZD-receive broadcast is outside the valid range (1 to 2000) or if it overlaps another CAN identifier, error F080 appears when status 4 "hardware configuration" is left. After acknowledgement of the error, you are in the "hardware configuration" status again and can correct the incorrect parameterization.											
Example: The CAN identifier for PZD-receive broadcast in P716 / P701 is set to 100. This results in a CAN identifier of 100 for the first CAN data telegram of PZD-receive broadcast. The further CAN data telegrams for PZD-receive broadcast have CAN identifiers 101 to 103.											

P717 (CB parameter 7)	P702 (CB parameter 7)										
CAN identifier for PZD-receive multicast With this parameter, the CAN identifier for PZD-receive multicast (receiving process data = setpoints / control words) can be set. A multicast telegram is to be received by a group of slaves on the bus. This parameter must be set the same for all slaves in this group. Because only four setpoints (= 8 bytes) can be transferred with one CAN data telegram but 16 setpoints are supported by MASTERDRIVES units, a total of four CAN data telegrams with four CAN identifiers are needed for transferring the setpoints. The following three CAN identifiers are therefore also provided for PZD-receive multicast. The following table applies:											
<table border="1"> <thead> <tr> <th>Contents</th><th>CAN identifier</th></tr> </thead> <tbody> <tr> <td>Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2</td><td>P717/P702</td></tr> <tr> <td>Setpoint 5 to Setpoint 8</td><td>P717/P702 + 1</td></tr> <tr> <td>Setpoint 9 to Setpoint 12</td><td>P717/P702 + 2</td></tr> <tr> <td>Setpoint 13 to Setpoint 16</td><td>P717/P702 + 3</td></tr> </tbody> </table>		Contents	CAN identifier	Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2	P717/P702	Setpoint 5 to Setpoint 8	P717/P702 + 1	Setpoint 9 to Setpoint 12	P717/P702 + 2	Setpoint 13 to Setpoint 16	P717/P702 + 3
Contents	CAN identifier										
Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2	P717/P702										
Setpoint 5 to Setpoint 8	P717/P702 + 1										
Setpoint 9 to Setpoint 12	P717/P702 + 2										
Setpoint 13 to Setpoint 16	P717/P702 + 3										
With the value 0 (pre-assigned) in this parameter, PZD-receive multicast is deactivated. If the CAN identifiers for PZD-receive multicast are outside the valid range (1 to 2000) or if they overlap another CAN identifier, error F080 appears when status 4 "hardware configuration" is left. After acknowledgement of the error, you are in the "hardware configuration" status again and can correct the incorrect parameterization.											
Example: The CAN identifier for PZD-receive multicast in P717 / P702 is set to 50. This results in a CAN identifier of 50 for the first CAN data telegram of PZD-receive multicast. The further CAD data telegrams for PZD-receive multicast have CAN identifiers 51 to 53.											

P718 (CB parameter 8)	P703 (CB parameter 8)										
CAN identifier for PZD-receive cross With this parameter, the CAN identifier for PZD-receive cross (receiving process data = setpoints / control words) can be set. By means of cross data traffic between slaves, the actual values sent by a slave (by means of PZD-send) can be used as setpoints by another slave. For this, the parameter value of this parameter is set to the CAN identifier of the CAN data telegram from which the setpoints are to be obtained. Because only four setpoints (= 8 bytes) can be transferred with one CAN data telegram but 16 setpoints are supported by MASTERDRIVES units, a total of four CAN data telegrams with four CAN identifiers are needed for transferring the setpoints. The following three CAN identifiers are therefore also provided for PZD-receive cross. The following table applies: <table> <tr> <th>Contents</th><th>CAN identifier</th></tr> <tr> <td>Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2</td><td>P718/P703</td></tr> <tr> <td>Setpoint 5 to Setpoint 8</td><td>P718/P703 + 1</td></tr> <tr> <td>Setpoint 9 to Setpoint 12</td><td>P718/P703 + 2</td></tr> <tr> <td>Setpoint 13 to Setpoint 16</td><td>P718/P703 + 3</td></tr> </table> With the value 0 (pre-assigned) in this parameter, PZD-receive cross is deactivated. If the CAN identifiers for PZD-receive cross are outside the valid range (1 to 2000) or if they overlap another CAN identifier, error F080 appears when status 4 "hardware configuration" is left. After acknowledgement of the error, you are in the "hardware configuration" status again and can correct the incorrect parameterization. Example: The data telegram with CAN identifier 701 is to be used as setpoint 5 to setpoint 8. For this, the CAN identifier for PZD-receive cross in P718 / P703 must be set to 700. This results in a CAN identifier of 700 for the first CAN data telegram of PZD-receive cross. The further CAN data telegrams have CAN identifiers 701 to 703, i.e. the data telegram 701 results in setpoint 5 to setpoint 8.		Contents	CAN identifier	Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2	P718/P703	Setpoint 5 to Setpoint 8	P718/P703 + 1	Setpoint 9 to Setpoint 12	P718/P703 + 2	Setpoint 13 to Setpoint 16	P718/P703 + 3
Contents	CAN identifier										
Control word 1 / Setpoint 2 / Setpoint 3 / Setpoint 4 or Control word 2	P718/P703										
Setpoint 5 to Setpoint 8	P718/P703 + 1										
Setpoint 9 to Setpoint 12	P718/P703 + 2										
Setpoint 13 to Setpoint 16	P718/P703 + 3										

P719 (CB parameter 9)	P704 (CB parameter 9)
CAN identifier for PKW-task broadcast With this parameter, the CAN identifier for PKW-task broadcast (parameter task) can be set. A broadcast telegram is to be received by all slaves on the bus. This parameter must therefore be set the same for all slaves. With the help of this function, a parameter task can be simultaneously issued to all slaves on the bus. The parameter reply is given with the CAN identifier of the PKW-reply (see P711 / P696), namely $(\text{Parameter value of P711 / P696}) + (\text{Parameter value of P918}) \cdot 2 + 1$ With the value 0 (pre-assigned) in this parameter, PKW-task broadcast is deactivated. If the calculated CAN identifier for PKW-task broadcast is outside the valid range (1 to 2000) or if it overlaps another CAN identifier, error F080 appears when status 4 "hardware configuration" is left. After acknowledgements of the error, you are in the "hardware configuration" status again and can correct the incorrect parameterization. Example: The basic identifier for parameterization in P711 / P696 is set to 1500. The node address in P918 is 50. This results in a CAN identifier of $1500 + 50 \cdot 2 = 1600$ for PKW-task and 1601 for PKW-reply. The CAN identifier for PKW-reply broadcast in P719 / P704 is set to 1900. A parameter task can be issued by means of PKW-task broadcast, namely with CAN identifier 1900, whereas the reply is given with CAN identifier 1601 by means of PKW-reply.	

P720 (CB parameter 10)	P705 (CB parameter 10)								
Baud rate of the slave on the CAN bus									
With this parameter, the baud rate of the slave on the CAN bus is set. The following applies:									
If the baud rate is outside the valid range, error F080 appears when status 4 "hardware configuration" is left. After acknowledgement of the error, you are in the "hardware configuration" status again and can correct the incorrect parameterization.									
Parameter value	0	1	2	3	4	5	6	7	8
Baud rate [kBit/s]	10	20	50	100	125	250	500	800	1000

P721 (CB parameter 11)

P706 (CB parameter 11)

Special CAN bus settings

This parameter is only present in MASTERDRIVES units from the following software versions onwards:

MASTERDRIVES	Software version
SIMOVERT MASTERDRIVES MC	≥ 1.0
SIMOVERT MASTERDRIVES FC	≥ 1.3
SIMOVERT MASTERDRIVES VC	≥ 1.3
SIMOVERT MASTERDRIVES SC	≥ 1.2
SIMOVERT MASTERDRIVES E/R	≥ 3.1
SIMOVERT MASTERDRIVES AFE	≥ 1.0

- Index i001:** With this parameter, different CAN profiles can be set in future. At the present time, only the value 0 (pre-assigned) is valid.
- Index i002:** With this parameter, the bus timing on the CAN bus can be influenced. With the value 0 (pre-assigned), the internal setting resulting from the baud rate is made. All other values are directly set without a plausibility check.

This parameter should generally be allowed to keep its pre-assigned setting of 0!

Meaning of the parameter-value bits:

Bit0 - Bit5: BRP (Baud rate prescaler).

Bit6 - Bit7: SJW SJW (Synchronization Jump Width). Maximum shortening or lengthening of a bit time by means of resynchronization.

Bit8 - Bit11: TSEG1 (Time Segment 1). Time intervals before the scanning time. Valid values are 2 to 15.

Bit12 - Bit14: TSEG2 (Time Segment 2). Time interval after the scanning time. Valid values are 1 to 7. In addition TSEG2 must be greater than SJW.

Bit 15: Not assigned

Internal standard pre-assignments of the bus timing, depending on the baud rate:

Baud rate	BRP	SJW	TSEG1	TSEG2	Hex value
10 kBit (P720/P705 = 0)	39	2	15	2	2FA7
20 kBit (P720/P705 = 1)	19	2	15	2	2F93
50 kBit (P720/P705 = 2)	7	2	15	2	2F87
100 kBit (P720/P705 = 3)	3	2	15	2	2F83
125 kBit (P720/P705 = 4)	3	1	12	1	1C43
250 kBit (P720/P705 = 5)	1	1	12	1	1C41
500 kBit (P720/P705 = 6)	0	1	12	1	1C40
800 kBit (P720/P705 = 7)	0	1	6	1	1640
1 MBit (P720/P706 = 8)	0	1	4	1	1440

P721 (CB parameter 11)	P706 (CB parameter 11)
Formula for calculating the baud rate from the constants: time quantum = $tq = (BRP+1) * 2 * tClk$ Clock Period = $tClk = 62.5 \text{ ns (at 16 MHz)}$ Synchronization segment = $tSync-Seg = tq$ Time Segment 1 (before scanning time) = $tTSeg1 = (TSEG1+1)*tq$ Time Segment 2 (after scanning time) = $tTSeg2 = (TSEG2+1)*tq$ Bit time = $tSync-Seg + tTSeg1 + tTSeg2$ Baud rate = $1 / \text{bit time}$ The parameter value corresponds to the value of the bit timing register of the CAN component. A more exact description of this bit timing register can be found in the manual of the CAN module of the C167CR or in the manual of the component, INTEL 82527 (extended CAN).	

P918.1 (CBC bus address)	P918 (CBC bus address)
Here, the node address of the unit on the CAN bus is set. It is included in the calculation of the CAN identifier for parameter tasks and replies (PKW task / PKW reply) and process data (PZD-receive / PZD-send). (See also P711 / P696, P712 / P697 and P713 / P698).	

NOTE

When the above settings have been made, the CBC is regarded as registered in the converter and is ready for communication via the CAN bus.

Changing parameters or specifying process data via the CAN bus is not yet possible after this step.

Parameterization must first be enabled and the process data still have to be softwired in the converter.

8.4.5.2 Process-data softwiring in the units

Definition

Process data interconnection involves the linking up of setpoints and control bits to the RAM interface. The transferred process data only become effective when the used bits of the control words as well as the setpoints, status words and actual values are allocated (connected) to the dual-port RAM interface.

The received process data are stored by the CBC at fixed, pre-defined addresses in the dual-port RAM. A connector (e.g. 3001 for PZD1) is assigned to each item of process data (PZDi, $i = 1$ to 10). The connector also determines whether the corresponding PDZi ($i = 1$ to 10) is a 16-bit value or a 32-bit value.

With the help of selector switches (e.g. P554.1 = selector switch for bit 0 of control word 1), the setpoints or the individual bits of the control words can be assigned to a particular PZDi in the dual-port RAM. In order to do this, the connector belonging to the required PZDi is assigned to the selector switch.

NOTE

In function classes CUPM, CUMC, CUVC and Compact PLUS, the control words STW1 and STW2 are also available in bit form on so-called binectors (explanations of BICO systems can be found in Chapter 4 "Function Blocks and Parameters").

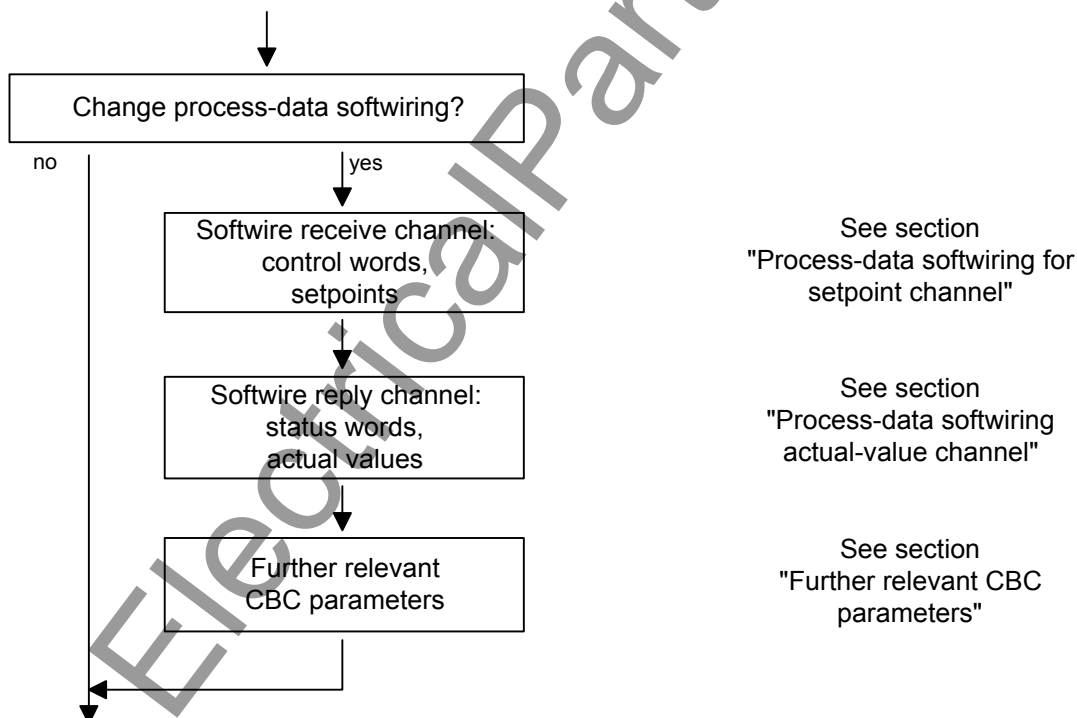


Fig. 8.4-18 Procedure for altering process data

Example

On the following pages, you will find examples of how the transferred data are routed in the units by means of process-data softwiring (logical connection).

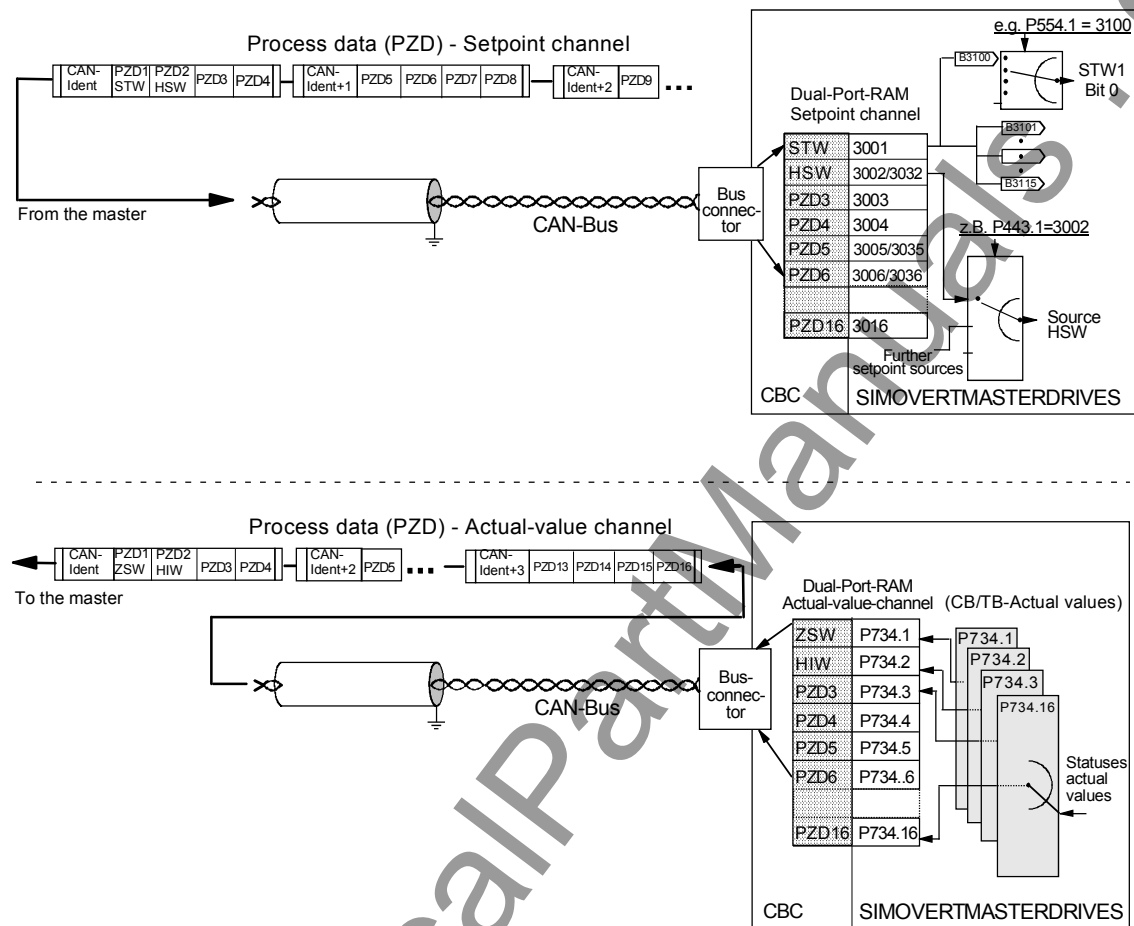


Fig. 8.4-19 Example of process-data connection for function classes CUPM, CUMC and CUVC

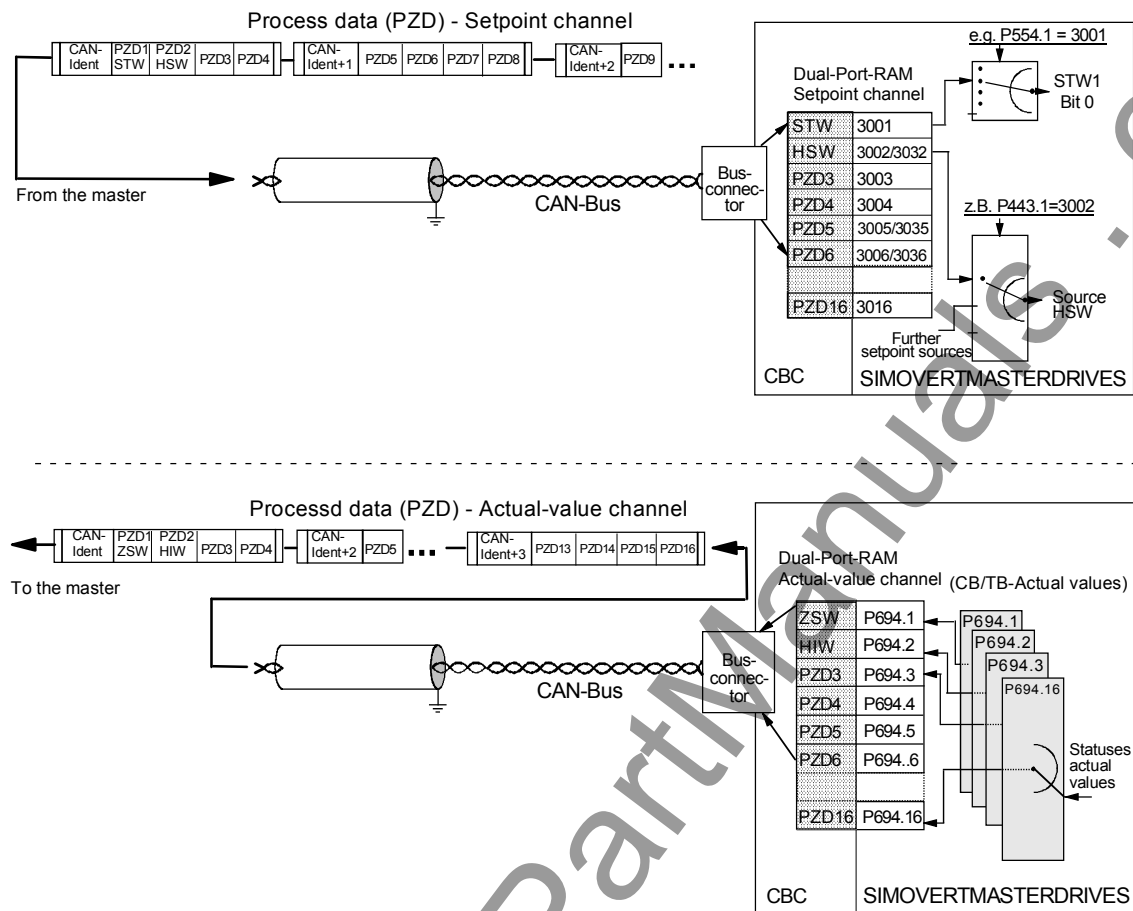


Fig. 8.4-20 Example of process-data interconnection for function classes CU1, CU2 or CU3

**Process-data
connection -
Setpoint channel**

- ◆ The "tens digit" of the binector enables a distinction to be made between a 16-bit item of process data (e.g. 3002) and a 32-bit item of process data (e.g. 3032).
- ◆ If an item of process data is transferred as a 16-bit quantity, you must assign the connector which belongs to the desired PZDi and which is for a 16-bit item of process data (e.g. if PZD2 is assigned a 16-bit item of process data, the relevant connector is 3002) to the selection switch (see section "Control Word and Status Word" in the instruction manual of the converter).
- ◆ If an item of process data is transferred as a 32-bit quantity, you must assign the connector which belongs to the desired PZDi and which is for a 32-bit item of process data (e.g. if PZD2+PZD3 are assigned a 32-bit item of process data, the relevant connector is 3032) to the selection switch (see section "Control Word and Status Word" in the instruction manual of the converter).
- ◆ The first word (relevant connector: 3001) of the received process data is always allocated to control word 1 (STW1). The meaning of the control-word bits is given in the operating instructions for the converter in the section, "Start-up aids".
- ◆ The second word is always allocated to the main setpoint (HSW). If the main setpoint is transferred as a 32-bit item of process data, it also occupies word 3. In this case, the most significant component is transferred in word 2 and the least significant component in word 3.
- ◆ If a control word 2 (STW2) is transferred, the fourth word (relevant connector = 3004) is always allocated to STW2. The meaning of the control-word bits is given in the instruction manual for the converter in the section, "Start-up aids".
- ◆ The connector is always a four-digit number. The connectors assigned to the process data (PZD1 to PZD16) are given in the function plan.
- ◆ The connector is entered at the PMU as a 4-digit number (e.g. 3001). During parameterization via the CAN bus, the connector is entered via the bus in the same way as via the PMU (e.g. connector 3001 is transferred as 3001(hex)).

NOTE

Process-data connection (softwiring) of the setpoint channel can also be carried out via the CAN bus as long as P053 has previously been set to an odd number.

Example for the setpoint channel

PZD connection for the bits of control word 1 (STW1) and of the main setpoint (HSW) and the bits of control word 2 (STW2).

At the converter via the PMU		Meaning
P554.1 = <u>3100</u>	P554.1 = <u>3001</u>	Control word 1 bit 0 (Source ON/OFF1) via DPR interface (word 1)
P555.1 = <u>3101</u>	P555.1 = <u>3001</u>	Control word 1 bit 1 (Source ON/OFF2) via DPR interface (word 1)
P443.1 = <u>3002</u>	P443.1 = <u>3002</u>	16-bit main setpoint (Source main setpoint) via DPR interface (word 2)
P588.1 = <u>3412</u>	P588.1 = <u>3004</u>	Control word 2 bit 28 (Src no ext. alarm1) via DPR interface (word 4)

Based on the factory setting of the converter, the above example of parameterization represents a functioning method of connecting (softwiring) the process data (setpoints).

Italics:

Parameter number (for the PMU as a decimal number; via the CAN bus as an equivalent HEX number).

Single underline:

Index (for the PMU as a decimal number, via the CAN bus as an equivalent HEX number).

Double underline:

Connector: defines whether the parameter selected by means of the *parameter number* is transferred as a 16-bit value or as a 32-bit value and at which position in the PZD setpoint telegram (PZDi) the parameter is transferred.

- White background = MASTERDRIVES, CUPM, CUMC or CUVc
- Grey background = MASTERDRIVES FC (CU1), VC (CU 2) or SC (CU 3)

Process-data connection - Actual-value channel

The actual-value process data (PZDi, i = 1 to 16) are assigned to the corresponding status words and actual values by means of the indexed parameter P734.i / P694.i (CB/TB actual values). Each index stands for an item of process data (e.g. B. 5 → PZD5 and so on). Please enter the number of the parameter - whose value you wish to transfer with the corresponding process data - in parameter P734.i / P694.i (see also "Parameter list") under the relevant index.

The status word should be entered in the PZD1 word of the PZD reply (actual-value channel) and the main actual value in the PZD2 word. Further assignment of the PZDs (PZD1 to PZD16, if necessary) is not defined. If the main actual value is sent as a 32-bit value, it is assigned to PZD2 and PZD3.

The meaning of the status-word bits can be found in the operating instructions of the converter in the section "Start-up aids".

Example for the actual-value channel

PZD connection for status word 1 (ZSW1), the main actual value (HIW) and status word 2 (ZSW2)

At the converter via the PMU		Meaning
P734. <u>1</u> = <u>32</u>	P694. <u>1</u> = <u>968</u>	Status word 1 (K032 / P968) is transferred in the actual-value channel by means of PZD1.
P734. <u>2</u> = <u>151</u>	P694. <u>2</u> = <u>218</u>	The actual speed n/f (KK151 / P218) is transferred in the actual-value channel by means of PZD2 (here as a 16-bit quantity; PZD3 not occupied here).
P734. <u>4</u> = <u>33</u>	P694. <u>4</u> = <u>553</u>	Status word 2 (K033 / P553) is transferred in the actual-value channel by means of PZD4.
Example: 32-bit main actual value		
P734. <u>2</u> = <u>151</u>	P694. <u>2</u> = <u>218</u>	The actual speed n/f (KK151 / P218) is transferred in the actual-value channel by means of PZD2 ...
P734. <u>3</u> = <u>151</u>	P694. <u>3</u> = <u>218</u>	... and by means of PZD3 as a 32-bit value.

Italics:

P734 / P694 (CB/TB actual values), for the PMU, shown as a decimal number; via the CAN bus, transferred as an equivalent HEX number (2B6 Hex).

Single underline:

Index (for the PMU, as a decimal number; via the CAN bus, as an equivalent HEX number). Specifies at which position in the PZD actual-value telegram (PZDi) the actual value selected by means of the parameter number is to be transferred.

Double underline:

Parameter number of the desired actual value.

- White background = MASTERDRIVES, CUPM, CUMC or CUVC
- Grey background = MASTERDRIVES FC (CU1), VC (CU 2) or SC (CU 3)

NOTE

If actual values are sent as a 32-bit datum, you must enter the associated connector number at two consecutive words (indices).

Other relevant CBC parameters

P722 (CB/TB TIgOFF)	P695 (CB/TB TIgOFF)
Telegram failure time With parameter P722 / P695 (see also operating instructions of the converter, section "Parameter list"), you can specify whether the entry of process data into the dual-port RAM by the CBC is to be monitored by the converter. The parameter value of this parameter corresponds to the telegram failure time in ms. The pre-assigned value of this parameter is 10 ms, i.e. there must be a maximum of 10 ms between two received process-data CAN telegrams, otherwise the converter switches off with F082. With the parameter value 0, the monitoring function is de-activated. The converter monitors the entry of process data into the dual-port RAM from that point of time at which the CBC enters process data into the dual-port RAM for the first time. Only from this point of time onwards can error F082 be triggered!	

DANGER

If the "On" command (bit 0) has been softwired to the dual-port RAM, the following measures must be taken for reasons of safety:

An "OFF2" or "OFF3" command (see instruction manual of the converter, section "Control word") must be additionally parameterized to the terminal strip / PMU as, otherwise, the converter can no longer be turned off by means of a defined command if the communications system breaks down!

P692 (Reaction TIgOFF)
Reaction to telegram failure With parameter P692 (see also instruction manual of the converter, section "Parameter list"), you can specify how the converter is to react to telegram failure. With the parameter value 0 "Fault", the converter immediately switches off with fault F082. The drive coasts to a stop. With parameter value 1 "OFF3 (fast stop)", the drive carries out an OFF3 command (OFF with fast stop) and only then assumes a fault status with fault F082.

P781.13 (fault delay; only applies to CUPM, CUMC and CUVC)
With this parameter, P731.13, fault F082 can be delayed, i.e. the drive is not turned off immediately when a fault occurs but only after expiry of the time entered in the parameter. This makes it possible to react flexibly to a bus failure. With the help of binector B0035 "CB/TB telegram failure", the drive can be shut down (OFF1 or OFF3) by making the fault delay longer than the ramp-down time.

8.4.6 Diagnosis and troubleshooting

NOTE

With regard to basic parameterization, please note the differences to the types of unit with the older function classes FC (CU1), VC (CU2) and SC (CU3). These differences are described below.

In order to make these differences clear, the parameter numbers and other deviations are either printed in dark gray or have a dark gray background.

8.4.6.1 Evaluation of hardware diagnostics

LED displays

On the front of the optional CBC board, there are three LED displays which give information on the current operating status. The following LEDs are provided:

- ◆ CBC on (red)
- ◆ Data exchange with the basic unit (yellow)
- ◆ Telegram traffic via CAN (green)

Status display

LED	Status	Diagnostic information
Red	Flashing	CBC in operation; voltage supply on
Yellow	Flashing	Fault-free data exchange with the basic unit
Green	Flashing	Fault-free process-data transfer via the CAN bus

Table 8.4-9 Status display of the CBC

Fault displays

LED	Status	Diagnostic information
Red	Flashing	Cause of fault:
Yellow	Continuously lit	Serious fault in the CBC
Green	Continuously lit	Remedy: replace CBC

Table 8.4-10 Fault display for CBC faults

LED	Status	Diagnostic information
Red	Flashing	CBC is waiting for the start
Yellow	Off	of parameterization by the
Green	Continuously lit	converter / inverter

Table 8.4-11 Fault display during parameterization

LED	Status	Diagnostic information
Red	Flashing	CBC is waiting for completion of parameterization by the converter / inverter
Yellow	Continuously lit	
Green	Off	

Table 8.4-12 Fault display during parameterization

LED	Status	Diagnostic information
Red	Flashing	No net-data traffic via the CAN bus, e.g. bus connector pulled out, EMC fault, interchanged connection, nodes are not being supplied with net data via the CAN bus
Yellow	Flashing	
Green	Off	

Table 8.4-13 Fault display during operation

NOTE

During normal operation, all three LEDs light up synchronously and for the same length of time (flashing)!

The stationary status of an LED (on or off) indicates an unusual operating status (parameterization phase or fault)!

8.4.6.2 Fault displays and alarms on the basic unit

If errors/faults occur in CAN-bus communication with the CBC, corresponding errors or alarms are also displayed on the PMU or OP1S of the basic unit.

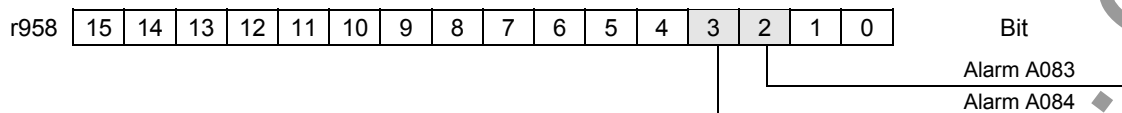
Alarms

Alarm	Meaning
A 083	CAN telegrams with errors are being received or sent and the internal error counter has exceeded the alarm limit. The CAN telegrams with errors are ignored. The data last sent remain valid. If these CAN telegrams contain process data, the telegram-failure monitor (P722 / P695) can respond – depending on the setting – with error F082 (DPR telegram failure). If the PKW CAN telegrams contain errors or are defective, there is no reaction in the converter. → Check parameter P720 / P705 (baud rate) for each bus node and, if necessary, correct. → Check cable connection between the bus nodes → Check cable shield. The bus cable must be shielded on both sides. → Lower the EMC loading → Replace CBC board
A 084	CAN telegrams with errors are being received or sent and the internal error counter has exceeded the fault limit. The CAN telegrams with errors are ignored. The data last sent remain valid. If these CAN telegrams contain process data, the telegram monitor (P722 / P695) – depending on the setting – can respond with error F082 (DPR telegram failure). If the PKW CAN telegrams contain errors or are defective, there is no reaction in the converter. → Check parameter P720 / P705 (baud rate) for each bus node and, if necessary, correct. → Check CAN-bus master → Check cable connection between the bus nodes → Check cable shield. The bus cable must be shielded on both sides. → Lower the EMC loading → Replace CBC board

- Possible cause
- Remedy

Table 8.4-14 Alarm displays on the basic unit

Alarms A083 and A084 are also stored as information in alarm parameter 6 (r958). The individual alarms are assigned to the corresponding bits in r958 (Bit x = 1: alarm present):



Fault/error display

When the CBC is combined with the control/technology board (CU/TB), the following fault messages can occur:

Fault	Meaning
F 080	TB/CB Init.: Incorrect initialization and parameterization of the CBC via the dual-port RAM interface (DPR interface) - CBC selected with parameter P090/P091, but not inserted (not in the case of CUPM, CUMC or CUVC) → Correct parameter P090 P091, insert CBC - Parameterization for CBC false, cause of incorrect parameterization in diagnostic parameter r731.01 → Correct CB parameter P711-P721 / P696 - P706. Correct CB bus address P918 - CBC defective → Replace CBC
F 081	DPR heartbeat: The CBC is no longer processing the heartbeat counter. - CBC not correctly inserted into the electronics box → Check CBC - CBC defective → Replace CBC
F 082	DPR telegram failure: The telegram-failure time set by means of parameter P722 / P695 has expired - CAN-bus master has failed (green LED on the CBC is continuously off) - Cable connection between the bus nodes has been interrupted (green LED on the CBC is continuously off) → Check the bus cable - EMC loading of the bus cable too high. → Refer to EMC notes - Telegram monitoring time has been set too low (the green LED on the CBC flashes) → Increase the parameter value in P722 / P695 - CBC defective → Replace CBC

- Possible cause
- Remedy

Table 8.4-15 Fault displays on the basic unit

8.4.6.3 Evaluation of the CBC diagnostic parameter

NOTE

Please note that, for types of unit with the older function classes FC (CU1), VC (CU2) and SC (CU3), indexed parameter r731.i is to be used appropriately instead of r732.i

The CBC stores this information in a diagnostics buffer to support start-up and for service purposes. The diagnostic information can be read out with indexed parameter r732.i (CB/TB diagnosis). This parameter is displayed as a hexadecimal. The CBC diagnostics buffer is assigned as follows:

CBC-diagnosis parameter

Meaning	r731.i	r732.i
Fault detection configuration	r731.1	r732.1
Counter: telegrams received without faults/errors	r731.2	r732.2
Counter: lost PZD telegrams	r731.3	r732.3
Counter for Bus-Off states	r731.4	r732.4
Counter for error-warning states	r731.5	r732.5
Assigned internally	r731.6	r732.6
Assigned internally	r731.7	r732.7
Assigned internally	r731.8	r732.8
Assigned internally	r731.9	r732.9
Counter for PZD telegrams sent without errors/faults	r731.10	r732.10
Counter for faults during transfer of PZD telegrams	r731.11	r732.11
Assigned internally	r731.12	r732.12
Assigned internally	r731.13	r732.13
Counter for PKW tasks processed without errors/faults	r731.14	r732.14
Counter for faults/errors during processing of PKW tasks	r731.15	r732.15
Type of fault/error in the case of faults during processing of PKW tasks	r731.16	r732.16
Assigned internally	r731.17	r732.17
Counter for lost PKW tasks	r731.18	r732.18
Reserved	r731.19	r732.19
Reserved	r731.20	r732.20
Reserved	r731.21	r732.21
Reserved	r731.22	r732.22
Reserved	r731.23	r732.23
Assigned internally	r731.24	r732.24
Assigned internally	r731.25	r732.25
Software version	r731.26	r732.26
Software identification	r731.27	r732.27
Software date, day/month	r731.28	r732.28
Software date, year	r731.29	r732.29

Table 8.4-16 CBC diagnostics buffer

8.4.6.4 Meaning of CBC diagnosis

P732.1

Fault detection configuration

If an invalid value or an invalid combination of parameter values is contained in the CB parameters, the converter switches to fault mode with fault F080 and fault value 5 (r949). The cause of the incorrect parameterization can then be determined by means of this index of CB diagnostic parameter r731.

Value (hex)	Meaning
00	No fault/error
01	Incorrect bus address (P918)
02	Incorrect CAN ID in the case of a PKW task (P711 / P696)
03	<i>Internal</i>
04	<i>Internal</i>
05	Incorrect CAN ID in the case of a PKW-task broadcast (P719 / P704)
06	<i>Internal</i>
07	Incorrect CAN ID in the case of a PZD-receive (P712 / P697)
08 - 0C	<i>Internal</i>
0D	Incorrect CAN ID in the case of a PZD-send (P713 / P698)
0E	PZD-send length is 0 (P714 / P699)
0F	PZD-send length too great (>16) (P714 / P699)
10 - 13	<i>Internal</i>
14	Incorrect CAN ID in the case of a PZD-receive broadcast (P716 / P701)
15	Incorrect CAN ID in the case of a PZD-receive multicast (P717 / P702)
16	Incorrect CAN ID in the case of a PZD-receive cross (P718 / P703)
17	Invalid baud rate (P720 / P705)
18 - 22	<i>Internal</i>
23	Incorrect CAN protocol type (P721 / P706.01)
24	PKW-request broadcast (P719 / P704) without PKW task (P711 / P696)
25 - 2F	<i>Reserved</i>
30	Overlapping of CAN identifier PKW <-> PKW-broadcast
31	Overlapping of CAN identifier PKW <-> PZD-receive
32	Overlapping of CAN identifier PKW <-> PZD-send
33	Overlapping of CAN identifier PKW <-> PZD-receive broadcast
34	Overlapping of CAN identifier PKW <-> PZD-receive multicast
35	Overlapping of CAN identifier PKW <-> PZD-receive cross
36	Overlapping of CAN identifier PKW-broadcast <-> PZD-receive
37	Overlapping of CAN identifier PKW-broadcast <-> PZD-send
38	Overlapping of CAN identifier PKW-broadcast <-> PZD-receive broadcast

Value (hex)	Meaning
39	Overlapping of CAN identifier PKW-broadcast <-> PZD-receive-Multicast
3A	Overlapping of CAN identifier PKW-broadcast <-> PZD-receive cross
3B	Overlapping of CAN identifier PZD-receive <-> PZD-send
3C	Overlapping of CAN identifier PZD-receive <-> PZD-receive-Broadcast
3D	Overlapping of CAN identifier PZD-receive <-> PZD-receive multicast
3E	Overlapping of CAN identifier PZD-receive <-> PZD-receive cross
3F	Overlapping of CAN identifier PZD-send <-> PZD-receive broadcast
40	Overlapping of CAN identifier PZD-send <-> PZD-receive multicast
41	Overlapping of CAN identifier PZD-send <-> PZD-receive cross
42	Overlapping of CAN identifier PZD-receive broadcast <-> PZD-receive multicast
43	Overlapping of CAN identifier PZD-receive broadcast <-> PZD-receive cross
44	Overlapping of CAN identifier PZD-receive multicast <-> PZD-receive cross

r731.02**Counter PZD-receive CAN telegrams**

Counter for PZD CAN telegrams received error-free since voltage ON.

r731.03**Counter Lost PZD CAN telegrams**

Counter for lost PZD telegrams since voltage ON. If the CAN-bus master sends process-data telegrams faster than the slave can process them, telegrams are lost. These lost telegrams are totaled here.

r731.04**Counter Bus-Off**

Counter of the bus-off states since voltage ON (alarm A084).

r731.05**Counter Error-Warning**

Counter of the error-warning states since voltage ON (alarm A083).

r731.10**Counter PZD-send CAN telegrams**

Counter for PZD telegrams sent error-free since voltage ON.

r731.11**Counter Errors PZD-send CAN telegrams**

Counter for errors during sending of PZD telegrams, i.e. when a PZD telegram was to be sent but it was not possible, e.g. in the case of bus overload.

r731.14 Counter PKW CAN telegrams
Counter for PKW tasks and replies processed error-free since voltage ON.

r731.15 Counter Errors PKW CAN telegrams
Counter for errors during processing of PKW tasks, e.g. due to bus overload or missing reply from the basic unit.

r731.16 Error type PKW CAN telegrams
Here, an error identifier is entered if an error occurs during processing of a PKW task.

Value	Meaning
0	No error
1	<i>Internal</i>
2	<i>Internal</i>
3	<i>Internal</i>
4	<i>Internal</i>
5	<i>Internal</i>
6	<i>Internal</i>
7	<i>Internal</i>
8	<i>Internal</i>
9	Error during sending of PKW reply (in the case of waiting for a free channel)
10	<i>Internal</i>
11	Time out in the case of waiting for a PKW reply from the basic unit (basic unit does not process any PKW tasks)
12	Time out in the case of waiting for a free channel (bus overload)

r731.18 Counter Lost PKW CAN telegrams
Counter for PKW tasks lost since voltage ON. If the CAN-bus master sends PKW tasks faster than the slave can process them, PKW tasks are lost. These lost PKW tasks are totaled here.

r731.26 Software version

r731.27 Software identifier

r731.28 Software date
Software date, day (high byte) and month (low byte) shown in hexadecimal form

r731.29 Software date
Software date, year (shown in hexadecimal form)

8.4.7 Appendix

Technical data

Order No.	6SE7090-0XX84-0FG0
Size (length x width)	90 mm x 83 mm
Degree of pollution	Pollution degree 2 to IEC 664-1 (DIN VDE 0110/T1), Moisture condensation during operation is not permissible
Mechanical specifications	To DIN IEC 68-2-6 (if board correctly mounted)
During stationary use	
• deflection	0.15 mm in the frequency range 10 Hz to 58 Hz
• acceleration	19.6 m/s ² in the frequency range > 58 Hz to 500 Hz
During transport	
• deflection	3.5 mm in the frequency range 5 Hz to 9 Hz
• acceleration	9.8 m/s ² in the frequency range > 9 Hz to 500 Hz
Climatic class	Class 3K3 to DIN IEC 721-3-3 (during operation)
Type of cooling	Natural-air cooling
Permissible ambient or cooling-medium temperature	
• during operation	0° C to +70° C (32° F to 158° F)
• during storage	-25° C to +70° C (-13° F to 158° F)
• during transport	-25° C to +70° C (-13° F to 158° F)
Humidity rating	Relative humidity ≤ 95 % during transport and storage ≤ 85 % during operation (moisture condensation not permissible)
Supply voltage	5 V ± 5 %, max. 500 mA, internally from the basic unit

8.5 CBC CANopen communication board

This chapter describes the CANopen software functions.

The CANopen software functions comply with profile definitions:

CiA DS 301 4.01

CiA DSP 402 V1.1

This functionality is available for the MASTERDRIVES MC and a freely definable CANopen device from CBC SW version 3.0 and later.

CAUTION

Before installing and commissioning a MASTERDRIVES with a communication board, you must read Section 8.4 "CBC communication board" (part CBC) as well as the safety instructions given in Subsections 8.4.1 and 8.4.3.

The terms and abbreviations used in this document are defined in Subsection 8.5.13.

Preconditions

Certain conditions must be fulfilled before the CBC can be operated with CANopen.

- ◆ CANopen functionality is available only with MASTERDRIVES MC **firmware version 1.5 and later**.
- ◆ The actual value weighting factor (AVWF) must be normalized in μm to ensure that actual values and setpoints are converted correctly.
- ◆ The free CANopen device can be a MASTERDRIVES VC, a rectifier/regenerative feedback unit or an AFE.

The following CANopen modi have been implemented for the MASTERDRIVES MC with F01:

- ◆ Profile Velocity mode (speed control)
- ◆ Profile Position mode (MDI positioning)
- ◆ Homing mode (homing)
- ◆ Synchronous mode (electronic gearbox)
- ◆ Setup mode
- ◆ Automatic Position mode
- ◆ Automatic Single Block mode

The following modes have been implemented for the MASTERDRIVES MC:

- ◆ Profile Velocity mode (speed control)
- ◆ Profile Position mode (B-pos positioning)
- ◆ Homing mode
- ◆ Profile Torque mode
- ◆ Setup mode

Parameters must be assigned in the MASTERDRIVES system before the individual CANopen modes can be used.

The relevant scriptfiles are stored on the Drive Monitor CD supplied with every unit. The scriptfiles contain the basic parameter settings for CANopen communication, but no motor settings or optimization data. The scriptfiles have to be adapted to suit individual applications.

You therefore need to adapt, for example, the PDO mapping or the coarse pulse selection.

The necessary interconnections and signal outputs from the CBC are stored as drawings in Subsection 8.5.12.

The CBC CANopen functionality can be utilized only by one CBC in one device. If you require a second CBC to be able to transmit particular process data from the MASTERDRIVES, you will need to operate the second board on layer 2 (P721.01 = 0).

What is the difference between MASTERDRIVES MC, MASTERDRIVES MC-F01 and the free CANopen device?

MASTERDRIVES MC	The modes Profile Position, Setup and Homing with basic positioning are available to the MASTERDRIVES MC without the F01 technology option. The Profile Velocity and Profile Torque modes are processed via the basic unit.
MASTERDRIVES MC-F01	All modes are processed via the F01 technology option on MASTERDRIVES MC-F01.

NOTE

The MASTERDRIVES MC-F01 functionality is available only if you have enabled the F01 option.

The CANopen modes Profile Position, Profile Velocity and Homing are available on the MASTERDRIVES MC-F01. They are controlled via object 6040h.

Manufacturer-specific modes Synchronous and Setup as well as Automatic Position and Automatic Single Block modes can also be used.

The Synchronous mode and Setup modes are controlled via word 6040h. The manufacturer-specific modes (Automatic Position and Automatic Position Single Block mode) are controlled via object 4040h (technology control word).

The above modes are controlled as described in Chapter 9 "Technology Option F01" in the compendium. Object 4041h is used as the status word for the mode. The objects are assigned in the same way as described in function diagrams [FP 809] and [FP 811].

The difference in mode functionalities between the MC and MC-F01 versions is negligible. Some objects differ in terms of their transferability or subindices. Detailed information about these differences can be found in the list of Objects (Subsection 8.5.1) and the receive and transmit PDO tables (Subsection 8.5.2.4).

In Homing mode with the MC but without technology option F01, only homing methods 17-35 are available and referencing (homing) without the technology option is less accurate.

Free CANopen device

The free CANopen device allows units such as AFEs or R/RFs that are not specified in the profile to be linked to CANopen systems. The MASTERDRIVES VC which is not operating as a CANopen device on the CBC can therefore be linked to a CANopen bus system as well.

The objects 'control word 6040h', 'status word 6041h', 'modes of operation 6060h' and 'modes of operation display 6061h' are provided in the free CANopen device. The objects 6040h and 6041h must be linked to the control word and status words of the connected device. Objects 6060h and 6061h transfer the bits only 1:1. You must use free blocks to interconnect the bits in a meaningful manner.

A range of manufacturer-specific objects are provided for the transmission of setpoints or actual values. These must be mapped to a PDO so that they are available in the device. The tables in Subsections 8.5.1 and 8.5.2.4 specify which PDOs and objects are provided for the free CANopen device.

Description

CANopen is a standardized application for distributed industrial automation systems based on CAN and the communication standard CAL. CANopen is a CAN in Automation (CiA) standard which has been used very widely ever since its launch.

CANopen can be regarded as the definitive standard for the implementation of industrial CAN-based system solutions in Europe.

CANopen is founded on a so-called "communications profile" that specifies the basic communication mechanisms and their definition [CiA DS 301].

The primary device types employed in industrial automation systems such as, for example,

- ◆ digital and analog input / output modules [CiA DS 401]
- ◆ drives and motion control [CiA DSP 402]
- ◆ control units [CiA DSP 403]
- ◆ controllers [CiA DSP 404]
- ◆ PLCs [CiA DSP 405]
- ◆ encoders [CiA DSP 406]

are defined in so-called "device profiles".

The device profiles specify the functionality of standard equipment of the relevant type.

A central element of the CANopen standard is the description of device functionality using an "object directory" (OD).

The object directory is divided into two areas, one containing general data about the device such as identification, manufacturer name, etc., plus communication parameters and a second area containing a description of the device functionality.

An entry ("object") in the object directory is identified by a 16-bit index and an 8-bit subindex.

The entries in the object directory make the "application objects" of a device, such as input and output signals, device parameters, device functions or network variables, accessible in standardized form via the network.

In a similar manner to other field bus systems, CANopen also uses two basic data transmission mechanisms, i.e. high-speed exchange of short process data via so-called "Process Data Objects" (PDOs) and access to entries in the object directory via so-called "Service Data Objects" (SDOs). The primary purpose of the latter is to transfer parameters while equipment is being configured and, in general, to transmit long data areas. Process data objects are generally transferred in event-oriented form, cyclically or - on request - as broadcast objects without additional protocol overhead.

A total of 8 bytes of data can be transmitted in one PDO. The transmission and receipt of PDOs can be synchronized throughout the network ("synchronous PDOs") using a synchronization message. The assignment between application objects and a particular PDO (transmission object) can be configured via a structure definition ("PDO mapping") in the OD. Assignments can thus be adapted to meet the requirements of a particular application.

SDOs are transmitted as a confirmed data transfer, with two CAN objects per transmission, in the form of a peer-to-peer connection between two network nodes. The relevant object directory entry is addressed through specification of the index and subindex of the OD entry. Messages with a total length of 5 bytes can be transferred. Transferring SDO messages involves an additional overhead.

Standardized, higher-priority, event-oriented alarm messages ("Emergency_Messages") are provided to signal device faults.

The functionality required for the preparation and coordinated starting of a distributed automation system complies with the mechanisms defined by the CAL Network Management (NMT) specification. The principle of "Node Guarding" underlying the cyclical node monitoring functions are also compliant with NMT.

CAN message identifiers can be assigned to PDOs and SDOs through the direct entry of identifiers in the data structures of the object directory or, for simple system structures, through the use of predefined identifiers.

8.5.1 Object directory

The following tables show a complete list of all implemented objects. The table contains the index and subindex of the object, as well as a brief description of its functionality.

The transmission mode for each object is specified, i.e. the table indicates whether the PDO or SDO transfer method is used for the relevant object. The parameters or connectors which contain the object data are also listed. If the table specifies both modes of transmission, the object can be transferred either as an SDO only or as both an SDO and PDO.

For more detailed information about objects, please refer to the CANopen profiles DS 301 V4.0 and DSP 402 V1.1.

CiA DS 301

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
1000h		Device Type	Device type	Yes	④	No	-
1001h		Error Register	Group register for errors	Yes	①	No	-
1003h		Pre-defined error field	Display parameter for error code	Yes	①	No	-
	.0	Number of errors	Number of errors	Yes	①	No	-
	.1	Standard error field	Error code of error	Yes	①	No	-
1005h		COB-ID SYNC Message	Identifier of SYNC message	Yes	③ ⑥	No	-
1008h		Manufacturer Device Name	Manufacturer, device name	Yes	④	No	-
100Ah		Manufacturer Software Version	CBC software version	Yes	③	No	-
100Bh		Node-ID	Number of node on bus	Yes	P918	No	-
100Ch		Guard Time	Period between two guarding messages	Yes	③ ⑥	No	-
100Dh		Life Time Factor	Number of permissible guarding message failures until life time event	Yes	③ ⑥	No	-
100Eh		Node Guarding Identifier	Node guarding identifier	Yes	⑫ ⑥	No	-
100Fh		Number of SDOs supported	Number of implemented SDO channels	Yes	③	No	-
1014h		COB-ID Emergency Message	EMERGENCY message identifier	Yes	⑫ ⑥	No	-

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
1018		Identity Object	Identifying object	Yes		No	-
	.0	Number of entries	Number of subindices	Yes	③	No	-
	.1	Vendor-ID	Manufacturer number allocated by Cia	Yes	③	No	-
1029h		Error behaviour object	Object for configuring error reaction of bus node	Yes		No	-
	.0	No. Of error classes	Number of different error reactions	Yes	③	No	-
	.1	Communication error	Reaction of bus node to a life guarding event	Yes	④ ⑥	No	-
1200h		SDO-Parameters	Identifier for SDO messages	Yes		No	-
	.0	Number of entries	Number of subindices	Yes	③	No	-
	.1	COB-ID Client>Server	SDO request identifier	Yes	⑫ ⑥	No	-
	.2	COB-ID Server>Client	SDO response identifier	Yes	⑫ ⑥	No	-
1400h-1403h		Receive PDO Communication Parameters	Setting parameters for receive PDOs	Yes		No	-
	.0	Number of entries	Number of subindices	Yes	③	No	-
	.1	COB-ID PDO	PDO identifier	Yes	⑫ ⑥	No	-
	.2	Transmission type	Setting parameters for transmission mode	Yes	② ⑥	No	-
1600h-1603h		Receive PDO Mapping Parameters	Parameters for mapped objects in receive PDOs	Yes		No	-
	.0	Number of mapped objects in PDO	Number of mapped objects	Yes	②	No	-
	.1	First mapped object	First mapped object. Dependent on PDO selected from R_PDO list	Yes	②	No	-
	.2	Second mapped object	Second mapped object. Dependent on PDO selected from R_PDO list	Yes	②	No	-
	.3	Third mapped object	Third mapped object. Dependent on PDO selected from R_PDO list	Yes	②	No	-
	.4	Fourth mapped object	Fourth mapped object. Dependent on PDO selected from R_PDO list	Yes	②	No	-
1800h-1803h		Transmit PDO Communication Parameters	Setting parameters for transmit PDOs	Yes		No	-
	.0	Number of entries	Number of subindices	Yes	③	No	-
	.1	COB-ID PDO	PDO identifier	Yes	⑫ ⑥	No	-
	.2	Transmission type	Setting parameters for transmission mode	Yes	② ⑥	No	-

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
1A00h-1A03h		Transmit PDO Mapping Parameters	Parameters for mapped objects in transmit PDOs	Yes		No	-
	0.	Number of mapped objects in PDO	Number of mapped objects	Yes	②	No	-
	1.	First mapped object	First mapped object. Dependent on PDO selected from T_PDO list	Yes	②	No	-
	2.	Second mapped object	Second mapped object. Dependent on PDO selected from T_PDO list	Yes	②	No	-
	3.	Third mapped object	Third mapped object. Dependent on PDO selected from T_PDO list	Yes	②	No	-
	4.	Fourth mapped object	Fourth mapped object. Dependent on PDO selected from T_PDO list	Yes	②	No	-

Manufacturer-specific objects

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
2002h		Gear ratio	Speed ratio factor for slave gears in synchronous operation	Yes	⑨	Yes	9)
	.0	Number of entries	Number of subindices	Yes	③	No	-
	.1	Numerator	Gear numerator	Yes	U604.01	Yes	K3005-K3014
	.2	Denominator	Gear denominator	Yes	U604.02	Yes	K3005-K3014
2003h		Version_Parameter_Set	Parameter for storage of a parameter set in MASTERDRIVES	Yes	U017 ⑨ ⑩	No	-
2100h		Transmission Rate	Parameter for baud rate setting	Yes	P720 ⑨ ⑩	No	-
2101h		Node Number	Parameter for device address setting	Yes	P918 ⑨ ⑩	No	-
2200h	.0	Number of entries	Number of subindices	Yes	③ ⑩	No	-
	.1	Nominal speed	Nominal speed	Yes	P205	No	-
	.2	Reference speed before the decimal point	Reference speed before the decimal point	Yes	P353.01	No	-
	.3	Reference speed after the decimal point	Reference speed after the decimal point	Yes	P353.02	No	-
	.4	Norm maximum deceleration	Rated acceleration	Yes	U857	No	-

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
3001h		Free object 3001h / 16 Bit	Free 16-bit object for receiving in PDO 1	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3003
3002h		Free object 3002h / 16 Bit	Free 16-bit object for receiving in PDO 1	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3004
3003h		Free object 3003h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	K3007 K3011 K3015
3004h		Free object 3004h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	K3008 K3012 K3016
3005h		Free object 3005h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	K3007 K3011 K3015
3006h		Free object 3006h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	K3008 K3012 K3016
3007h		Free object 3007h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	K3007 K3011 K3015
3008h		Free object 3008h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	K3008 K3012 K3016
3009h		Free object 3009h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3005 K3009 K3013
300Ah		Free object 300Ah / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3006 K3010 K3014
300Bh		Free object 300Bh / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3007 K3011 K3015
300Ch		Free object 300Ch / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3008 K3012 K3016
300Dh		Free object 300Dh / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3005 K3009 K3013
300Eh		Free object 300Eh / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3006 K3010 K3014
300Fh		Free object 300Fh / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3007 K3011 K3015
3010h		Free object 3010h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3008 K3012 K3016
3011h		Free object 3011h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑩	Yes	K3006 K3010 K3014

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
3012h		Free object 3012h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑩	Yes	K3007 K3011 K3015
3013h		Free object 3013h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑩	Yes	K3008 K3012 K3016
3014h		Free object 3014h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑩	Yes	K3008 K3012 K3016
3015h		Free object 3015h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	K3005 K3009 K3013
3016h		Free object 3016h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	K3006 K3010 K3014
3017h		Free object 3017h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	K3007 K3011 K3015
3018h		Free object 3018h / 16 Bit	Free 16-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	K3008 K3012 K3016
3020h		Free object 3020h / 32 Bit	Free 32-bit object for receiving in PDO 1	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3033
3021h		Free object 3021h / 32 Bit	Free 32-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	K3037 K3041 K3045
3022h		Free object 3022h / 32 Bit	Free 32-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	K3037 K3041 K3045
3023h		Free object 3023h / 32 Bit	Free 32-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	K3037 K3041 K3045
3024h		Free object 3024h / 32 Bit	Free 32-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3035 K3039 K3043
3025h		Free object 3025h / 32 Bit	Free 32-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3037 K3041 K3045
3026h		Free object 3026h / 32 Bit	Free 32-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3035 K3039 K3043
3027h		Free object 3027h / 32 Bit	Free 32-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	K3037 K3041 K3045
3028h		Free object 3028h / 32 Bit	Free 32-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑩	Yes	K3036 K3040 K3044
3029h		Free object 3029h / 32 Bit	Free 32-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	K3035 K3039 K3043

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
302Ah		Free object 302Ah / 32 Bit	Free 32-bit object for receiving in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	K3037 K3041 K3045
3101h		Free object 3101h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑪	Yes	P734.07 P734.11 P734.15
3103h		Free object 3103h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	P734.07 P734.11 P734.15
3104h		Free object 3104h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	P734.08 P734.12 P734.16
3105h		Free object 3105h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	P734.07 P734.11 P734.15
3106h		Free object 3106h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	P734.08 P734.12 P734.16
3107h		Free object 3107h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.05 P734.09 P734.13
3108h		Free object 3108h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.06 P734.10 P734.14
3109h		Free object 3109h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.07 P734.11 P734.15
310Ah		Free object 310Ah / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.08 P734.12 P734.16
310Bh		Free object 310Bh / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.05 P734.09 P734.13
310Ch		Free object 310Ch / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.06 P734.10 P734.14
310Dh		Free object 310Dh / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.07 P734.11 P734.15
310Eh		Free object 310Eh / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.08 P734.12 P734.16
310Fh		Free object 310Fh / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑩	Yes	P734.06 P734.10 P734.14
3110h		Free object 3110h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑩	Yes	P734.07 P734.11 P734.15

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
3111h		Free object 3111h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑩	Yes	P734.08 P734.12 P734.16
3112h		Free object 3112h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	P734.05 P734.09 P734.13
3113h		Free object 3113h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	P734.06 P734.10 P734.14
3114h		Free object 3114h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	P734.07 P734.11 P734.15
3115h		Free object 3115h / 16 Bit	Free 16-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	P734.08 P734.12 P734.16
3121h		Free object 3121h / 32 Bit	Free 32-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	P734.07/.08 P734.11/.12 P734.15/.16
3122h		Free object 3122h / 32 Bit	Free 32-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩	Yes	P734.07/.08 P734.11/.12 P734.15/.16
3123h		Free object 3123h / 32 Bit	Free 32-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.05/.06 P734.09/.10 P734.13/.14
3124h		Free object 3124h / 32 Bit	Free 32-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.07/.08 P734.11/.12 P734.15/.16
3125h		Free object 3125h / 32 Bit	Free 32-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.05/.06 P734.09/.10 P734.13/.14
3126h		Free object 3126h / 32 Bit	Free 32-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑨ ⑩ ⑪	Yes	P734.07/.08 P734.11/.12 P734.15/.16
3127h		Free object 3127h / 32 Bit	Free 32-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑩	Yes	P734.06/.07 P734.10/.11 P734.14/.15
3128h		Free object 3128h / 32 Bit	Free 32-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	P734.05/.06 P734.09/.10 P734.13/.14
3129h		Free object 3129h / 32 Bit	Free 32-bit object for sending in PDO 2,3,4	Yes	② ⑤ ⑪	Yes	P734.07/.08 P734.11/.12 P734.15/.16

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
4001h		Parameter Download	Object for reading and writing all parameters in MASTERDRIVES	Yes		No	-
	.0	Number of entries	Number of subindices	Yes	③	No	-
	.1	Request identifier, index	Subindex containing the PROFIBUS request, the index and subindex of the parameter to be assigned.	Yes	All (see Subsection 8.5.3.1)	No	-
	.2	Value	Value to be assigned to the parameter (downloads only). To be read out with uploads	Yes	See Subsection 8.5.3.1	No	-
	.3	Response identifier, index	This subindex contains the response identifier to PROFIBUS for downloads or uploads	Yes	See Subsection 8.5.3.1	No	-
4040h		Technology control word	The technology can be controlled completely by this control word. The connection is 1:1 to the technology control word	Yes	⑤ ⑨	Yes	K3003
4041h		Technology status word	This status word contains all technology checkback signals. The connection is 1:1 to the technology status word	Yes	⑨	Yes	P734.02 P734.03
5001h-5FFFh		Parameters of connected device	Objects for reading and writing all parameters of the connected device	Yes	See Subsection 8.5.3.1	No	-

Device Profile
DSP 402

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
6007h		abort connection option code	Converter reaction to a lifeguarding event	Yes	④ ⑥ ⑦	No	-
6040h		Controlword	CANopen control word	Yes	⑤ ⑦	Yes	K3001 K3002
6041h		Statusword	CANopen status word	Yes	⑤ ⑦	Yes	K431 K432/ K889 ⑩ KK315 ⑨ K250
6060h		modes of operation	Object for selecting the operating mode	Yes	⑤	Yes	K3002
6061h		modes_of_operation_display	Object for displaying the selected operating mode	Yes	⑤	Yes	KK315 ⑨ K432 ⑩
6064h		position actual value	Object for displaying the current position	Yes	r185 ⑨ ⑩	Yes	KK120
6067h		position_window	Exact stop window	Yes	⑨ U501.17 U502 ⑩ U859	No	-
6068h		position_window_time	Time in exact stop window	Yes	⑨ U501.16 U502 ⑩ U864	No	-
6069h		velocity sensor actual value	Actual speed value in inc./s	Yes	r002 ⑨ P171 ⑩	No	-
606Ah		sensor_selection_code	Selection of velocity source	Yes	③ ⑨ ⑩	No	-
606Bh		velocity_demand_value	Velocity command variable	Yes	r229 ⑨ ⑩	No	-
606Ch		velocity_actual_value	Object for displaying the current velocity	Yes	r230 ⑨ ⑩	Yes	KK91
6071h		target_torque	Object for torque setpoint	Yes	⑩ U008	Yes	K3003/ K3005/ K3009/ K3013
6077h		torque_actual_value	Actual torque	Yes	⑩ r007	Yes	K0241
6078h		current_actual_value	Actual current	Yes	⑩ r004	No	-
607Ah		target_position	Target position	Yes	⑨ U015 ⑩ U874.01	Yes	KK3033 KK3035 KK3039 KK3043
607Ch		home offset	Reference point offset	Yes	⑨ U501.04 U502	No	-

Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
607Dh		software_position_limit	Software limit switch	Yes		No	-
	.0	Number of entries	Number of subindices	Yes	③	No	-
	.1	Min_position_limit	Software limit switch in negative direction	Yes	⑨ U501.12 U502 ⑩ U865.1	No	-
	.2	Max_position_limit	Software limit switch in positive direction	Yes	⑨ U501.13 U502 ⑩ U865.2	No	-
6081h		profile_velocity	Positioning travel velocity	Yes	U016 ⑨ ⑩	Yes	KK3037..
6083h		profile_acceleration	Positioning acceleration	Yes	⑨ U501.18 U502 ⑩ U873.2	Yes	-
6084h		profile_deceleration	Positioning deceleration	Yes	⑨ U501.19 U502 ⑩ U873.3	Yes	-
6086h		Motion profile type	Motion profile	Yes	③	No	-
6087h		torque_slope	Torque slope	Yes	U471.1	No	-
6088h		torque profile type	Torque profile	Yes	③ ⑩	No	-
6092h		feed_constant	Feed constant	Yes	⑦ ⑨	No	-
	.0	Number of entries	Number of subindices	Yes	③	No	-
	.1	Nominal speed	Rated velocity in P205	Yes	③ ⑥	No	-
	.2	Reference speed	Reference (homing) speed	Yes	③ ⑥	No	-
6098h		homing_method	Referencing operating mode	Yes	⑧ ⑨ ⑩	No	-
6099h		homing speeds	Referencing (homing) velocity	Yes		No	-
	.0	Number of entries	Number of subindices	Yes	③	No	-
	.1	Speed_during_search_for_switch	Reference point (home position) approach velocity	Yes	⑨ U501.07 U502 ⑩ U873.1	No	-
	.2	Speed during search for_zero	Reference creep velocity	Yes	⑨ U501.06 U502	No	-
609Ah		homing acceleration	Homing acceleration	Yes	⑩ U006	No	-
60FDh		Digital_inputs	Digital inputs	Yes	⑤ ⑩	Yes	P734.5/6...
60FEh		Digital_outputs	Digital outputs	Yes	⑤ ⑩	Yes	K3033...
60FFh		target_velocity	Target velocity	Yes	U018	Yes	KK3035..
6502h		supported drive modes	Implemented traversing operating modes	Yes	③ ⑨ ⑩	No	-
67FFh		single device type	Type of part of equipment	Yes	④ ⑨	No	-

Device profile
DS 401

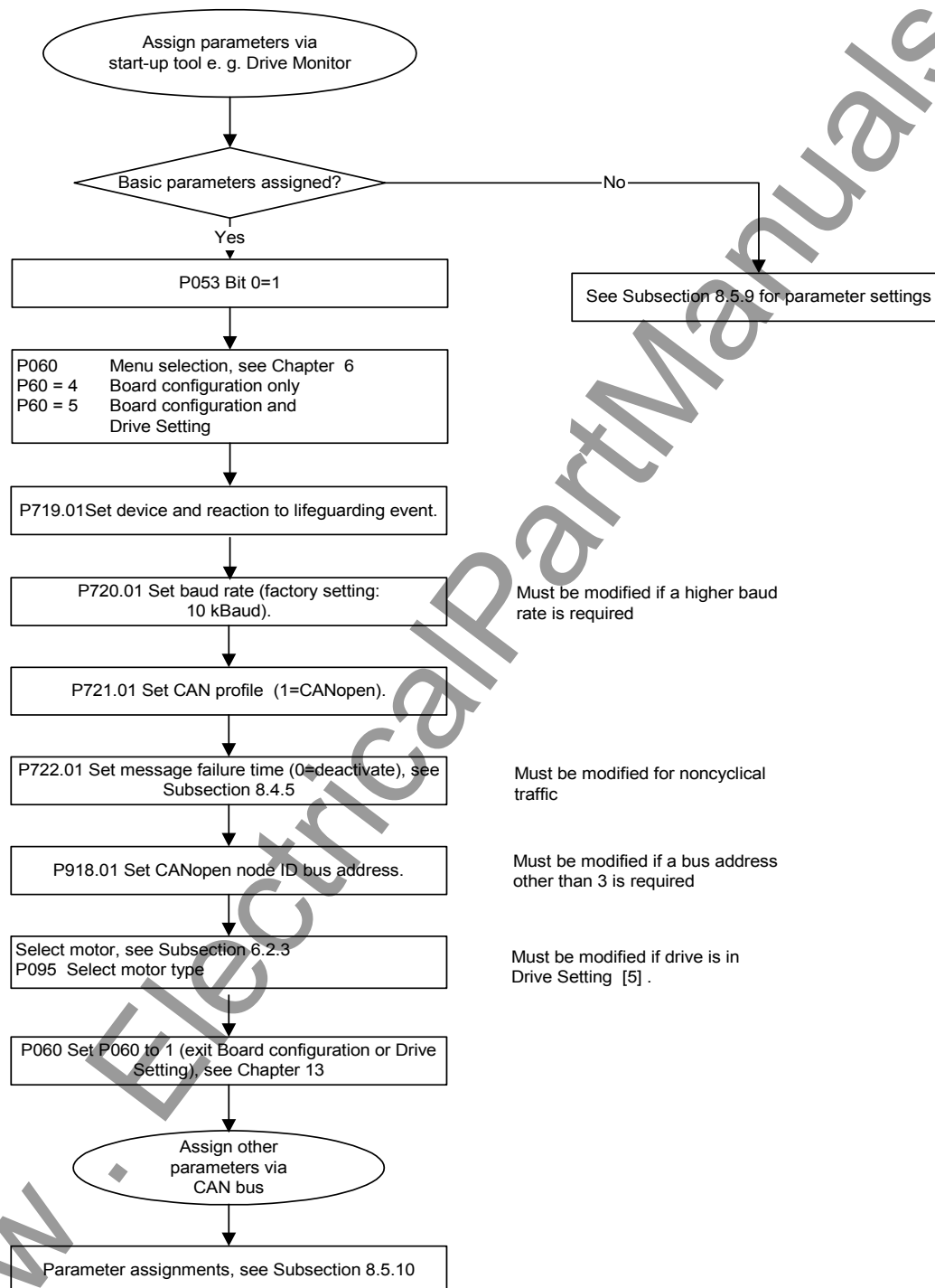
Object index	Sub-index	Object name	Description	Transmitted by SDO		Transmitted by PDO	
				Yes / No	Connector / Parameter	Yes / No	Connector / Parameter
6FFFh		single device type	Type of part of equipment	Yes	③ ⑨	No	-
6C01h		read_analogue_inp	Analog input				
	.0	number analogue inp. 16	No. of 16-bit analog inputs	Yes	③	No	-
	.1	read analogue inp. 1	Analog input 1	Yes	⑨ r637	Yes	KK011

- ① Entry is generated by the CBC through readout of connector K250
- ② Entry is generated by the CBC after parameters P711-P718 have been set
- ③ Value is permanently stored on the CBC
- ④ Entry is generated through scanning of parameter P719
- ⑤ Object can be transmitted as a PDO if a PDO in which this object is mapped is selected from the PDO table
- ⑥ Changes are stored in volatile memory on the CBC
- ⑦ Object not implemented as prescribed in the profile
- ⑧ See Subsection 8.5.7.4 "Homing mode"
- ⑨ Object connector or parameter available only when device codes 193 and 194 are entered in parameter P719
- ⑩ Object connector or parameter available only when device codes 93 and 94 are entered in parameter P719
- ⑪ Object connector or parameter available only when device code 0 is entered
- ⑫ Value is calculated as a function of the node address and stored on the CBC.

8.5.2 Commissioning the CBC

8.5.2.1 General settings

A number of settings need to be made to commission the CBC with CANopen. After the basic parameters have been assigned, the others can be set via the CAN bus using the CAN bus master or a CAN commissioning tool.



P053 (enable parameterization))

Parameter P053 (see also "Parameter List" in converter operating guide) must be set to an uneven number (e.g. 1, 3, 7 etc.). This parameter defines the sources (e.g. PMU, CBC, etc.) from which parameters may be modified. The CBC requires this enable setting to be able to execute SDO tasks.

Example: P053 = 1: Parameterization enable for CBC only
= 3: Parameterization enable for CBC+PMU
= 7: Parameterization enable for CBC+PMU+SST1 (OP)

The parameter modification enable must be set via the CBC (P053 = 1, 3 etc.) before any other parameter settings can be altered via SDO tasks from the CAN bus master.

P060 (menu selection)

P60 = 4 Select "Board configuration" function

P60 = 5 Select "Board configuration and drive setting" function

P711 (R_PDO parameter 1)**Parameter for R_PDO1**

The settings for Receive_PDO1 can be made with this parameter.

PDO communication is deactivated when the parameter is set to 0 (default setting).

Once the CB parameters have been input, the CBC checks the input against the table. If it detects a parameterization error, it activates error F80 when the system exits state 4 "Board configuration" or state 5 "Drive setting". The error value for the parameterization error is then stored in r732.1 (see Subsection 8.5.8.3). Once you have acknowledged the error, the system returns to the "Hardware configuration" state where you can correct the faulty parameter assignment.

The parameter comprises 8 bits for PDO number and eight bits for transmission type.

Transmission type								PDO No.							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PDOs 1-26 can be set in this parameter according to the device selected in parameter P719 from the list of Receive PDOs (Subsection 8.5.2.4). Not all PDOs can be selected in every set device. Impermissible selections are identified in the list.

The transmission type is specified in the following table:

Transmission type	PDO transmission				
	cyclic	Acyclic	synchr.	asynchr.	RTR only
0		X	X		
1-240	X		X		
241-251	reserved				
252			X		X
253				X	X
254				X	
255				X	

Example:

Receive PDO 23 must be received cyclically after every Sync object.

The PDO number must be converted to a hexadecimal value and entered in byte 0. The transmission type is entered as a hexadecimal value in byte 1 of P711.

The CB parameters acknowledge only decimal values. For this reason, the word, consisting of a high part (transmission type) and a low part (PDO No.) must be converted to a decimal number.

	High byte (transmission type)	Low byte (PDO number)
Decimal value	1	23
Hex value	1	17
Param. value	279 (117 hex)	

P712 (R_PDO parameter 2)**Parameter for R_PDO2**

The settings for Receive_PDO2 can be made with this parameter.

PDO communication for PDO 2 is deactivated when the parameter is set to 0 (default setting).

Once the CB parameters have been input, the CBC checks the input against the table. If it detects a parameterization error, it activates error F80 when the system exits state 4 "Board configuration" or state 5 "Drive setting". The error value for the parameterization error is then stored in r732.1 (see Subsection 8.5.8.3). Once you have acknowledged the error, the system returns to the "Board configuration" state where you can correct the faulty parameter assignment.

The parameter comprises 8 bits for PDO number and eight bits for transmission type.

Transmission type								PDO No.							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PDOs 1-72 can be set in this parameter according to the device selected in parameter P719 from the list of Receive PDOs (Subsection 8.5.2.4). Not all PDOs can be selected in every set device. Impermissible selections are identified in the list.

The transmission type is specified in the following table:

Transmission type	PDO transmission				
	cyclic	Acyclic	synchr.	asynchr.	RTR only
0		X	X		
1-240	X		X		
241-251	Reserved				
252			X		X
253				X	
254				X	
255				X	

Receive PDO 28 must be received asynchronously.

The PDO number must be converted to a hexadecimal value and entered in byte 0. The transmission type is entered as a hexadecimal value in byte 1.

The CB parameters acknowledge only decimal values. For this reason, the word, consisting of a high part (transmission type) and a low part (PDO No.) must be converted to a decimal number.

	High byte (transmission type)	Low byte (PDO number)
Decimal value	255	28
Hex value	FF	1C
Param. value	65308 (FF1C hex)	

P713 (R_PDO parameter 3)**Parameter for R_PDO3**

The settings for Receive_PDO3 can be made with this parameter.

PDO communication for R_PDO3 is deactivated when the parameter is set to 0 (default setting).

Once the CB parameters have been input, the CBC checks the input against the table. If it detects a parameterization error, it activates error F80 when the system exits state 4 "Board configuration" or state 5 "Drive setting". The error value for the parameterization error is then stored in r732.1 (see Subsection 8.5.8.3). Once you have acknowledged the error, the system returns to the "Board configuration" state where you can correct the faulty parameter assignment.

The settings are the same as in parameter P712.

P714 (R_PDO parameter 4)**Parameter for R_PDO4**

The settings for Receive_PDO4 can be made with this parameter.

PDO communication for R_PDO4 is deactivated when the parameter is set to 0 (default setting).

Once the CB parameters have been input, the CBC checks the input against the table. If it detects a parameterization error, it activates error F80 when the system exits state 4 "Board configuration" or state 5 "Drive setting". The error value for the parameterization error is then stored in r732.1 (see Subsection 8.5.8.3). Once you have acknowledged the error, the system returns to the "Board configuration" state where you can correct the faulty parameter assignment.

The settings are the same as in parameter P712.

P715 (T_PDO parameter 1)**Parameter for T_PDO1**

The settings for Transmit_PDO1 can be made with this parameter.

When the parameter is set to 0 (default setting), PDO communication is deactivated.

Once the CB parameters have been input, the CBC checks the input against the table. If it detects a parameterization error, it activates error F80 when the system exits state 4 "Board configuration" or state 5 "Drive setting". The error value for the parameterization error is then stored in r732.1 (see Subsection 8.5.8.3). Once you have acknowledged the error, the system returns to the "Board configuration" state where you can correct the faulty parameter assignment.

The parameter comprises 8 bits for PDO number and eight bits for transmission type.

Transmission type								PDO No.							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PDOs 1-22 can be set in this parameter according to the device selected in parameter P719 from the list of Transmit PDOs (Subsection 8.5.2.4). Not all PDOs can be selected in every set device. Impermissible selections are identified in the list.

The transmission type is specified in the following table:

Transmission type	PDO transmission				
	cyclic	Acyclic	synchr.	asynchr.	RTR only
0		X	X		
1-240	X		X		
241-251	reserved				
252			X		X
253				X	X
254				X	
255				X	

Example:

Receive PDO 1 must be transmitted cyclically after every Sync object.

The PDO number must be converted to a hexadecimal value and entered in byte 0. The transmission type is entered as a hexadecimal value in byte 1 in P715.

The CB parameters acknowledge only decimal values. For this reason, the word, consisting of a high part (transmission type) and a low part (PDO No.) must be converted to a decimal number.

	High byte (transmission type)	Low byte (PDO number)
Decimal value	1	1
Hex value	1	1
Param. Value	257 (101 hex)	

P716 (T_PDO parameter 2)**Parameter for T_PDO2**

The settings for Transmit_PDO2 can be made with this parameter.

When the parameter is set to 0 (default setting), PDO communication is deactivated for PDO2.

Once the CB parameters have been input, the CBC checks the input against the table. If it detects a parameterization error, it activates error F80 when the system exits state 4 "Board configuration" or state 5 "Drive setting". The error value for the parameterization error is then stored in r732.1 (see Subsection 8.5.8.3). Once you have acknowledged the error, the system returns to the "Board configuration" state where you can correct the faulty parameter assignment.

The parameter comprises 8 bits for PDO number and eight bits for transmission type.

Transmission type								PDO No.							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PDOs 1-62 can be set in this parameter according to the device selected in parameter P719 from the list of Transmit PDOs (Subsection 8.5.2.4). Not all PDOs can be selected in every set device. Impermissible selections are identified in the list.

The transmission type is specified in the following table:

Transmission type	PDO transmission				
	cyclic	Acyclic	synchr.	asynchr.	RTR only
0		X	X		
1-240	X		X		
241-251	Reserved				
252			X		X
253				X	
254				X	
255				X	

Example:

Receive PDO 128 must be received asynchronously.

The PDO number must be converted to a hexadecimal value and entered in byte 0. The transmission type is entered as a hexadecimal value in byte 1 in P715.

The CB parameters acknowledge only decimal values. For this reason, the word, consisting of a high part (transmission type) and a low part (PDO No.) must be converted to a decimal number.

	High byte (transmission type)	Low byte (PDO number)
Decimal value	255	28
Hex value	FF	1C
Param. Value	65308 (FF1C hex)	

P717 (T_PDO parameter 3)**Parameter for T_PDO3**

The settings for Transmit_PDO3 can be made with this parameter.

When the parameter is set to 0 (default setting), PDO communication is deactivated for PDO3.

Once the CB parameters have been input, the CBC checks the input against the table. If it detects a parameterization error, it activates error F80 when the system exits state 4 "Board configuration" or state 5 "Drive setting". The error value for the parameterization error is then stored in r732.1 (see Subsection 8.5.8.3). Once you have acknowledged the error, the system returns to the "Board configuration" state where you can correct the faulty parameter assignment.

The settings are the same as in parameter P716.

P718 (T_PDO parameter 4)**Parameter for T_PDO4**

The settings for Transmit_PDO4 can be made with this parameter.

When the parameter is set to 0 (default setting), PDO communication is deactivated for PDO4.

Once the CB parameters have been input, the CBC checks the input against the table. If it detects a parameterization error, it activates error F80 when the system exits state 4 "Board configuration" or state 5 "Drive setting". The error value for the parameterization error is then stored in r732.1 (see Subsection 8.5.8.3). Once you have acknowledged the error, the system returns to the "Board configuration" state where you can correct the faulty parameter assignment.

The settings are the same as in parameter P716.

P719 (CB parameter 9)															
Device on CAN bus and reaction to life guarding event The device in which the CBC is inserted is identified to the CBC by this parameter. It also defines the reaction of the device and the CAN node. When the parameter is set to 0 (default), the general CANopen device is selected on the CAN bus.															
Drive reaction to a life guarding event (device behavior, object 6007h)				Bus node reaction to a life guarding event (communication behavior, object 1029h Ind 1)				Board (device identifier)							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Definition of device in which the CBC is inserted (device identifier): The device in which the CBC is inserted is specified in bits 0-7. The values must be taken from the table below and entered in the low word of the parameter in hexadecimal code. This parameter must be set before PDOs are mapped in parameters P711-P718 to ensure that the CBC accesses the correct mapping table.															
Board code				Device				Display in object 1008h Device name							
0				General CANopen device				CANA							
071				SIMOREG DC MASTER				SG 70							
092 (not selectable)				MASTERDRIVES VC (CUVC)				MDVV							
093				MASTERDRIVES MC				MDMC							
193				MASTERDRIVES MC F01				MCF01							
094				MASTERDRIVES MC Kompakt Plus				MDMP							
194				MASTERDRIVES MC Kompakt Plus F01				MPF01							
Reaction to life guarding event (communication error behavior) The reaction of the bus node to a life guarding event must be entered in bits 8-11.															
Binary value for P719 bits 8-11				Reaction to a communication error											
0 0 0 0				Pre-operational											
0 0 0 1				No state change											
0 0 1 0				Stopped											
0 0 1 1 – 1 1 1 1				Reserved / not permissible											
Drive reaction to a life guarding event (device error behavior) If life guarding is activated, you can configure the drive reaction to this type of event. The following table lists the options that must be entered in bits 12-15.															

P719 (CB parameter 9)			
Binary value for P719 bits 12-15	Response to life guarding event		
0 0 0 0	Remains in current state		
0 0 0 1	Shutdown to ready to start		
0 0 1 0	Change to Not Ready to Switch On		
0 0 1 1	Quick stop to Not Ready to Switch On		
0 1 0 0 - 1 1 1 1	Reserved / Not permissible		
If a life guarding event occurs, the converter reacts as configured in P719. An emergency message with error code 8130 Life Guard Error is also transmitted. If P719 is parameterized illegally, error F080 is displayed when the system exits state 4 "Board configuration" or 5 "Drive setting". Once you have acknowledged the error, the system returns to the "Board configuration" or "Drive setting" state where you can correct the faulty parameter assignment. Example: A MASTERDRIVES with technology option F01 must be configured as a CANopen slave. In response to a life guarding event, the drive must decelerate with OFF 1 and the bus node must not change its status. The hexadecimal value must be converted to a decimal value and entered in the parameter.			
	Bit 12-15 (device behavior)	Bit 8-11 (communication behavior)	Bit 0-7 (device identifier)
Decimal value	1	1	193
Hex value	1	1	C1
Param. value	4545 (11C1 hex)		

P720 (CB parameter 10)**Baud rate of slave on CAN bus**

This parameter sets the baud rate of the slave on the CAN bus according to the following table:

If the baud rate is outside the valid range, error F080 is displayed when the system exits state 4 "Board configuration". Once you have acknowledged the error, the system returns to the "Board configuration" state where you can correct the faulty parameter assignment.

Internal defaults for bus timing as a function of baud rate:

Parameter value	0	1	2	3	4	5	6	7	8
Baud rate [kbit/s]	10	20	50	100	125	250	500	800	1000

Baud rate	BRP	SJW	TSEG1	TSEG2	Hex value
10 kbit (P720 = 0)	39	2	15	2	2FA7
20 kbit (P720 = 1)	19	2	15	2	2F93
50 kbit (P720 = 2)	7	2	15	2	2F87
100 kbit (P720 = 3)	3	2	15	2	2F83
125 kbit (P720 = 4)	3	1	12	1	1C43
250 kbit (P720 = 5)	1	1	12	1	1C41
500 kbit (P720 = 6)	0	1	12	1	1C40
800 kbit (P720 = 7)	0	1	6	1	1640
1 MBit (P720 = 8)	0	1	4	1	1440

P721 (CB parameter 11)**Special CAN bus settings**

- **Index i001:** This parameter can be set to switch between CAN layers 2=0 and 7=1 (CANopen).

P918.1 (CBC bus address)

The node address of the device on the CAN bus is set in this parameter.

The default setting is 3, thereby providing the MASTERDRIVES with a valid bus address. It can be addressed directly under node ID 3.

NOTE

After the basic parameters have been set as specified in the flowchart at the beginning of this section, other parameters can be set via the CAN bus.

8.5.2.2 NMT state

The drive switches automatically to the "Initialization" status when voltage is connected to the control board. It switches to "Pre-operational" after initialization. In the "Pre-operational" state, the drive can be configured by means of SDOs and commissioned. **It cannot receive or transmit PDOs in this state.**

The drive is switched to the "Operational" state with NMT message "Start Remote Node". The drive is fully functional in this state.

In the "Stopped" state, the drive cannot be operated via the bus. It can only be switched out of this state again by NMT message "Enter pre-operational state" (SDO only) or "Start_Remote_Node" (PDO and SDO).

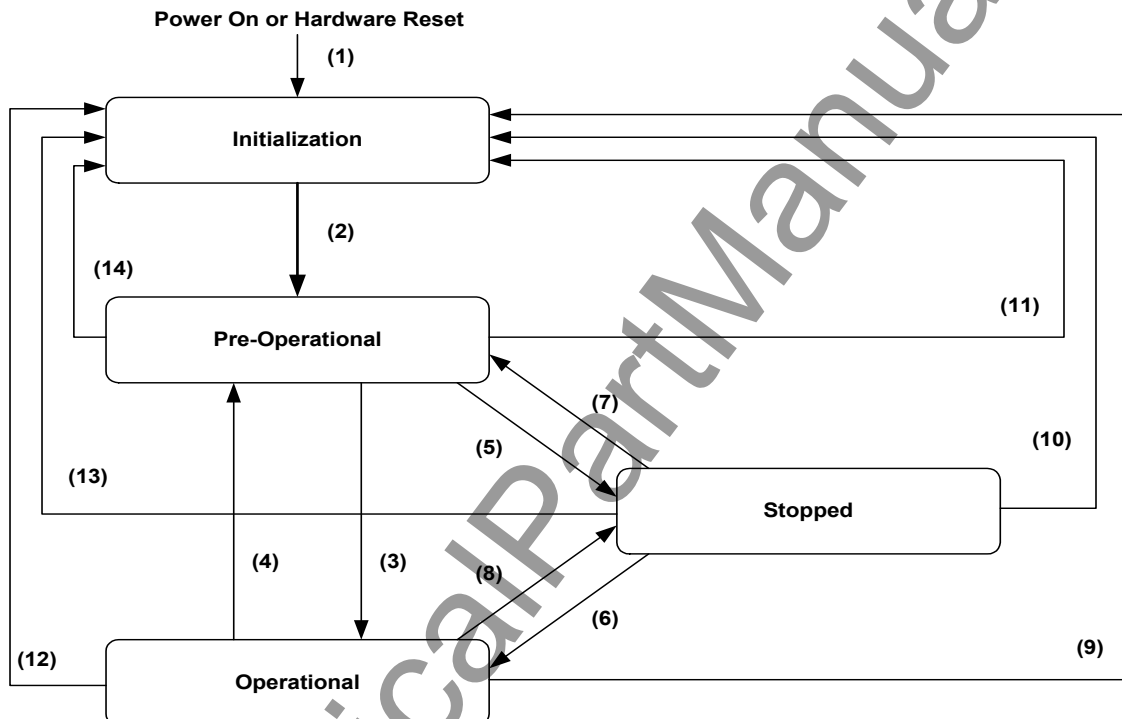


Fig. 8.5-1 State diagram of a device

(1)	The drive automatically switches to the initialization state at Power ON
(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

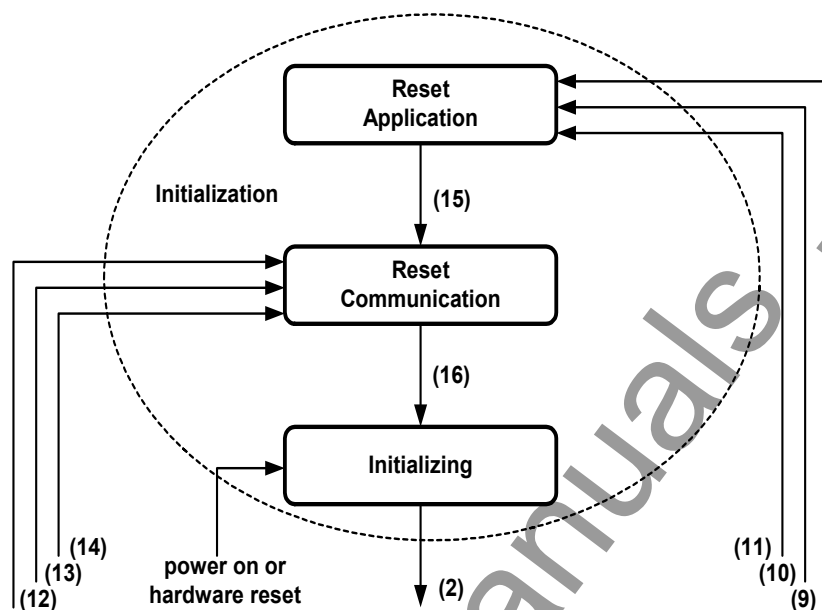


Fig. 8.5-2 Structure of the initialization state

(2)	Initialization finished – enter PRE-OPERATIONAL automatically
(12), (13), (14)	Reset_Communication indication
(9), (10), (11)	Reset_Node indication
(15)	Application Reset performed
(16)	Communication Reset performed

The pulses on the drive are disabled by an NMT command Reset Application. This causes the motor to coast to a standstill. The CBC then ceases to operate the heartbeat counter, causing activation of error F81. This is acknowledged by the CBC and the drive is then re-initialized. The re-initialization operation sets all objects specific to the manufacturer and device to the value following "Voltage On". The drive then switches to the Reset Communication state. All communication objects are reset to the default value. After initialization, the drive returns to the pre-operational state again (does not apply to general CANopen device).

	INITIALIZING	PRE-OPERATIONAL	OPERATIONAL	STOPPED
PDO			X	
SDO		X	X	
Synchronization Object			X	
Emergency Object		X	X	
Boot-Up Object	X			
Network Management Object		X	X	X

8.5.2.3 Relation between PDO/PZD and SDO/PKW

In the CANopen profile, every object can be read or written with an SDO task. This applies in the case of MASTERDRIVES only if the correct interconnections have been made. The last two columns in the table of objects specify the transmission options and the associated parameters or connectors.

MASTERDRIVES recognizes two transmission modes in connection with PROFIBUS, i.e. the PKW (parameter identifier value) task and PZD (process data).

A PKW task can be used to read or write MASTERDRIVES parameters. This type of task consists of a parameter number, a task identifier and a value.

PZD such as setpoints and actual values are updated cyclically with Profibus. Process data are 'wired' directly from the CB board to the target parameter via a dual port RAM channel. For this reason, they do not require addressing.

PZD values cannot be written by means of a PKW task nor are PZD capable of accessing parameters.

A CANopen SDO task is directly comparable to a PKW task.

PDOs correspond to PZD in the PROFIBUS.

All objects can be transmitted per SDO with CANopen.

Fixed setpoints are used as a means of writing process data in MASTERDRIVES via an SDO task. The connectors of the fixed setpoints must be wired to the corresponding setpoint parameters.

SDO tasks

SDO tasks are sent via identifiers 600h + NodeID (Client>Server) and 580h + NodeID (Server>Client).

If you want to send a DSP 402 object as simply an SDO from the CANopen master when it corresponds to a process data in MASTERDRIVES and has not been mapped into a PDO, then it is not transmitted as normal via the dual port RAM, but diverted via a fixed setpoint. When the SDO is addressed via the parameter channel of the CBC, the setpoint is set to the possibly re-normalized value which is stored in the SDO (see table of objects, Parameters / Connectors column).

The outgoing connector for the fixed setpoint must be "wired" up to the MASTERDRIVES location to which the setpoint must be applied.

Example

The object 60FFh target_velocity may only be transmitted as an SDO via the bus. To do this, proceed as follows:

Find the fixed setpoint (U018) to which the SDO of the object writes in the table of objects. Then take the connector (KK0418) belonging to the fixed setpoint and connect it up to parameter P212 (Src Ctrl Setp).

All objects that cannot be transferred as PDOs according to the table are MASTERDRIVES parameters that can only be transmitted as SDOs.

Parameters are stored on the CBC for DSP 402 objects which supply actual values. These are read out in the case of an SDO read task. Nothing needs to be "wired up" for these.

If an object is transferred as a PDO as a result of the predefined mapping table, the value converted or processed for the MC may emerge at the connector number of the dual port RAM specified in the table. This connector must in turn be "wired" to the right location in MASTERDRIVES. If an object is mapped to a PDO and thereby transferred via the dual port RAM, then it can also be written by means of an SDO task.

CAUTION

All parameter values modified by means of profile-specific objects are saved only to the RAM in MASTERDRIVES. If the control board of the MASTERDRIVES system is disconnected from the supply voltage, the values from the EEPROM are stored in the parameters and in the relevant objects when the supply is connected again. The R_PDO values are all set to zero.

To restore the object values as they were before the supply was disconnected, the objects must be written with their PDO or SDO tasks.

SDO aborts

Some objects are linked to parameters in the MASTERDRIVES. The CBC uses a parameter task to read or write them.

However, some parameters can be written only when the converter is in particular states.

If you want write an object that is linked to a parameter that can only be written in the "ready to start" state during operation via the CAN bus, an SDO abort is returned in response to the SDO task. The different SDO aborts are described in the following table.

Abort name	Abortcode	Reason (PKW error values)	Description
SERVICE_ERR	0x08000022	<u>Response identifier</u> (7) Task cannot be executed (with error number) <u>Error number</u> (17) Task cannot be executed in current operating status	Currently issued task cannot be executed in active converter state.
UNSUP_ACC	0x06010000	SDO access to a free object that is not mapped to a PDO Write access to objects that are read only or read access to objects that are write only With parameter tasks <u>Response identifier</u> (7) Task cannot be executed (with error number) <u>Error number</u> (1) Parameter value cannot be modified (6) Setting not permitted (resetting only) (7) Descriptive element cannot be modified (15) Text array does not exist (102) Channel width too small (106) Task not implemented (107) Text cannot be modified	The free object does not exist in the object directory until it has been mapped to a PDO. If the parameter is a visualization parameter Basically impossible for MASTERDRIVES Specific to MASTERDRIVES: For PKW short channels only
VAL_RANGE_EXC	0x06090030	<u>Response identifier</u> (7) Task cannot be executed (with error number) <u>Error number</u> (2) Lower or upper value limit exceeded	Parameter limits exceeded

Abort name	Abortcode	Reason (PKW error values)	Description
General Error	0x08000000	Incorrect parameter states: Parameter check buffer or parameter status buffer, in both directions New parameter task sent before response to previous task had been received	
DATA_TRANSFER_ERR	0x08000020	All other PKW errors <u>Response identifier</u> (7) Task cannot be executed	For all other PKW errors, see Errors, PKW task table.
Data cannot be transferred or stored to the application because of local control	0x08000021	<u>Response identifier</u> (7) Task cannot be executed <u>Error number</u> (11) No control command source status (12) Keyword missing (101) Parameter number currently deactivated <u>Response identifier</u> (8) Control command source status not assigned to PKW interface	Device parameters: Access key and/or param. special access not appropriately set Specific to MASTERDRIVES
Data type does not match, length of service parameter does not match, service parameter too high or too low	0x06070010	<u>Response identifier</u> (7) Task cannot be executed <u>Error number</u> (5) Incorrect data type	On access to a MASTERDRIVES word parameter with a double word identifier and vice versa
Sub-Index does not exist	0x06090011	SDO access to a non-existent subindex of an object <u>Response identifier</u> (7) Task cannot be executed <u>Error number</u> (3) Errored subindex (4) No array	On access to a MASTERDRIVES parameter
Data type does not match, length of service parameter too high	0x06070012	Attempt to write an excessively high value to an SDO	
Data type does not match, length of service parameter too low	0x06070013	Attempt to write an excessively low value to an SDO	

Abort name	Abortcode	Reason (PKW error values)	Description
Toggle Bit not alternated	0x05030000	Toggle bit is not toggled with a nonexpedited transfer.	
SDO protocol timed out	0x05040000		The MASTERDRIVES has not responded to a transmitted parameter task within 150 ms. The CBC aborts the parameter task and signals an abort
Object does not exist in the object dictionary	0x06020000	Access to a non-existent object <u>Response identifier</u> (7) Task cannot be executed <u>Error number</u> (0) Illegal parameter number	Access to a non-existent MASTERDRIVES parameter
General parameter incompatibility reason	0x06040043	<u>Response identifier</u> (7) Task cannot be executed <u>Error number</u> (104) Illegal parameter value	MASTERDRIVES-specific

8.5.2.4 PDO mapping

PDO mapping is possible only to a limited extent in MASTERDRIVES. The following table lists all the available premapped PDOs. Free mapping as described in CANopen cannot be implemented with the CBC and CANopen.

The mapped objects are stored in objects 1600h-1603h and 1A00h-1A03h and can be read out via the CAN bus.

CAUTION

The manufacturer-specific free objects 3xxxh can be addressed via an SDO only if they have been mapped to the dual port RAM as a PDO!

How to map PDOs

To be able to enter values in parameters P711-P718, the MASTERDRIVES must be switched to Drive Setting (P060 = 5) or Board Configuration (P060 = 4).

Search through the table until you find the most suitable premapped PDO for your application. The first column contains a number. Enter this number, for example, in parameter P711, byte 0, as a hexadecimal value. Enter the CANopen value for the PDO transmission (Transmission Type) in byte 1, also as a hexadecimal value. This value must now be converted to a decimal number since MASTERDRIVES permits only decimal CB values.

Please note that certain PDOs can only be entered in particular parameters. The selectable parameters are listed in the last column of the table. The first PDO must always contain the control word. A special interconnection specification, which is shown in the block diagrams (Subsection 8.5.12), has been defined for this purpose.

Once the CBC parameters have been set, the values of the receive PDOs must be "wired up" to the correct MASTERDRIVES location via the CBC receive connectors.

The connectors with the values for the PDOs must be wired to the correct location in the dual port RAM for the send data (P734).

PDOs which are parameterized in P711 and P715 can also be parameterized in P712-14 and P716-18. It is therefore possible to send a control word by two different transmission modes, for example, cyclical and asynchronous.

NOTE

The values of the objects are only ever transferred to one connector. Objects mapped as PDOs have priority, i.e. if an object is mapped to a PDO, the value is transferred to CB receive parameter K3xxx or KK3xxx, even if the object has been transferred as an SDO task. The U parameter that would be addressed without PDO mapping is not written in this instance.

If an object is written to two PDOs, as described above, the object value is transferred to the receive word mapped to the parameter with the lower number.

**Table of receive
PDOs**

R_PDO No.	Mapping object index	Mapping object name	DPR connector	R_PDO number in DPR	R_PDO identifier	When selected enter in parameters
1	6040h	Controlword	K3001	1	200h+NodeID	711/712/713/714
2	6040h 6060h	Controlword Modes of operation	K3001 K3002	1	200h+NodeID	711/712/713/714
3 ① ②	6040h 607Ah	Controlword Target_position	K3001 K3033	1	200h+NodeID	711/712/713/714
4 ① ②	6040h 60FFh	Controlword Target_velocity	K3001 K3033	1	200h+NodeID	711/712/713/714
5 ②	6040h 6071h	Controlword Target_torque	K3001 K3003	1	200h+NodeID	711/712/713/714
6		Reserve				
7 ②	6040h 60FEh	Controlword Digital_outputs	K3001 KK3033	1	200h+NodeID	711/712/713/714
8-21		Reserve				
18 ①	6040h 4040h	Controlword Technology Controlword	K3001 K3002/3003	1	200h+NodeID	711/712/713/714
19 ①	6040h 6060h 4040h	Controlword Modes of Operation Technology Controlword	K3001 K3002 KK3033	1	200h+NodeID	711/712/713/714
20 ③	6040h 6060h 3001h 3002h	Controlword Modes of Operation Free object 3001h 16 Bit Free object 3002h 16 Bit	K3001 K3002 K3003 K3004	1	200h+NodeID	711/712/713/714
21 ③	6040h 3001h	Controlword Free object 3001h 16 Bit	K3001 K3003	1	200h+NodeID	711/712/713/714
22	6040h 3001h 3002h	Controlword Free object 3001h/16 Bit Free object 3002h/16 Bit	K3001 K3003 K3004	1	200h+NodeID	711/712/713/714
23	6040h 3020h	Controlword Free object 3020h/32 Bit	K3001 K3033	1	200h+NodeID	711/712/713/714

R_PDO No.	Mapping object index	Mapping object name	DPR connector	R_PDO number in DPR	R_PDO identifier	When selected enter in parameters
24	6040h 6060h 3001h	Controlword Modes_of_operation Free object 3001h 16 Bit	K3001 K3002 K3003	1	200h+NodeID	711/712/713/714
25	6040h 6060h 3020h	Controlword Modes_of_operation Free object 3020h 32 Bit	K3001 K3002 K3033	1	200h+NodeID	711/712/713/714
26 ① ②	6040h 6081h	Controlword Profile Velocity	K3001 K3033	1	200h+NodeID	711/712/713/714
27 ① ②	60FFh	Target_velocity	K3035 K3039 K3043	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
28 ① ②	60FFh 3003h	Target_velocity Free object 3003h/16 Bit	K3035/K3039/ K3043 K3007/K3011/ K3015	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
29 ① ②	60FFh 3003h 3004h	Target_velocity Free object 3003h/16 Bit Free object 3004h/16 Bit	K3035/K3039/ K3043 K3007/K3011/ K3015 K3008/K3012/ K3016	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
30 ① ②	60FFh 3021h	Target_velocity Free object 3021h/32 Bit	K3035/K3039/ K3043 K3037/3041/ K3045	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
31 ① ②	607Ah	Target_position	K3035/K3039/ K3043	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
32 ① ②	607Ah 6081h	Target_position Profile_velocity	K3035/K3039/ K3043 K3037/K3041/ K3045	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
33 ① ②	607Ah 3005h	Target_position Free object 3005h/16 Bit	K3035/K3039/ K3043 K3007/K3011/ K3015	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
34 ① ②	607Ah 3005h 3006h	Target_position Free object 3005h/16 Bit Free object 3006h/16 Bit	K3035/K3039/ K3043 K3007/K3011/ K3015 K3008/K3012/ K3016	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714

R_PDO No.	Mapping object index	Mapping object name	DPR connector	R_PDO number in DPR	R_PDO identifier	When selected enter in parameters
35 ① ②	607Ah 3022h	Target_position Free object 3022h/32 Bit	K3035/K3039/ K3043 K3037/K3041/ K3045	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
36 ① ②	6081h	Profile_velocity	K3035/K3039/ K3043	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
37 ① ②	6081h 3007h	Profile_velocity Free object 3007h/16 Bit	K3035/K3039/ K3043 K3007/K3011/ K3015	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
38 ① ②	6081h 3007h 3008h	Profile_velocity Free object 3007h/16 Bit Free object 3008h/16 Bit	K3035/K3039/ K3043 K3007/K3011/ K3015 K3008/K3012/ K3016	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
39 ① ②	6081h 3023h	Profile_velocity Free object 30023h/32 Bit	K3035/K3039/ K3043 K3037/K3041/ K3045	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
40 ②	6083h 6084h	Profile acceleration Profile deceleration	K3005/..9/..13 K3007/..11/..15	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
41	3009h	Free object 3009h/16 Bit	K3005/K3009/ K3013	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
42	3009h 300Ah	Free object 3009h/16 Bit Free object 300Ah/16 Bit	K3005/K3009/ K3013 K3006/K3010/ K3014	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
43	3009h 300Ah 300Bh	Free object 3009h/16 Bit Free object 300Ah/16 Bit Free object 300Bh/16 Bit	K3005/K3009/ K3013 K3006/K3010/ K3014 K3007/K3011/ K3015	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714

R_PDO No.	Mapping object index	Mapping object name	DPR connector	R_PDO number in DPR	R_PDO identifier	When selected enter in parameters
44	3009h	Free object 3009h/16 Bit	K3005/K3009/ K3013	2	300h+NodeID 400h+NodeID 500h+NodeID	712
	300Ah	Free object 300Ah/16 Bit	K3006/K3010/ K3014	3		713
	300Bh	Free object 300Bh/16 Bit	K3007/K3011/ K3015	4		714
	300Ch	Free object 300Ch/16 Bit	K3008/K3012/ K3016			
45	3024h	Free object 3024h/32 Bit	K3035/K3039/ K3043	2	300h+NodeID 400h+NodeID 500h+NodeID	712
				3		713
				4		714
46	3024h	Free object 3024h/32 Bit	K3035/K3039/ K3043	2	300h+NodeID 400h+NodeID 500h+NodeID	712
				3		713
				4		714
47	3024h	Free object 3024h/32 Bit	K3035/K3039/ K3043	2	300h+NodeID 400h+NodeID 500h+NodeID	712
				3		713
				4		714
48	3024h	Free object 3024h/32 Bit	K3035/K3039/ K3043	2	300h+NodeID 400h+NodeID 500h+NodeID	712
				3		713
				4		714
49	300Dh	Free object 300Dh/16 Bit	K3005/K3009/ K3013	2	300h+NodeID 400h+NodeID 500h+NodeID	712
				3		713
				4		714
50	300Dh	Free object 300Dh/16 Bit	K3005/K3009/ K3013	2	300h+NodeID 400h+NodeID 500h+NodeID	712
				3		713
				4		714
51	300Dh	Free object 300Dh/16 Bit	K3005/K3009/ K3013	2	300h+NodeID 400h+NodeID 500h+NodeID	712
				3		713
				4		714

R_PDO No.	Mapping object index	Mapping object name	DPR connector	R_PDO number in DPR	R_PDO identifier	When selected enter in parameters
52	300Dh	Free object 300Dh/16 Bit	K3005/K3009/ K3013	2	300h+NodeID 400h+NodeID 500h+NodeID	712
	300Eh	Free object 300Eh/16 Bit	K3006/K3010/ K3014	3		713
	300Fh	Free object 300Fh/16 Bit	K3007/K3011/ K3015	4		714
	3010h	Free object 3010h/16 Bit	K3008/K3012/ K3016			
53	3026h	Free object 3026h/32 Bit	K3035/K3039/ K3043	2	300h+NodeID 400h+NodeID 500h+NodeID	712
				3		713
				4		714
54	3026h	Free object 3026h/32 Bit	K3035/K3039/ K3043	2	300h+NodeID 400h+NodeID 500h+NodeID	712
				3		713
				4		714
55	3026h	Free object 3026h/32 Bit	K3035/K3039/ K3043	2	300h+NodeID 400h+NodeID 500h+NodeID	712
	300Fh	Free object 300Fh/16 Bit	K3007/K3011/ K3015	3		713
				4		714
	3010h	Free object 3010h/16 Bit	K3008/K3012/ K3016			
56	3026h	Free object 3026h/32 Bit	K3035/K3039/ K3043	2	300h+NodeID 400h+NodeID 500h+NodeID	712
				3		713
	3027h	Free object 3027h/32 Bit	K3037/K3041/ K3045	4		714
57 ① ②	2002.01h	Gear ratio numerator	K3005/K3009/ K3013	2	300h+NodeID 400h+NodeID 500h+NodeID	712
	2002.02h	Gear ratio denominator	K3006/K3010/ K3014	3		713
58 ②	6071h	Target torque	K3005/ K3009/ K3013	4	300h+NodeID 400h+NodeID 500h+NodeID	714
59 ②	6071h	Target torque	K3005/ K3009/ K3013	2	300h+NodeID 400h+NodeID 500h+NodeID	712
	3011h	Free object 3011h 16 Bit	K3006/ K3010/ K3014	3		713
60 ②	6071h	Target torque	K3005/ K3009/ K3013	4	300h+NodeID 400h+NodeID 500h+NodeID	714
60 ②	3011h	Free object 3011h 16 Bit	K3006/ K3010/ K3014	4	300h+NodeID 400h+NodeID 500h+NodeID	714
60 ②	3012h	Free object 3012h 16 Bit	K3007/ K3011/ K3015	4	300h+NodeID 400h+NodeID 500h+NodeID	714

R_PDO No.	Mapping object index	Mapping object name	DPR connector	R_PDO number in DPR	R_PDO identifier	When selected enter in parameters
61 ②	6071h 3011h 3012h 3013h	Target torque Free object 3011h 16 Bit Free object 3012h 16 Bit Free object 3013h 16 Bit	K3005/ K3009/ K3013 K3006/ K3010/ K3014 K3007/ K3011/ K3015 K3008/ K3012/ K3016	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
62 ②	6071h 3028h	Target torque Free object 3028h 32 Bit	K3005/ K3009/ K3013 KK3036/ KK3040/ KK3044	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
63 ②	6071h 3028h 3014h	Target torque Free object 3028h 32 Bit Free object 3014h 16 Bit	K3005/ K3009/ K3013 KK3036/ KK3040/ KK3044 K3008/ K3012/ K3016	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
64 ②	60FEh	Digital outputs	KK3035/ KK3039/ KK3043	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
65 ③	3015h	Free object 3015h/16 Bit	K3005/K3009/ K3013	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
66 ③	3015h 3016h	Free object 3015h/16 Bit Free object 3016h/16 Bit	K3005/K3009/ K3013 K3006/K3010/ K3014	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
67 ③	3015h 3016h 3017h	Free object 3015h/16 Bit Free object 3016h/16 Bit Free object 3017h/16 Bit	K3005/K3009/ K3013 K3006/K3010/ K3014 K3007/K3011/ K3015	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
68 ③	3015h 3016h 3017h 3018h	Free object 3015h/16 Bit Free object 3016h/16 Bit Free object 3017h/16 Bit Free object 3018h/16 Bit	K3005/K3009/ K3013 K3006/K3010/ K3014 K3007/K3011/ K3015 K3008/K3012/ K3016	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714

R_PDO No.	Mapping object index	Mapping object name	DPR connector	R_PDO number in DPR	R_PDO identifier	When selected enter in parameters
69 ③	3029h	Free object 3029h/32 Bit	K3035/K3039/ K3043	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
70 ③	3029h 3017h	Free object 3029h/32 Bit Free object 3017h/16 Bit	K3035/K3039/ K3043 K3007/K3011/ K3015	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
71 ③	3029h 3017h 3018h	Free object 3026h/32 Bit Free object 3017h/16 Bit Free object 3018h/16 Bit	K3035/K3039/ K3043 K3007/K3011/ K3015 K3008/K3012/ K3016	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714
72 ③	3029h 302Ah	Free object 3029h/32 Bit Free object 302Ah/32 Bit	K3035/K3039/ K3043 K3037/K3041/ K3045	2 3 4	300h+NodeID 400h+NodeID 500h+NodeID	712 713 714

- ① PDO can be selected only when device codes 193 and 194 are entered in parameter P719
- ② PDO can be selected only when device codes 93 and 94 are entered in parameter P719
- ③ PDO can be selected only when device code 0 is entered in parameter P719

**Table of transmit
PDOs**

T_PDO No.	Mapping object index	Mapping object name	DPR parameter	T_PDO number in DPR	T_PDO identifier	When selected enter in parameters
1	6041h	Statusword	P734.01-0.4	1	180h+NodeID	715/716/717/ 718
2	6041h 6061h	Statusword Modes_of_operation _display	P734.01-04	1	180h+NodeID	715/716/717/ 718
3 ③	6041h 3101h	Statusword Free object 3107h 16 Bit	P734.01 P734.03	1	180h+NodeID	715/716/717/ 718
4 ③	6041h 6061h 3101h	Statusword Modes of operation display Free object 3107h 16 Bit	P734.01 P734.02 P734.03	1	180h+NodeID	715/716/717/ 718
5-20		Reserve				

T_PDO No.	Mapping object index	Mapping object name	DPR parameter	T_PDO number in DPR	T_PDO identifier	When selected enter in parameters
21	6041h 4061h	Statusword technology statusword	P734.01-04	1	180h+NodeID	715/716/717/ 718
22	6041h 6061h 4041	Statusword Modes_of_operation _display technology statusword	P734.01-04	1	180h+NodeID	715/716/717/ 718
23 ① ②	606Ch	Velocity actual value	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
24 ① ②	606Ch 3103h	Velocity actual value Free object 3103h 16 Bit	P734.05/.06 P734.09/.10 P734.13/.14 P734.07 P734.11 P734.15	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
25 ① ②	606Ch 3103h 3104h	Velocity actual value Free object 3103h 16 Bit Free object 3104h 16 Bit	P734.05/.06 P734.09/.10 P734.13/.14 P734.07 P734.11 P734.15 P734.08 P734.12 P734.16	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
26 ① ②	606Ch 3121h	Velocity actual value Free object 3121h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14 P734.07/.08 P734.11/.12 P734.15/.16	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
27 ① ②	6064h	Position actual value	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
28 ① ②	6064h 3105h	Position actual value Free object 3105h 16 Bit	P734.05/.06 P734.09/.10 P734.13/.14 P734.07 P734.11 P734.15	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
29 ①	6064h 6C01h	Position actual value Read Analog Input	P734.05/.06 P734.09/.10 P734.13/.14 P734.07 P734.11 P734.15	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718

T_PDO No.	Mapping object index	Mapping object name	DPR parameter	T_PDO number in DPR	T_PDO identifier	When selected enter in parameters
30 ① ②	6064h	Position actual value	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	3105h	Free object 3105h 16 Bit	P734.07 P734.11 P734.15			
	3106h	Free object 3106h 16 Bit	P734.08 P734.12 P734.16			
31 ① ②	6064h	Position actual value	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	3122h	Free object 3122h 32 Bit	P734.07/.08 P734.11/.12 P734.15/.16			
32	3107h	Free object 3107h 16 Bit	P734.05 P734.09 P734.13	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
33	3107h	Free object 3107h 16 Bit	P734.05 P734.09 P734.13	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	3108h	Free object 3108h 16 Bit	P734.06 P734.10 P734.14			
34	3107h	Free object 3107h 16 Bit	P734.05 P734.09 P734.13	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	3108h	Free object 3108h 16 Bit	P734.06 P734.10 P734.14			
	3109h	Free object 3109h 16 Bit	P734.07 P734.11 P734.15			
35	3107h	Free object 3107h 16 Bit	P734.05 P734.09 P734.13	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	3108h	Free object 3108h 16 Bit	P734.06 P734.10 P734.14			
	3109h	Free object 3109h 16 Bit	P734.07 P734.11 P734.15			
	310Ah	Free object 310Ah 16 Bit	P734.08 P734.12 P734.16			
36	3123h	Free object 3123h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718

T_PDO No.	Mapping object index	Mapping object name	DPR parameter	T_PDO number in DPR	T_PDO identifier	When selected enter in parameters
37	3123h	Free object 3123h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	3109h	Free object 3109h 16 Bit	P734.07 P734.11 P734.15			
38	3123h	Free object 3123h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	3109h	Free object 3109h 16 Bit	P734.07 P734.11 P734.15			
	310Ah	Free object 310Ah 16 Bit	P734.08 P734.12 P734.16			
39	3123h	Free object 3123h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	3124h	Free object 3124h 32 Bit	P734.07/.08 P734.11/.12 P734.15/.16			
40	310Bh	Free object 310Bh 16 Bit	P734.05 P734.09 P734.13	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
41	310Bh	Free object 310Bh 16 Bit	P734.05 P734.09 P734.13	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	310Ch	Free object 310Ch 16 Bit	P734.06 P734.10 P734.14			
42	310Bh	Free object 310Bh 16 Bit	P734.05 P734.09 P734.13	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	310Ch	Free object 310Ch 16 Bit	P734.06 P734.10 P734.14			
	310Dh	Free object 310Dh 16 Bit	P734.07 P734.11 P734.15			
43	310Bh	Free object 310Bh 16 Bit	P734.05 P734.09 P734.13	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	310Ch	Free object 310Ch 16 Bit	P734.06 P734.10 P734.14			
	310Dh	Free object 310Dh 16 Bit	P734.07 P734.11 P734.15			
	310Eh	Free object 310Eh 16 Bit	P734.08 P734.12 P734.16			

T_PDO No.	Mapping object index	Mapping object name	DPR parameter	T_PDO number in DPR	T_PDO identifier	When selected enter in parameters
44	3125h	Free object 3125h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
45	3125h	Free object 3125h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	310Dh	Free object 310Dh 16 Bit	P734.07 P734.11 P734.15			
46	3125h	Free object 3125h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	310Dh	Free object 310Dh 16 Bit	P734.07 P734.11 P734.15			
	310Eh	Free object 310Eh 16 Bit	P734.08 P734.12 P734.16			
47	3125h	Free object 3125h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
	3126h	Free object 3126h 32 Bit	P734.07/.08 P734.11/.12 P734.15/.16			
48 (2)	6077h 310Fh	Torque actual value	P734.5 P734.9 P734.13	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
49 (2)	6077h 310Fh	Torque actual value Free object 310F 16 Bit	P734.5 P734.9 P734.13 P734.6 P734.10 P734.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
50 (2)	6077h 310Fh 3110h	Torque actual value Free object 310F 16 Bit Free object 3110 16 Bit	P734.5 P734.9 P734.13 P734.6 P734.10 P734.14 P734.7 P734.11 P734.15	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718

T_PDO No.	Mapping object index	Mapping object name	DPR parameter	T_PDO number in DPR	T_PDO identifier	When selected enter in parameters
51 ②	6077h 310Fh 3110h 3111h	Torque actual value Free object 310F 16 Bit Free object 3110 16 Bit Free object 3111 16 Bit	P734.5 P734.9 P734.13 P734.6 P734.10 P734.14 P734.7 P734.11 P734.15 P734.8 P734.12 P734.16	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
52 ②	6077h 3127h	Torque actual value Free object 3127 32 Bit	P734.5 P734.9 P734.13 P734.6/.7 P734.10/.11 P734.14/.15	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
53 ②	6077h 3127h 3111h	Torque actual value Free object 3127 32 Bit Free object 3111 16 Bit	P734.5 P734.9 P734.13 P734.6/.7 P734.10/.11 P734.14/.15 P734.8 P734.12 P734.16	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
54 ②	60FDh	Digital inputs	P734.5/.6 P734.9/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
55 ③	3112h	Free object 3112h 16 Bit	P734.05 P734.09 P734.13	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
56 ③	3112h 3113h	Free object 3112h 16 Bit Free object 3113h 16 Bit	P734.05 P734.09 P734.13 P734.06 P734.10 P734.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
57 ③	3112h 3113h 3114h	Free object 3112h 16 Bit Free object 3113h 16 Bit Free object 3114h 16 Bit	P734.05 P734.09 P734.13 P734.06 P734.10 P734.14 P734.07 P734.11 P734.15	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718

T_PDO No.	Mapping object index	Mapping object name	DPR parameter	T_PDO number in DPR	T_PDO identifier	When selected enter in parameters
58 ③	3112h 3113h 3114h 3115h	Free object 3112h 16 Bit Free object 3113h 16 Bit Free object 3114h 16 Bit Free object 3115h 16 Bit	P734.05 P734.09 P734.13 P734.06 P734.10 P734.14 P734.07 P734.11 P734.15 P734.08 P734.12 P734.16	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
59	3128h	Free object 3128h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
60 ③	3128h 3114h	Free object 3128h 32 Bit Free object 3114h 16 Bit	P734.05/.06 P734.09/.10 P734.13/.14 P734.07 P734.11 P734.15	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
61 ③	3128h 3114h 3115h	Free object 3128h 32 Bit Free object 3114h 16 Bit Free object 3115h 16 Bit	P734.05/.06 P734.09/.10 P734.13/.14 P734.07 P734.11 P734.15 P734.08 P734.12 P734.16	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718
62 ③	3128h 3129h	Free object 3129h 32 Bit Free object 3128h 32 Bit	P734.05/.06 P734.09/.10 P734.13/.14 P734.07/.08 P734.11/.12 P734.15/.16	2 3 4	280h+NodeID 380h+NodeID 480h+NodeID	716 717 718

① PDO can be selected only when device codes 193 and 194 are entered in parameter P719

② PDO can be selected only when device codes 93 and 94 are entered in parameter P719

③ PDO can be selected only when device code 0 is entered in parameter P719

8.5.3 Manufacturer-specific objects

8.5.3.1 Parameter editing

There are two different methods by which parameters in the MASTERDRIVES can be read or written via the CAN bus.

- ◆ One option involves utilizing the complete PKW channel (parameter identifier value) with object 4001h. To do this, however, you must be familiar with the parameter editing principles of the MASTERDRIVES.
- ◆ A simpler parameterization interface is provided to facilitate assignment of the MASTERDRIVES parameters via the CBC with SDO tasks. This interface only allows data to be written to the EEPROM. The number for the parameter to be edited must be identified in the MASTERDRIVES parameter list. 5000h must be added to this parameter number and the product is then the object number which can be read or written. The parameter index is entered in the SDO object index.

An SDO write task must be sent from the master to the MASTERDRIVES specifying the data type, subindex and value. The object number and subindex must be specified in an object read task. The response is sent automatically by the CBC.

Parameter editing with object 4001h

Object 4001_h (Parameter download)

Object with which parameters can be edited via an SDO task. Since most of the parameters in MASTERDRIVES are not converted to CANopen objects, it is possible to gain read or write access via CANopen to all parameters with an SDO task using manufacturer-specific object 4001h.

◆ **Subindex 01**

The parameter number, task identifier and index are entered in subindex 01. These values are then sent by the master with a write task.

The Page Select bit must be set to write parameters higher than 1999. The Page Select bit is bit 15 of the 2nd word.

1 st word:		Parameter identifier (PKE)			
		Byte 1		Byte 0	
Bit No.:	15	12	11	10	0
	TI		SPM	PNU	

2 nd word:		Parameter index (IND)			
		Byte 3		Byte 2	
Bit No.:	15	8	7	0	
	Value = 0x0 or 0x80			Index	

Object 4001_h (Parameter download)

TI: Task identifier

SPM: Toggle bit for parameter change report processing (not supported by CBC)

PNU: Parameter number

◆ **Subindex 02**

The value to be assigned to the parameter with a parameter write task must be entered in subindex 02. This subindex need not be written for a parameter read task. The parameter value is stored in subindex 02 after the parameter read task. The data must be read out of this subindex.

Parameter value (PWE)	
1 st word:	Byte 1 : Byte 0
Parameter value Low (PWE1)	
2 nd word:	Byte 3 : Byte 2
Parameter value High (PWE2)	

When a negative response identifier is returned in subindex 03, the error value of the parameter task is stored in subindex 2.

Parameter value (PWE)	
1 st word:	Byte 1 : Byte 0
Error number	

◆ **Subindex 03**

Subindex 03 contains the response identifier of the parameter task, parameter number and parameter index.

Subindex 03 also supplies the response identifier for all parameter tasks transmitted as SDOs.

1 st word:	Parameter identifier (PKE)			
	Byte 1		Byte 0	
Bit No.:	15	12	11	10
	RI		SPM	PNU
2 nd word:	Parameter index (IND)			
	Byte 3		Byte 2	
Bit No.:	15	8	7	0
	0 or 8		Value = 0	Index

RI: Response identifier

SPM: Toggle bit for parameter change report processing (not supported by CBC)

PNU: Parameter number

Table of task identifiers

Task identifier	Meaning	Response identifier	
		positive	negative
0	No task	0	7 or 8
1	Request parameter value	1 or 2	↑
2	Change parameter value (word) for nonindexed parameters	1	
3	Change parameter value (double word) for nonindexed parameters	2	
6	Change parameter value (array) ¹⁾	4 or 5	
7	Change parameter value (array, word) for indexed parameters ²⁾	4	
8	Change parameter value (array, double word) for indexed parameters ²⁾	5	
10	Reserved	-	
11	Change parameter value (array, double word) and save to EEPROM ²⁾	5	
12	Change parameter value (array, word) and save to EEPROM ²⁾	4	
13	Change parameter value (double word) and save to EEPROM	2	
14	Change parameter value (word) and save to EEPROM	1	↓

1) The desired element of the parameter description is specified in IND (2nd word)

2) The desired element of the indexed parameter is specified in IND (2nd word)

Table of response identifiers

0	No response
1	Transfer parameter value with nonindexed parameters (word)
2	Transfer parameter value with nonindexed parameters (double word)
4	Transfer parameter value (array word) with indexed parameters ¹⁾
5	Transfer parameter value (array double word) with indexed parameters ¹⁾
7	Task cannot be executed (with error number)
8	No control command source status for PKW interface
13	Reserved
14	Reserved

1) The desired element of the parameter description is specified in IND (2nd word)

Table of error values

No.	Meaning	
0	Illegal parameter number (PNU)	If no PNU is connected at all
1	Parameter value cannot be changed	If the parameter is a visualization parameter
2	Lower or upper value limit exceeded	-
3	Errored subindex	-
4	Not an array	In response to tasks for indexed parameters addressed to a nonindexed parameter. E.g. task: "Change parameter value (word, array)" for nonindexed parameter
5	Incorrect data type	-
6	Setting not allowed (resetting only)	-
7	Description element cannot be changed	Task cannot be executed on MASTERDRIVES
11	No control command source status	-
12	Keyword missing	Device parameter: "Access key" and/or "Par. special access" not appropriately set
15	Text array does not exist	-
17	Task cannot be executed in current operating state	Current converter status prohibits execution of transmitted task
101	Parameter number currently deactivated	-
102	Channel not wide enough ¹⁾	Parameter response too long for CAN message
103	Incorrect number of PKWs	Error cannot occur on CBC
104	Illegal parameter value ²⁾	-
105	The parameter is indexed	In response to tasks for nonindexed parameters addressed to an indexed parameter. E.g. task: "Change PWE word" for indexed parameter
106	Task not implemented	-
107	Text cannot be changed	-
108	Number of PWEs <> index number	-

¹⁾ This error number is transferred if the parameter response to a parameter task is longer than the available 8 bytes of CAN data message which means that it cannot be sent. The data are not divided up between data messages.

²⁾ This error number is transferred if no function is assigned in the device to the parameter to be passed or if it cannot currently be accepted for internal reasons (although it is within legal limits).

**Parameter editing
with offset 5000h**

Example:

The value 2600 must be written to parameter U156 (ON posit. cam1), index 1.

1. Object type and subindex from parameter list:
Integer 32, subindices 4
2. Calculation of object number
U156 = 2156dec = 86Chex
86Chex + 5000hex = 586Chex (object number)
3. SDO write task to index 586C, integer32, subindex 1 and value 2600.

8.5.3.2**Example: Change parameter value with object 4001h**

The value 65282 must be written to parameter P711.

**To change a
parameter value**

The drive must be in the Board Configuration (4) or Drive Setting (5) state before a new value can be written to parameter P711.

A value of 4 or 5 must be written to parameter P60 for this purpose.

The value 0x0000203C must then be written to SDO 4001 subindex 1.

The value is calculated as follows:

Convert parameter number 60 to a hex value = 0x03C

Bits 0-11 of the 1st word are written with the parameter number.

Select a task identifier from the list. You must first check the MASTERDRIVES parameter list to find out whether the selected parameter is indexed or a 16-bit or 32-bit parameter. You must then identify the task identifier in the task identifier list.

"2" is the task for a nonindexed parameter that must not be written to the EEPROM and is only 16 bits in size.

Parameter P60 must not be written to the EEPROM since this causes only a status change in the MASTERDRIVES.

The task identifier is entered in bits 12-15 of the 1st word.

This produces the value 0x203C.

Since parameter 60 is not indexed, no index need be entered in word 2 of object 4001 subindex 1. Since the value of parameter P 60 is lower than 1999, it is not necessary to set bit 15 of the second word (Page Select bit).

This produces a double word with the value 0x0000203C which can be written to SDO 4001.01 in this form.

The value to be assigned to parameter P60 must now be entered in object 4001, subindex 2. This is 4 or 5 in this example.

The converter should now be in state 4 (Board Configuration) or 5 (Drive Setting).

The new value can now be written to parameter P711.

The same procedure as described under "Precondition" must be followed now.

Convert P711 to hex = 0x2C7

Find task identifier in list. Check MASTERDRIVES parameter list beforehand to see whether it is an indexed or a 16- or 32-bit parameter.

You must then decide whether the parameter must be written to the EEPROM or the RAM.

Task identifier from list = 12.

Resultant value 0xC2C7

Since parameter P711 is indexed and the value must be written to index 1, the index of the 2nd word must contain a "1". Since parameter P711 is less than 1999, the Page Select bit need not be set.

Resultant value 0x0001C2C7

The value to be written to parameter P711 must then be transferred with object 4001 subindex 2.

To display the converter in the parameter menu again, a "1" must be written to parameter P60 again.

In the case of a parameter with a value higher than 1999, the Page Select bit must be set and subtracted from parameter number 2000. The resultant value must then be converted to a hex quantity and entered in the first word of object 4001.01.

8.5.3.3 Setting factory values (defaults) via CANopen

A "Restore factory settings" operation can be initiated via CANopen on the MASTERDRIVES using the object (4001) parameter. To restore the factory setting (default), parameter P60 (non-indexed, word) must be set to 2, as shown in Subsection 8.5.3.2. "1" must then be written to parameter P366 (non-indexed, word). The restore default operation is finally initiated by overwriting parameter P970 (non-index, word) with 0. The CB parameters are not reset so as to avoid interruption in communication with the CANopen master.

8.5.3.4 Changing the baud rate and bus address (on MASTERDRIVES MC only)

Change the baud rate

It is possible to alter the baud rate and bus address using the CBC and CANopen via manufacturer-specific objects 2100h and 2101h.

The baud rate of the CAN slave can be changed in two different ways:

- ◆ By using parameter P720 (see Subsection 8.5.2)
- ◆ By using object 2100h

Changing the baud rate with object 2100h:

Object 2100 _h (transmission rate)						
Type	UInteger8	Access	rw	PDO mapping	No	
Baud rate of slave on CAN bus You can change the baud rate of the node via the bus using object 2100h. To activate the baud rate, you must first enter the value and then execute a Reset Communication. The relevant node then receives and sends at the new baud rate. Parameter 60 is set to 1 by the CBC after the Reset Communication. Parameter 720.01 contains the new baud rate stored in the EEPROM.						
CAUTION: The value stored in P720.01 is not the same as the value entered via CANopen (see table for assignment).						

Object value	8	7	6	5	4	3	2	1	0
Baud rate [kbit/s]	10	20	50	100	125	250	500	800	1000

Change the bus address

Object 2101 _h (Node number)					
Type	UInteger 32	Access	rw	PDO mapping	No
The node ID (bus address) of the node is set with this object. The new node ID does not become active until the node receives a Reset Communication command from the CANopen master. Parameter 60 is set to 1 by the CBC after the Reset Communication. Parameter 918 then contains the new baud rate stored in the EEPROM.					

8.5.4 Faults and alarms

When a fault occurs on the MASTERDRIVES, the CBC sends an EMERGENCY message on the bus.

These errors are listed in Subsection 8.5.4.2 with CANopen error code, error register, MASTERDRIVES errors and alarms and a brief description of the error.

All objects which are associated in some way with faults or alarms are described below.

8.5.4.1 Structure of object 1003_h (pre-defined error field)

Object 1003 _h (pre defined error field)							
◆ Subindex 0 Number of error entries							
Type	UInteger8	Access	rw	PDO mapping	No		
◆ Subindex 1 Error code The error code of the most recent error is stored in subindex 1. The content of this object (subindex 1) is transferred in the emergency message. The meaning of the error code is specified in the error table in Subsection 8.5.4.2.							
Structure of an emergency message							
Type	UInteger8	Access	rw	PDO mapping	No		
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Error code	Error code	Error register	Alarm number	Error number	Reserve	Error value	Error value
The CANopen error code is transferred in bytes 0 and 1. The error register in byte 2, as with object 1001_h. The MC alarm number assigned to the error code is displayed in byte 3. The MC error number which produces this error code is displayed in byte 4. In bytes 6 and 7, the error value for initialization error F151 or the identifier of the PDO with an insufficient number of databytes (A86) is displayed on MASTERDRIVES MC without F01.							

8.5.4.2 Error codes

Error code 1000 hex Undecoded errors are assigned error code 1000 hex. Read the error number from byte 4.

Error code 1001 hex Undecoded alarms are assigned error code 1001 hex. Read the alarm number from byte 3.

Table of error codes

NOTE

The CB errors and alarms can be found in Subsection 8.5.8.

Error code (hex)	Meaning	MC error	MC alarm	Error text / description
2300	current on device output side	F021		Motor I ² t
2310	continuous over current	F011		Overcurrent
2331	earth leakage phase U	F025		UCE top switch/UCE phase L1
2332	earth leakage phase V	F026		UCE bottom switch/UCE phase L2
2333	earth leakage phase W	F027		Pulse resistor fault/UCE phase L3
2380	I ² t Converter		A025	I ² t converter
2381	I ² t motor		A029	I ² t motor
3210	DC link over-voltage	F006		DC link overvoltage
3220	DC link under-voltage	F008		DC link undervoltage
3230	load error	F002		Fault/precharge
4210	Excess temperature device	F023		Inverter overtemperature
4280	Inverter temperature		A022	Inverter temperature
4310	Excess temperature drive	F020		Motor overtemperature
4380	Motor temperature		A023	Motor temperature
5112	U ₂ =supply +24V	F017		SAFE STOP
5114	SAVE OFF alarm active		A017	SAFE STOP alarm active
5530	EEPROM	F041		EEPROM error
5580	Fault in EEPROM	F255		Error in EEPROM
6100	internal software	F040		Internal error sequence control
6180	Time slot overflow	F042		Time slot overflow
6181	DSP link	F043		DSP coupling
6182	BICO Manager fault	F044		BICO manager error
6183	Time slot overflow		A001	Time slot overflow
6310	loss of parameters	F038		Voltage OFF during parameter save

Error code (hex)	Meaning	MC error	MC alarm	Error text / description
6320	parameter error	F058		Parameter error in parameter task
6380	parameter error	F046		Parameter link error
6381	Parameter fault in follow-up task	F053		Parameter error in following task
6382	Parameter fault after factory setting/init.	F059		Parameter error after factory setting/init.
6383	Incorrect parameterization	F061		Parameterization error
7000	additional modules	F045		HW fault on option boards
7080	TB/CB initialization fault	F080		TB/CB initialization error
7081	OptBoard Heartbeat-counter	F081		Opt. board heartbeat counter
7082	Add. CB Initialization fault	F085		Add. CB initialization error
7121	Motor blocked	F015		Motor blocked
7180	Brake checkback Brake still closed		A036	Brake checkback signal "Brake still closed"
7181	Brake checkback Brake still open		A037	Brake checkback signal "Brake still open"
7182	Motor stall/lock		A042	Mot. stalled/blocked
7183			A075	The values of the leakage or rotor resistance measurement vary widely
7184	Stands. Meas		A078	Standstill measurement
7300	sensor	F051		Encoder fault
7320	Actual Position value not O.K.		A135	Actual position value not o.k.
7380	Encoder board initialisation fault	F054		Encoder board initialization error
7381	Encoder adjustment		A018	Encoder adjustment
7382	Encoder data serial protocol		A019	Encoder data serial protocol
7500	communication	F082		TB/CB message failure
7580	communication warning 1		A083	CB alarm 1 (Subsect. 8.5.8.1)
7581	communication warning 2		A084	CB alarm 2 (Subsect. 8.5.8.1)
8130	communication warning 3		A085	CB alarm 3 (Subsect. 8.5.8.1)
8210	communication warning 4		A086	CB alarm 4 (Subsect. 8.5.8.1)
8480	Overspeed		A033	Overspeed
8481	Setpoint/actual value deviation		A034	Setpoint/actual deviation
8500	Following in error in stand still		A140	Following error at standstill
8611	Following error in motion		A141	Following error in motion
8680	In position - timer monitoring		A142	Position reached - time monitoring

Error code (hex)	Meaning	MC error	MC alarm	Error text / description
FF00	external fault 1	F035		External fault 1
FF01	external fault 2	F036		External fault 2
FF02	PIN is missing	F063		PIN missing
FF03	Fault 1 Function blocks	F148		Fault 1 function blocks
FF04	Fault 2 Function blocks	F149		Fault 2 function blocks
FF05	Fault 3 Function blocks	F150		Fault 3 function blocks
FF06	Fault 4 Function blocks	F151		Fault 4 function blocks
FF07	SIMOLINK start Alarm		A002	SIMOLINK startup alarm
FF08	Drive not synchronous		A003	Drive not synchronous
FF09	Simulation active alarm		A014	Simulation active alarm
FF10	External alarm 1		A015	External alarm 1
FF11	External alarm 2		A016	External alarm 2
FF12	Alarm 1 Function blocks		A061	Alarm 1 function blocks
FF13	Alarm 2 Function blocks		A062	Alarm 2 function blocks
FF14	Alarm 3 Function blocks		A063	Alarm 3 function blocks
FF15	Alarm 4 Function blocks		A064	Alarm 4 function blocks
FF16	Axis does not exist machine data 1=0		A129	Axis does not exist, machine data 1=0
FF17	Operating conditions do not exist		A130	Operating conditions not fulfilled
FF18	OFF1 Missing		A131	OFF1 missing
FF19	OFF2 Missing		A132	OFF2 missing
FF20	OFF3 Missing		A133	OFF3 missing
FF21	Enable Controller ENC Missing		A134	No controller enable ENC
FF22	Machine data 1 changed		A136	Machine data 1 (position encoder type) changed, RESET necessary
FF23	Actual value disable not allowed – axis stand still		A145	Actual value disabling illegal - axis standstill
FF24	Direction of movement not allowed		A146	Direction of movement illegal
FF25	Deceleration=0		A148	Deceleration=0
FF26	Distance to go negative		A149	Distance to go negative
FF27	Setup speed=0		A160	Set up velocity stage=0
FF28	Reference approach velocity		A161	Homing approach velocity=0
FF29	Reference point reducing=0		A162	Homing creep velocity=0
FF30	MDI Block number not allowed		A165	MDI travel block number illegal

Error code (hex)	Meaning	MC error	MC alarm	Error text / description
FF31	No position has programmed in MDI mode		A166	Position MDI does not exist
FF32	No velocity has been programmed in MDI mode		A167	Velocity MDI does not exist
FF33	G91 not allowed with MDI on the fly		A168	On-the-fly MDI with G91 illegal
FF34	Start conditions for flying MDI do not exist		A169	Start condition for on-the-fly MDI does not exist
FF35	Negative overtravel reached		A195	Software limit switch approach in negative direction
FF36	Positive overtravel reached		A196	Software limit switch approach in positive direction

8.5.5 Life guarding / node guarding

MASTERDRIVES with CANopen supports life and node guarding.

After objects 100Ch (guard time) and 100Dh (life time factor) have been set, the MASTERDRIVES is ready to receive the first node guarding message. The monitor in the MASTERDRIVES is activated as soon as the first message has been received.

If the MASTERDRIVES has received a node guarding message, it returns the life guarding message using the same identifier. The MASTERDRIVES toggles the highest bit in every transmitted life guarding message, thereby indicating to the master that it is still alive. In turn, the MASTERDRIVES monitors the CANopen master and waits for the time set in object 100Ch multiplied by the number in object 100Dh.

When this time period has expired, the MASTERDRIVES switches to the state set in parameter P719, see Subsection 8.5.2.1.

8.5.6 The state machine

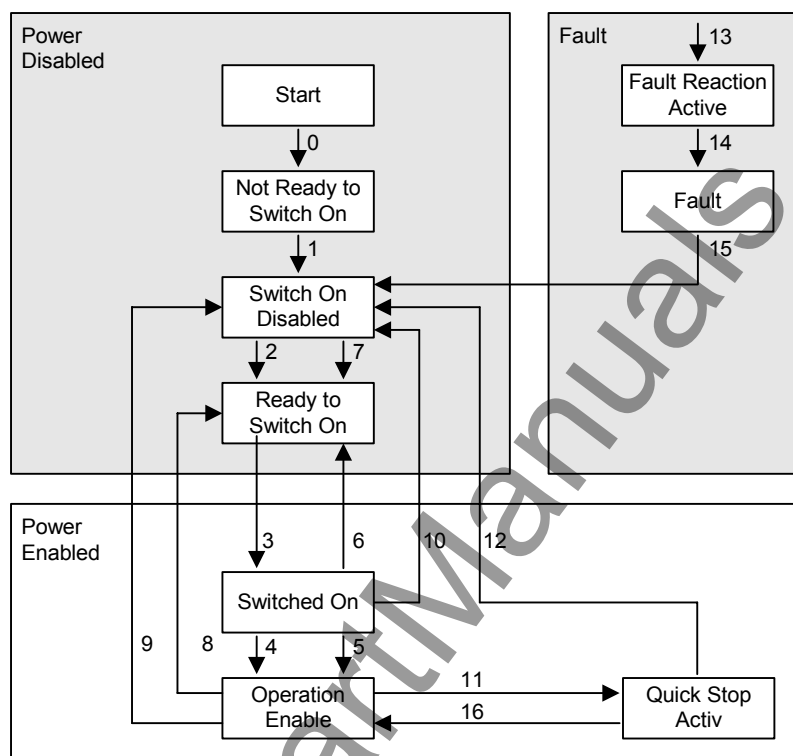


Fig. 8.5-3 The state machine

The state machine is similar to that of the MASTERDRIVES. Please see Chapter 12 "Function Diagrams", diagram 520 for an explanation of how individual states of the MASTERDRIVES can be reached.

Transition 8 is the only transition which is **not** available in MASTERDRIVES.

It has been implemented by skipping the status change from Operation Enable to Switch On Disabled in transition 9.

8.5.6.1 Control word

The bits of the control word must be interconnected in the converter according to the function diagram.

Some of them must be linked to the control word and some to the technology control word or the B-Pos control word of the MASTERDRIVES system.

The control word in the CANopen profile also includes manufacturer-specific bits. These can be freely connected within the MASTERDRIVES. A variety of control functions can be implemented via the MASTERDRIVES control word in this manner (see function diagrams 8513 and 8517).

6040h control word						
Bit	Name					
0	switch on					
1	disable voltage (please note: Implemented with OFF2 only)					
2	quick stop					
3	enable operation					
4	Profile position mode	Profile velocity mode	Synchronous mode	Homing mode	Torque Profile mode	Setup mode
	new setpoint	reserved	synchronous operation start	homing operation start	reserved	reserved
5	change set immediately	reserved	reserved	reserved	reserved	reserved
6	absolute/relative	reserved	reserved	reserved	reserved	[F_S]
7	reset fault					
8	halt (not in synchronous mode, automatic single block mode, automatic position mode)					
9	Reserved					
10	Reserved					
11	J-FWD [D_FWD]	reserved	reserved	reserved	reserved	J-FWD [D_FWD]
12	J-BWD [D_BWD]	reserved	reserved	reserved	reserved	J-BWD [D_BWD]
13	user specific					
14	user specific					
15	user specific					

8.5.6.2 Status word

Some of the bits of status word 1 in the MASTERDRIVES must be connected to a binector/connector converter. Please refer to the function diagram 8512 for MC F01 or 8518 for MC B-Pos in Subsection 8.5.12. These interconnections must always be made since the MASTERDRIVES status word contains bits which do not remain constantly at one or zero.

Every time a bit changes in the status word, the CBC sends, depending on the transmission type, a message with the content of the status word. This can cause evaluation problems and place an unnecessary burden on the bus.

6041h status word						
Bit	Name					
0	ready switch on					
1	switched on					
2	operation enabled					
3	Fault					
4	voltage disabled (please note: Implemented with OFF2 only)					
5	quick stop					
6	switch on disabled					
7	Warning					
8	user specific					
9	Remote					
10	target reached					
11	internal limit active					
12	profile position mode	profile velocity mode	homing mode	synchronous mode	torque profile mode	Setup mode
	setpoint acknowledge	speed=0	homing attained	synchronous mode active	Reserved	reserved
13	following error	reserved	homing error	reserved	Reserved	reserved
14	user specific					
15	user specific					

To provide a better overview of the conversion from MASTERDRIVES status word to CANopen status word, the relevant bits are listed in the table below.

State	Bit 6 Not Ready to Switch On		Bit 5 Quick stop		Bit 4 Voltage Disabled		Bit 3 Fault		BIT 2 Operation Enable		Bit 1 Switched on		Bit 0 Ready to switch on	
Not ready to switch on	0	0	X	X	X	X	0	0	0	0	0	0	0	0
Not ready to switch on	1	1	X	X	X	X	0	0	0	0	0	0	0	1
Ready to switch on	0	0	1	1	0	1	0	0	0	0	0	0	1	1
Switched on	0	0	1	1	0	1	0	0	0	0	1	1	1	0
Operation Enabled	0	0	1	1	0	1	0	0	1	1	1	0	1	0
Fault	0	0	X	X	X	X	1	1	1	0	1	0	1	0
Fault Reaction active	0	0	X	X	X	X	1	1	1	0	1	0	1	0
Quick stop active	0	0	0	0	0	1	0	0	1	1	1	0	1	0

The white fields correspond to the bits in the CANopen specification status word. The gray fields correspond to the bits in the MASTERDRIVES status word.

8.5.6.3 Modes of operation

All available modes are listed in the following table.

The last column in the table specifies the unit which provides the mode.

Object 6060 _h (modes of operation)					
Type	Integer8	Access	wo	PDO mapping	Yes
Modes of operation	Mode			Device which supports mode	
FBh	Automatic single block mode			MASTERDRIVES MC with F01	
FCh	Automatic position mode			MASTERDRIVES MC with F01	
FDh	Setup mode			MASTERDRIVES MC	
FFh	Synchronous mode			MASTERDRIVES MC with F01	
1h	Profile position mode			MASTERDRIVES MC	
3h	Profile velocity mode			MASTERDRIVES MC	
4h	Torque profile mode			MASTERDRIVES MC without F01	
6h	Homing mode			MASTERDRIVES MC	

NOTE

Where specified in the above table, "MASTERDRIVES MC" applies to MASTERDRIVES MC units both with and without F01.

On the general CANopen device, the data for free connection is transferred in the highest bits of connector 3002. The highest bits of parameter P734.02 are transferred with object 6061.

8.5.7 Description of individual modes

8.5.7.1 Profile Position mode

NOTE

The MASTERDRIVES MC with F01 is switched directly to "Profile Position mode" after booting.

The MASTERDRIVES MC without F01 is switched directly to "Profile Velocity mode" after booting.

Positioning operations in "Profile Position mode" are processed differently on MASTERDRIVES MCs with and without F01.

Positioning with MASTERDRIVES MC F01

Positioning on MASTERDRIVES MC F01 is implemented via MDI technology block 0. A number of connections must be made for this purpose. The speed setpoint connector must be wired to U533 and the position setpoint to U532. The required interconnections are contained in scriptfile MC F01.

The CBC automatically resets bit STA P2710.3 in response to a command that causes a status change from Operation Enable. It then waits for bit STA_EN and subsequently executes the command.

For an exact description of the internal interconnections of the CBC, please see Subsection 8.5.11.

Nonflying positioning

1. 6083h Profile_acceleration (MD 18), 6084h profile_deceleration (MD 19) are transferred to the converter.

NOTE

Parameter U502 (transfer of machine data) is set to 2 when object 6083h (MD18) or 6084h (MD19) is written. This is done automatically by the CBC.

Position setpoints are input in LU. The AVWF factor determines how they are normalized (see Subsection 9.4.8 "Position Sensing System for Motor Encoder").

1. The bit change_set_immediately is in control word zero.
2. The traversing velocity 6081h profile_velocity is transferred to the MASTERDRIVES by CANopen (U533).
3. The target position 607Ah target_position is transferred (U532)
4. The command new_setpoint = 1 is transferred to the MASTERDRIVES on a rising edge (STA bit is set).
5. The MASTERDRIVES acknowledges the command with setpoint_acknowledge = 1 (FUR=1) on a rising edge.
6. The MASTERDRIVES resets bit setpoint_acknowledge (FUR=0) when the target position is reached (DRS=1).
7. The CANopen MASTER can now initiate a new positioning operation.

Flying positioning

1. The bit change_set_immediately is in control word one.
2. The traversing velocity 6081h profile_velocity is transferred to the MASTERDRIVES by the CANopen master.
3. The target position 607Ah target_position is transferred.
4. The command new_setpoint is transferred to the MASTERDRIVES (STA bit is set) on a rising edge.
5. The MASTERDRIVES acknowledges the command with setpoint_acknowledge=1 (FUR=1 and Toggle_out= toggle_in).
6. The CANopen master resets bit new_setpoint before positioning has finished (STA bit remains active).
7. The CBC sets setpoint_acknowledge to 0.
8. The new traversing velocity 6081h profile_velocity is transferred to the MASTERDRIVES by the CANopen master.
9. The new target position 607Ah target_position is transferred.
10. The command new_setpoint is sent to the MASTERDRIVES again (TGL bit is toggled).
11. The MASTERDRIVES acknowledges the command with setpoint_acknowledge=1 (toggle_out= toggle_in).
12. This process can now be repeated as often as required until the MASTERDRIVES reaches its target position.

NOTE

If an active positioning operation needs to be terminated prematurely, the CANopen master must send a shutdown or halt command. Bit STA is set to zero at the same time. In the case of shutdown, the OFF1 command is transferred to the MC with checkback signal STA-EN=1 (transition to ready to start (switch on) state).

The MC remains operative with a HALT command.

The interrupted positioning task is resumed when the halt command is canceled.

Positioning with MASTERDRIVES MC

Positioning on MASTERDRIVES MC is processed via free block Basic Positioner. A number of connections must be made for this purpose. The speed setpoint connector must be "wired" to an analog setpoint changeover switch upstream of U868 and the position setpoint to U867.

For an exact description of the connections of the CBC, please see Subsection 8.5.11.

The required interconnections are contained in scriptfile MCEPOS.

Nonflying positioning

1. 6083h Profile_acceleration (U869.01), 6084h profile_deceleration (U869.02) are transferred to the converter.
2. Bit change_set_immediately is zero in the control word. The bit absolute/relative can be 1 or 0.
3. The traversing velocity 6081h profile_velocity is transferred to the MASTERDRIVES by the CANopen master (U868).
4. The target position 607Ah target_position is transferred (U867).
5. The command new_setpoint = 1 is transferred to the MASTERDRIVES. ([SPV_RIE]=1) on a rising edge.
6. The MASTERDRIVES acknowledges the command with setpoint_acknowledge=1 ([SPV_RIE_ACK]=1).
7. The MASTERDRIVES resets bit setpoint_acknowledge ([POS_RUN]=0) when the target position is reached ([POS_OK]=1).
8. The CANopen MASTER can now initiate a new positioning operation again.

Flying positioning

1. Bit change_set_immediately is one in the control word.
2. The traversing velocity 6081h profile_velocity is transferred to the MASTERDRIVES by the CANopen master.
3. The target position 607Ah target_position is transferred.
4. The command new_setpoint is transferred to the MASTERDRIVES on a rising edge ([SPV_RIE]=1).
5. The MASTERDRIVES acknowledges the command with setpoint_acknowledge=1 ([SPV_RIE_ACK]=1).
6. The CANopen master resets bit new_setpoint before positioning has finished ([SPV_RIE] is set to 0).
7. The CBC sets setpoint_acknowledge to 0 ([SPV_RIE_ACK]=0).
8. The new traversing velocity 6081h profile_velocity is transferred to the MASTERDRIVES by the CANopen master.
9. The new target position 607Ah target_position is transferred.
10. The command new_setpoint is sent to the MASTERDRIVES again ([SPV_RIE]=1).
11. The MASTERDRIVES acknowledges the command with setpoint_acknowledge=1 ([SPV_RIE_ACK]=1).
12. This process can now be repeated as often as required until the MASTERDRIVES reaches its target position.

NOTE

If an active positioning operation needs to be terminated prematurely, the CANopen master must send a shutdown or halt command.

In the case of shutdown, the MASTERDRIVES is shut down via OFF 1.

The MC remains operative with a HALT command.

The interrupted positioning task is resumed when the halt command is canceled.

8.5.7.2 Profile Velocity mode

Profile Velocity mode with F01

The "Profile Velocity mode" is selected with [3] by the object 6060h Modes_of_operation (MASTERDRIVES technology mode control [4] or MASTERDRIVES speed control (master drive)).

In order to obtain a velocity as a setpoint or actual value in "Profile Velocity mode", object 6092h (feed constant) must be written by the CANopen master at every boot. This object is stored only on the CBC and deleted on every reinitialization or power OFF.

The value of V_{rated} must be entered in Lu/min (P205) in subindex 1 of 6092h. The entry in subindex 1 determines the transfer format of the setpoints and actual values.

The reference speed value (P353) must be entered in 6092h subindex 2.

Example

The drive is normalized to μm . However, these values can be quite high for a velocity input. If the value from P205 is entered in mm/min in subindex 1, all setpoints can be input in mm/min. Actual values are then also displayed in mm/min.

The reference speed (P353) must be entered in P6092 subindex 2.

The setpoint is specified in LU/min as defined according to object 6092.01h and may deviate from the quantity normalized in the drive by P169/P170. The unit of length LU is specified by the position-feedback scaling factor.

The actual values are returned in object 6092h depending on the normalization.

The setpoint is looped into parameter P212 (source setpoint control) via an analog value switch on the position controller. The analog value switch is connected upstream so that the setpoints are supplied by the technology during referencing (homing). If you decide you want the technology to supply the setpoints all the time, i.e. even in "Profile Velocity mode", you will not need the analog value switch (see Subsection 8.5.12).

In this case, object 60FFh has no influence on the speed. You will then need to modify the velocity values as described in the technology manual.

Checkback signals in Profile Velocity mode:

In "Profile Velocity mode", the status word includes the messages Target Reached and Internal Limit Active. Bit speed = 0 must be generated by a free block interconnection and wired to the location as specified in function diagram 8512.

The bit "target reached" is generated by a free block interconnection with a limit value monitor.

An exact interconnection of free blocks for the message is described in the Appendix.

Profile Velocity mode without F01

The Profile Velocity mode is implemented in the basic unit on MASTERDRIVES without F01. It corresponds to simple speed-controlled operation.

Profile Velocity mode is defined by the following objects

- ◆ Acceleration or deceleration can be set via objects 6083h (profile_acceleration) and 6084h (profile_deceleration).
- ◆ The target velocity for speed-controlled operation is transferred to the MASTERDRIVES via object 60FFh (target_velocity).
- ◆ The transmitted setpoint and actual velocity can be read out via objects 606Bh (velocity_demand_value) and 606Ch (velocity_actual_value).
- ◆ The actual speed in increments per second can be read out via object 6069h. Object 606Ah is a read-only object. The actual speed value is always supplied by the position encoder.

On a MASTERDRIVES MC without F01, objects 2200.01 (P205 rated velocity), 2200.02 (P353.01 reference speed before decimal point) and 2200.03 (P353.02 reference speed after decimal point) are read out as the converter is booting.

As a result, there is no need to write any objects. The CBC has all necessary information to normalize setpoints and actual values correctly.

If, however, a factor of 10 must be introduced because, for example, the speed setpoint is too high, a value raised by a factor of 10 can be written to object 2200.01.

Example

The drive is normalized to μm .

The entry in object 2200.01 defines the transfer format for setpoints and actual values. After the CBC has booted, the value from parameter P205 is stored in this object, i.e. in mm/min. To increase the setpoints even further, it is possible to re-normalize them to e.g. m/min by writing them to object 2200.01. The MASTERDRIVES is not aware that re-normalization has taken place. The value from object 2200.01 is not passed to the MASTERDRIVES.

The setpoint is specified in LU/min as defined according to object 2200.01h and may deviate from the quantity normalized in the drive by P205. The unit of length LU is specified by the position-feedback scaling factor.

The actual values are returned in object 2200.01h depending on the normalization.

The setpoint is wired to parameter P212 (source setpoint control).

Checkback signals in Profile Velocity mode:

In "Profile Velocity mode", the status word includes the messages Target Reached and Internal Limit Active. Bit speed = 0 must be generated by a free block interconnection and wired to the location as specified in function diagram 8518.

The bit "target reached" is generated by a free block interconnection with a limit value monitor.

An exact interconnection of free blocks for the message can be found in function diagram 8518.

Setpoint specification in Profile Velocity mode:

Setpoints are specified in Profile Velocity mode according to the value in object 2200.01 in which, for example, the value $v_{\text{rated}} = 6000$ mm/min is stored.

If the drive is to operate at one third of the rated velocity, the value 2000 mm/min must be specified via object 60FFh.

8.5.7.3 Synchronous mode

Synchronous mode is a manufacturer-specific mode.

The gear ratio can be set with object 2002h.

To activate

Write FF to object 6060h

Start up the MASTERDRIVES MC F01 and then set bit enable_Synchronous mode (STA-Bit). MASTERDRIVES will then switch to synchronous mode.

Bit 12 Synchronous_mode_active is set in the status word as a checkback.

NOTE

For more detailed information about synchronous operation, see Chapter 9 "Technology Option F01".

8.5.7.4 Homing mode

Control and status bits

The axis can be homed (referenced) by various methods in Homing mode.

Various objects listed below are required for this purpose. For detailed information about reference point approach (homing), see Chapter 7 "Functions" or Chapter 9 "Technology Option F01".

The bits of the control and status words are utilized, as specified in profile DSP 402, to operate homing mode.

After homing mode has been selected, the MASTERDRIVES MC must be switched to the "Operation Enable" status. The homing approach is then started by setting control word bit 4 (Homing Operation start).

Homing is stopped by the halt bit or a shutdown. Once the MASTERDRIVES has reached the home position (reference point), it sets bit 12 in the status word.

To start a new homing operation when the axis has already been homed, you must exit Homing mode and then restart it again.

If one of the following errors occurs in the course of homing with a MASTERDRIVES MC with F01, an EMERGENCY message is sent and bit 13 in the status word set:

- ◆ A130
- ◆ A131
- ◆ A132
- ◆ A133
- ◆ A134

The MASTERDRIVES MC without F01 does not generate any alarms.

Bit 11 in the status word is set if the selected homing position is not within the software limit switch area.

Bit 13 is set if the selected homing position is outside the software limit switch area or bit F_REF_BD of B-Pos is set to 1.

Homing mode objects

Object 607Ch: Home Offset

If it becomes necessary to move the located home position (reference point) for mechanical reasons, this can be done with the Home Offset object. The home offset value is entered in machine data 4 (object does not exist for MASTERDRIVES MC without F01).

Object 6098h: Homing_method

The homing method is selected with the Homing Method object.

If a homing method is selected, a number of parameter tasks are processed between the CBC and MASTERDRIVES before the SDO response arrives.

See tables (columns "J-FWD" and "J-BWD"; set with control bit Homing Operation Start depending on the selected mode).

Parameter P178 (Src Rough Pulse) is not assigned through the selection of a homing method, although this may need to be changed in some cases (see MASTERDRIVES Compendium Chapter 7 "Functions" and Chapter 9 "Technology Option F01").

Homing with MC F01 The following table shows the values of parameters that are assigned when a homing method is selected.

Mode	P183	U501.05	U501.08	U502	J-FWD	J-BWD	Reversal cam
1	0x0011	1	0	2	0	1	no
2	0x0021	2	0	2	1	0	no
3	0x0021	2	0	2	1	0	no
5	0x0011	1	0	2	0	1	no
7	0x0021	2	0	2	1	0	yes
10	0x0011	1	0	2	1	0	yes
11	0x0011	1	0	2	0	1	yes
14	0x0021	2	0	2	0	1	yes
17	0x0011	1	1	2	0	1	no
18	0x0021	2	1	2	1	0	no
19	0x0021	2	1	2	1	0	no
21	0x0011	1	1	2	0	1	no
23	0x0021	2	1	2	1	0	yes
26	0x0011	1	1	2	1	0	yes
27	0x0011	1	1	2	0	1	yes
30	0x0021	2	1	2	0	1	yes
32	0x0021	2	2	2	0	1	no
33	0x0011	1	2	2	1	0	no
34	X ¹⁾	3	0	2	1	0	no

1) not evaluated

NOTICE

The values entered in the parameters above as a result of a homing method selection are stored only in the RAM. For this reason, object 6098h must be written prior to **every** homing operation. The 3rd and 4th places of parameter P183 are always set to zero. As a result, the count direction for position sensing cannot be inverted nor can a fractional actual value evaluation factor be used.

NOTE

Please note that parameters MD45 and U536 must be adapted to the relevant hardware. When a homing method is selected, parameters P183, U501.05, U501.08 and U502 are assigned in the order given. J-FWD and J-BWD are set, depending on the homing method, in the control word when bit start_homing is set. Zero is selected as the homing method when the MASTERDRIVES boots. When bit start_homing is set in the control word, J-FWD and-BWD are set simultaneously. This is a way of avoiding the need to write object 6098h prior to every homing operation.

The object homing_method cannot, however, be set to zero. Any attempt to do so generates an abort message.

Homing with MC

The following table shows the values of the parameters assigned when a homing method is selected.

On a MASTERDRIVES MC, all parameters with parameter tasks are written to the EEPROM. If the homing method remains unchanged, object 6098h does not need to be written every time the system boots. The parenthesized parameters are addressed via an OR connection. Refer to the function diagrams for exact interconnection details.

Mode	U274.03 (U866.06) D-FWD	U275.03 (U866.07) D-BWD	U245 (U866.11)	U246 (U866.12)	U875.10	P648.01	U878.01	U878.02	U878.03	P172	P173
17	0	1	18	0	0	4	-	B220	-	-	-
18	1	0	0	18	1	4	-	B220	-	-	-
19	1	0	0	18	1	4	-	B220	-	-	-
21	0	1	18	0	0	4	-	B220	-	-	-
23	1	0	0	12&18	1	4	-	B220	-	-	-
26	1	0	18	12	0	4	-	B220	-	-	-
27	0	1	10 and 18	0	0	4	-	B220	-	-	-
30	0	1	10	18	1	4	-	B220	-	-	-
35	-	-	-	-	-	-	556	B220	627	880	3200

All available methods are described below. Any methods not described are not supported by the MASTERDRIVES.

On a MASTERDRIVES MC without technology board, only the homing methods listed in the table above can be selected.

NOTE

You must always use digital input 5 as the interrupt input on the MASTERDRIVES MC and B-Pos.

For this reason, parameter P647.01 must be set to 0 and parameter P648.01 to 4 on MASTERDRIVES MC without F01. If both inputs are defined as high-speed inputs (P647.01 and P648.01 = 4), the behavior during homing is different.

Reversal cams must be driven via Dig IN 1 and Dig IN 2.

The homing method must be changed before the MASTERDRIVES is switched to Homing mode.

No creep velocity for precise referencing is available for homing with B-Pos on MASTERDRIVES MC without F01 and the deceleration path is not retraced. A suitably low traversing velocity should therefore be selected.

Homing methods using P178 = 16 or 18 and P647.01 = 0 or P648.01 = 0

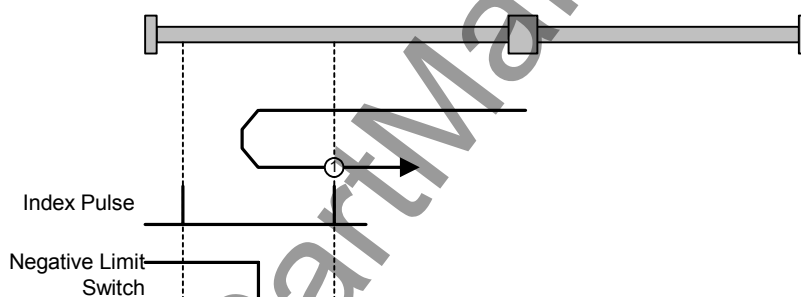
Homing_method 1

Fig. 8.5-4 Homing_method 1

The axis is positioned to the right of the Bero installed as a limit switch. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and reverses its direction of rotation.

When it exits the Bero in the positive direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control.

It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

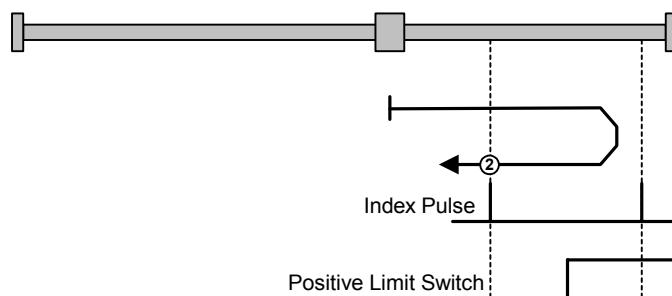
Homing_method 2

Fig. 8.5-5 Homing_method 2

The axis is positioned to the left of the Bero installed as a limit switch. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and reverses its direction of rotation. When it exits the Bero in the negative direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

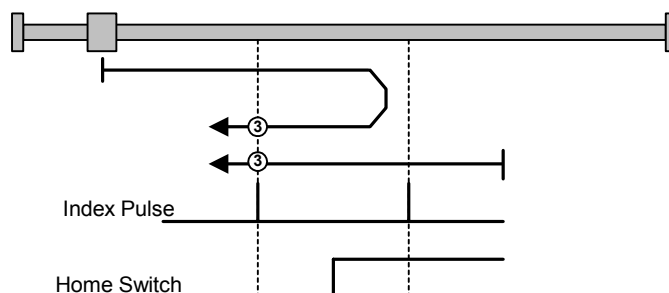
Homing_method 3

Fig. 8.5-6 Homing_method 3

The axis is positioned on the Bero installed as a homing switch. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing creep velocity v_R [MD6] in a negative direction. When it exits the Bero in the negative direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

The axis is positioned to the left of the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and reverses its direction of rotation. When it exits the Bero in the negative direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

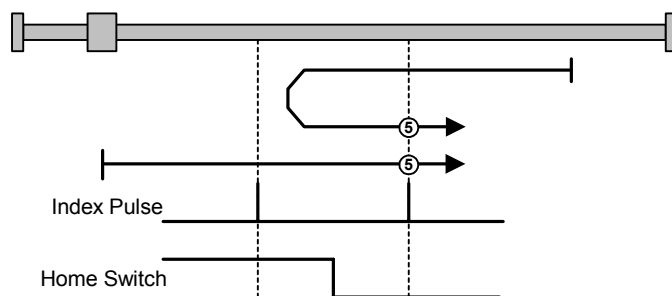
Homing_method 5

Fig. 8.5-7 Homing_method 5

The axis is positioned on the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing creep velocity v_R [MD6] in a positive direction. When it exits the Bero in the positive direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

The axis is positioned to the right of the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and reverses its direction of rotation. When it exits the Bero in the positive direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

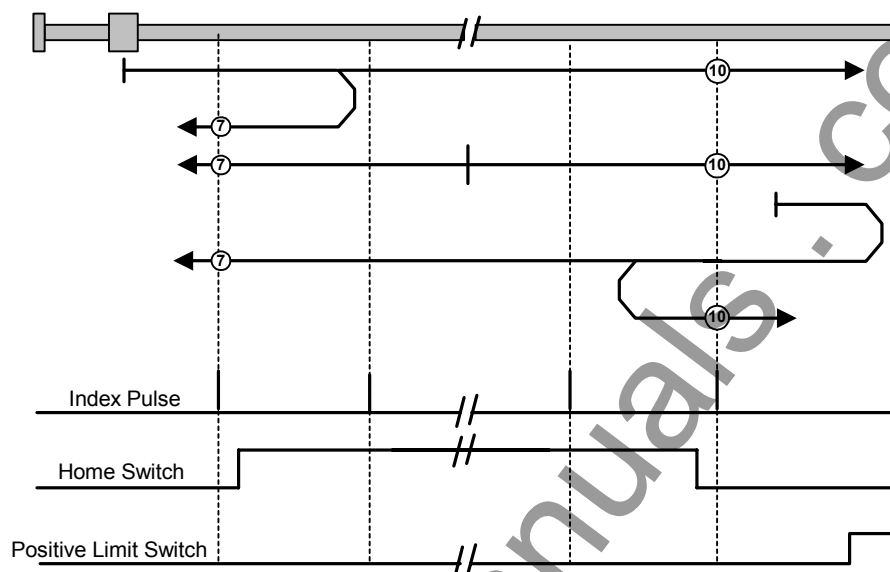
Homing_methods 7 and 10

Fig. 8.5-8 Homing_methods 7 and 10

⑦ The axis is positioned to the left of the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and reverses its direction of rotation. When it exits the Bero in the negative direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

⑦ The axis is positioned on the Bero installed as a homing switch. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing creep velocity v_R [MD6] in a negative direction. When it exits the Bero in the negative direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

⑦ The axis is positioned to the right of the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] in the direction of the positive limit switch (reversal cam). When the cam responds, the axis its direction of rotation and travels towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6]. When it exits the Bero in the negative direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

⑩ The axis is positioned to the left of the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6]. When it exits the Bero in the positive direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

⑩ The axis is positioned on the Bero installed as a homing switch. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing creep velocity v_R [MD6] in a positive direction. When it exits the Bero in the positive direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

⑩ The axis is positioned to the right of the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the positive limit switch (reversal cam). When the cam responds, the axis changes its direction of rotation and travels towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and reverses its direction of rotation. When it exits the Bero in the positive direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Homing_methods 11 and 14

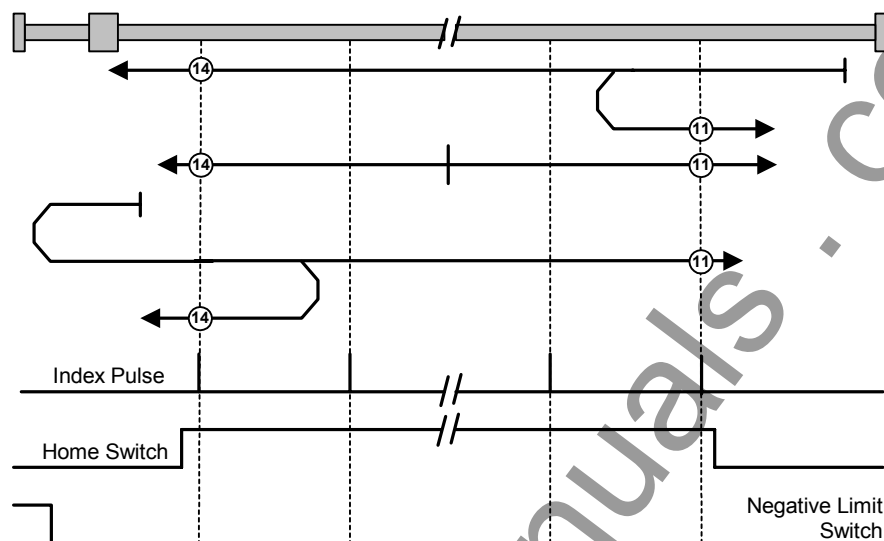


Fig. 8.5-9 Homing_methods 11 and 14

⑪ The axis is positioned to the right of the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and reverses its direction of rotation. When it exits the Bero in the positive direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

⑪ The axis is positioned on the Bero installed as a homing switch. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing creep velocity v_R [MD6] in a positive direction. When it exits the Bero in the positive direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

⑪ The axis is positioned to the left of the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the negative limit switch (reversal cam). When the cam responds, the axis changes its direction of rotation and travels towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6]. When it exits the Bero in the positive direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under

speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

⑭ The axis is positioned to the right of the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6]. When it exits the Bero in the negative direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

⑭ The axis is positioned on the Bero installed as a homing switch. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing creep velocity v_R [MD6] in a negative direction. When it exits the Bero in the negative direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

⑭ The axis is positioned to the left of the Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the negative limit switch (reversal cam). When the cam responds, the axis changes its direction of rotation and travels towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and reverses its direction of rotation. When it exits the Bero in the negative direction, it searches for the next zero pulse of the position encoder. When the zero pulse is found, the axis is braked to a standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

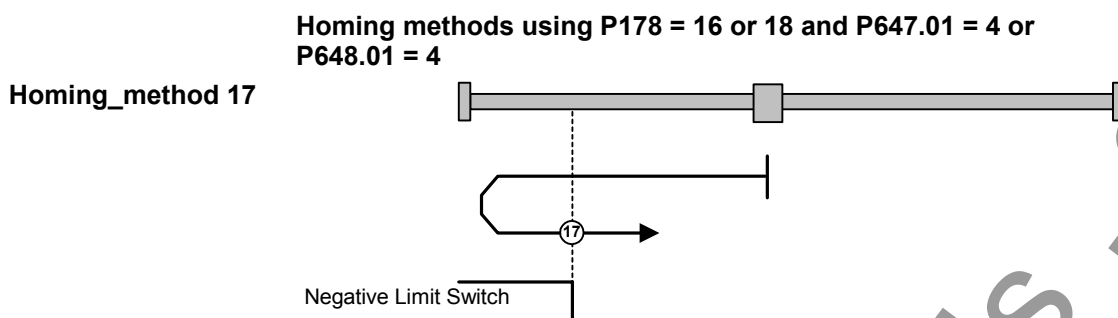


Fig. 8.5-10 Homing_method 17

⑰ Axis is positioned to the right of the BERO installed as a negative limit switch. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and changes its direction of rotation. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES MC without F01

The axis traverses under position control towards the Bero at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the Bero responds, the axis changes its direction of rotation and travels at the same speed in the opposite direction. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

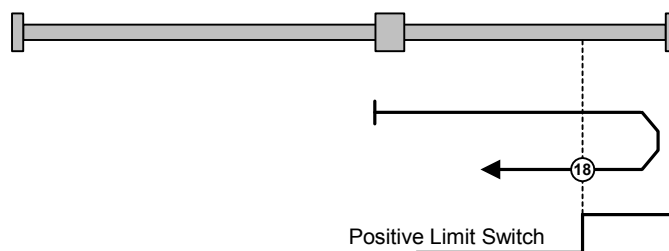
Homing_method 18

Fig. 8.5-11 Homing_method 18

⑱ Axis is positioned to the left of the BERO installed as a positive limit switch. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and changes its direction of rotation. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the Bero at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the Bero responds, the axis changes its direction of rotation and travels at the same speed in the opposite direction. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

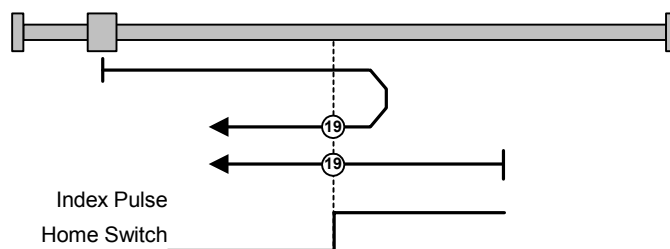
Homing_method 19

Fig. 8.5-12 Homing_method 19

①⁹ Axis is positioned to the left of the BERO installed as a homing switch. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and changes its direction of rotation. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the Bero at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the Bero responds, the axis changes its direction of rotation and travels at the same speed in the opposite direction. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

①⁹ The axis is positioned on the Bero installed as a homing switch. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing creep velocity v_R [MD6] in a negative direction. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the Bero at the values stored in U006 (homing acceleration) and U873.01 (homing speed) in a negative direction. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

Homing_method 21

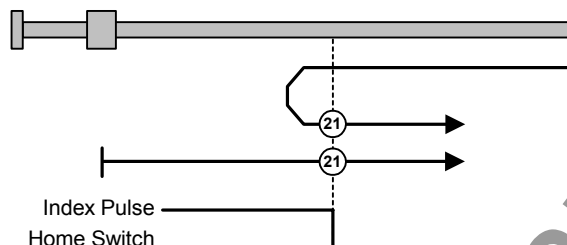


Fig. 8.5-13 Homing_method 21

②① Axis is positioned to the right of the BERO installed as a homing switch. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and changes its direction of rotation. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the Bero at the values stored in U006 (homing acceleration) and U873.01 (homing speed) towards the Bero. When the Bero responds, the axis changes its direction of rotation and travels at the same speed in the opposite direction. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

②① Axis is positioned on the BERO installed as a homing switch. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES with F01

The axis traverses under speed control at homing creep velocity v_R [MD6] in a positive direction. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the Bero at the values stored in U006 (homing acceleration) and U873.01 (homing speed) in a positive direction. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

Homing_methods 23 and 26

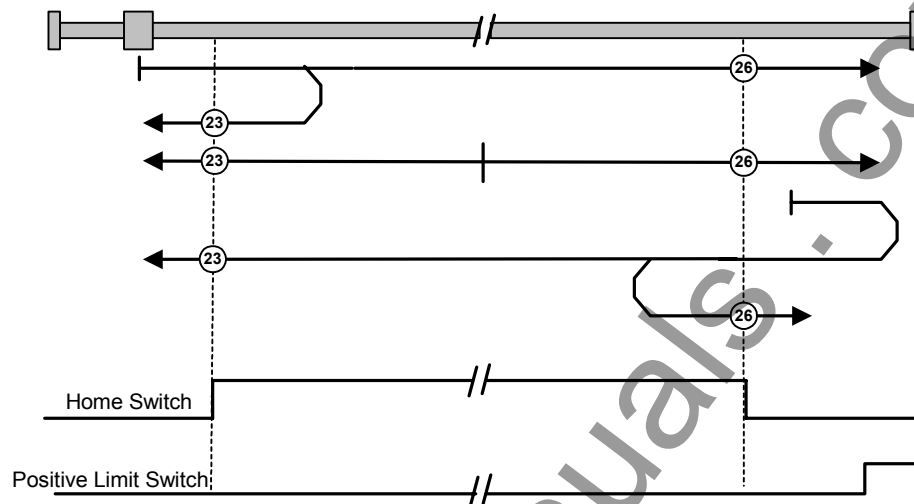


Fig. 8.5-14 Homing_methods 23 and 26

②③ Axis is positioned to the left of the Bero. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and changes its direction of rotation. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the Bero at the values stored in U006 (homing acceleration) and U873.01 (homing speed) towards the Bero. When the Bero responds, the axis changes its direction of rotation and travels at the same speed in the opposite direction. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

②③ Axis is positioned on the Bero installed as a homing switch. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing creep velocity v_R [MD6] in a negative direction. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis moves under position control in a negative direction at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

②③ Axis is positioned to the right of the Bero. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the positive limit switch (reversal cam). When the cam responds, the axis changes its direction of rotation and travels towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6]. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis moves under position control towards the positive limit switch (reversal cam) at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the reversal cam responds, the axis changes its direction of rotation and moves towards the Bero.

When the Bero responds, the axis maintains its velocity. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

②⑥ Axis is positioned to the left of the Bero. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6]. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the Bero at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the Bero responds, the axis continues traversing at the same velocity. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

②⑥ Axis is positioned on the Bero installed as a homing switch. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing creep velocity v_R [MD6] in a positive direction. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis moves under position control in a positive direction at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

②⑥ Axis is positioned to the right of the Bero. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the positive limit switch (reversal cam). When the cam responds, the axis changes its direction of rotation and travels towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and changes its direction of rotation. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the positive limit switch (reversal cam) at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the reversal cam responds, the axis changes its direction of rotation and moves towards the Bero.

When the Bero responds, the axis changes its direction of rotation and continues moving at the same velocity in the opposite direction. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

Homing_methods 27 and 30

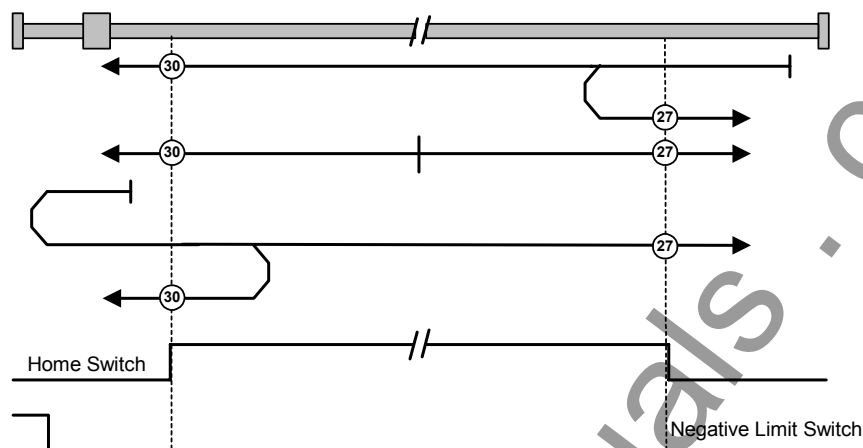


Fig. 8.5-15 Homing_methods 27 and 30

②⑦ Axis is positioned to the right of the Bero. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and changes its direction of rotation. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis moves under position control towards the Bero at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the Bero responds, the axis changes its direction of rotation and continues moving at the same velocity in the opposite direction. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

②⑦ Axis is positioned on the Bero installed as a homing switch. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing creep velocity v_R [MD6] in a positive direction. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis moves under position control in a positive direction at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

②⑦ Axis is positioned to the left of the Bero. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the negative limit switch (reversal cam). When the cam responds, the axis changes its direction of rotation and travels towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6]. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the negative limit switch (reversal cam) at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the reversal cam responds, the axis changes its direction of rotation and moves towards the Bero.

When the Bero responds, the axis maintains its velocity. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

③⑩ Axis is positioned to the right of the Bero. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6]. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the Bero at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the Bero responds, the axis continues moving at the same velocity. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006. It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

③⑩ Axis is positioned on the Bero installed as a homing switch. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing creep velocity v_R [MD6] in a negative direction. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis moves under position control in a negative direction at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006 (homing deceleration). It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

③⑩ Axis is positioned to the left of the Bero. The homing operation is started by bit 4 in the control word.

Behavior of MASTERDRIVES MC with F01

The axis traverses under speed control at homing approach velocity v_A [MD7] towards the negative limit switch (reversal cam). When the cam responds, the axis changes its direction of rotation and travels towards the Bero. When the Bero responds, the axis decelerates down to homing creep velocity v_R [MD6] and changes its direction of rotation. When the Bero falling edge appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Behavior of MASTERDRIVES without F01

The axis traverses under position control towards the negative limit switch (reversal cam) at the values stored in U006 (homing acceleration) and U873.01 (homing speed). When the reversal cam responds, the axis changes its direction of rotation and moves towards the Bero.

When the Bero responds, the axis changes its direction of rotation and continues moving at the same velocity in the opposite direction. When the Bero falling edge appears, the axis is decelerated to a standstill with the value from U006 (homing deceleration). It does not retrace the braking path traveled during the braking operation. It activates "Homing Attained" in the status word via bit ARFD.

Homing_methods 33 and 34

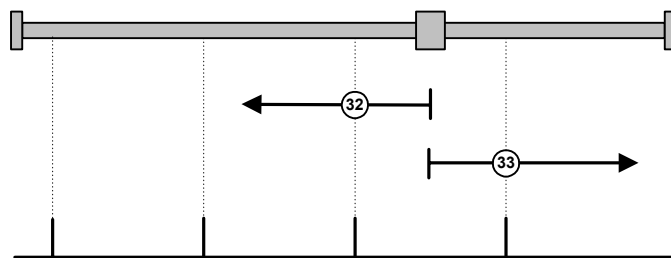


Fig. 8.5-16 Homing_methods 33 and 34

③ Axis is homed without a Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing creep velocity v_R [MD6] in a negative direction. When the zero pulse appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

④ Axis is homed without a Bero. The homing operation is started by bit 4 in the control word.

The axis traverses under speed control at homing creep velocity v_R [MD6] in a positive direction. When the zero pulse appears, the axis is decelerated down to standstill under speed control. It retraces the deceleration path traveled as a result of the braking operation by executing a position-controlled return motion (positioning) towards the zero pulse. The axis then activates "Homing Attained" in the status word via bit ARFD.

Homing methods using P178 = 0 and P647.01 = 1 or P648.01 = 1**Homing_method 35 Behavior of MASTERDRIVES MC with F01**

With Homing_method 35, the coordinate is set as soon as homing is activated by the user program. In this case, the value from MD 3 is set as the actual position value.

After Homing_method 35 has been selected, a value other than 0 can be entered in MD 3 (home position coordinate) by means of object 4001h (Parameter Download, see also Subsection 8.5.3).

After a value has been entered in MD 3, parameter U502 can be set to 2 by the user, again using object 4001h.

Home position setting with object 607Ch=0

When control bit 4 is set, the position setpoint and actual position value are set to the "Home position coordinate" [MD3] and the Homing Attained bit set in the status word.

Home position setting with object 607Ch ><0

When control bit 4 is set, the axis accelerates to "Homing creep velocity" (object 6099.02h) in a positive or negative direction (depending on the sign of object 607Ch) under position control and traverses the home position offset. The position setpoint and actual value are then set to the "Home position coordinate" [MD3] and status bit Homing Attained is set.

Behavior of MASTERDRIVES without F01

With Homing_method 35, the coordinate is set as soon as homing is activated by the user program. In this case, the value from U874.02 which is connected to U877.03, is set as the actual position value. After Homing_method 35 has been selected, a value other than 0 can be entered in U874.02 using a Parameter Download object.

**Object 6099h
Homing speeds**

Object Homing Speed 6099h is used to set the velocity in subindex 1 at which the drive traverses while it is searching for the homing switch. The input is in 1000LU/min.

The velocity at which the drive traverses as it is searching for the zero pulse from the encoder is specified in subindex 2. The input is the same as for subindex 1, i.e. 1000LU/min. Subindex 2 is not supported on MC without F01.

For further details about objects, please refer to Profile DSP 402 or Chapters 7 "Functions" and 9 "Technology Option F01".

8.5.7.5 Profile Torque mode

This mode is supported only on the MASTERDRIVES MC without F01. Profile Torque allows an axis to operate under torque control. To avoid involving the ramp-function generator in the basic unit, the setpoint is taken via the simple ramp-function generator 2 [FP786b]. To enable the simple RFG (unit LU) to interact with the basic unit, basic unit normalization 'per cent' has been selected as the LU, i.e. 100 % = 4000hex = 16384 dec.

If you want to set the ramp-function generator limits (torque limits) to ± 100 %, parameter U472.01, U472.02 must be set to ± 16384 . Accordingly, 50 % equals 8192 (= 2000h), etc.

Profile Torque mode can be set using the following objects:

The torque setpoint is specified via 6071h (target_torque). The acceleration ramp is defined by object 6087h (torque_slope). target_torque is specified per mille.

=> $U008 = 6071h \text{ (Target\_Torque)} / 1000 \times 4000h$
(1000 per mille = 100 % = 4000h = 16384)

The same applies to 6087h (torque_slope). The unit of [6087h] is per mille / sec according to the CAN profile. The unit of adjusting velocity of the basic unit is LU / sec.

But LU is % in this case.

=> $U471.01 = 6087h / 1000 \times 4000h$

(The CAN object 6088h (Torque Profile Type) belonging to Torque mode can only be read out in MASTERDRIVES).

8.5.7.6 Setup mode

Setup mode is supported by MASTERDRIVES MC with and without F01. Setup mode is activated with object 6060h (modes of operation) mode FDh. A position-controlled traversing motion is activated (Jog) with bits 11 and 12 in the control word. Bit 11 acts on bit J-FWD on MCs with technology F01 and on bit D_FWD with B-Pos. Bit 12 acts on J-BWD on MCs with technology F01 and on D_BWD with B-Pos. The velocity is supplied by parameter U510.01 or U510.02 [FP819] on MCs with technology F01. Bit 6 in the control word toggles between high and low velocity [F_S]. The B-Pos receives its setpoints via the connectors linked to the B-Pos block 'Setpoint transfer' [FP789a].

Setup mode can be set by means of the following objects:

Object 607Dh defines the position of the positive and negative software limit switches in LU.

On MCs without F01, object 6081h specifies the travel velocity in Setup mode and objects 6083h and 6084h sets the acceleration or deceleration.

8.5.7.7 Automatic Position mode

Automatic Position mode is supported only on MASTERDRIVES MC with F01. It is equivalent to technology mode [5].

The technology bits are driven via the technology control word 4040h [FP 809]. The basic unit is controlled via control word 6040h.

The technology checkback signals are passed via object 4041h [FP 811] and those of the basic unit via object 6041h.

For more detailed information about this mode, please refer to the MASTERDRIVES Motion Control compendium and the Motion Control for MASTERDRIVES MC manual.

8.5.7.8 Automatic Single Block mode

Automatic Single Block mode is supported only on MASTERDRIVES MC with F01. It is equivalent to technology mode [6].

The technology bits are driven via technology control word 4040h [FD 809]. The basic unit is controlled via control word 6040h.

The technology checkbacks only run via object 4041h [FD811], the checkbacks of the basic unit via object 6041h.

For more detailed information about this mode, please refer to the MASTERDRIVES Motion Control compendium and the Motion Control for MASTERDRIVES MC manual.

8.5.8 Diagnostics and troubleshooting

8.5.8.1 Error and alarm displays on basic unit

In the case of communication errors between the CAN bus and the CBC, appropriate errors and/or alarms are displayed on the PMU or OP1S of the basic unit.

CB alarms

Alarm	Meaning	Error-code (hex)	Meaning
A083	CB alarm Possible cause: Errored CAN messages are being received or transmitted and the internal error counter has exceeded the alarm limit. The errored CAN messages are ignored. The data last transmitted remain valid. If the errored CAN messages are process data, the message failure monitoring function (P722) may respond with error F082 (DPR message failure). If the PKW CAN messages are errored, the converter does not respond. Remedial measure: • Check parameter P720 (baud rate) for every node and correct if necessary. • Check the cable link between nodes. • Check the cable shield. The bus cable must be shielded at both ends. • Reduce electromagnetic interference. • Replace the CBC board.	7580	communication warning 1

Alarm	Meaning	Error-code (hex)	Meaning
A084	CB alarm Possible cause: Errored CAN messages are being received or transmitted and the internal error counter has exceeded the fault limit. The errored CAN messages are ignored. The data last transmitted remain valid. If the errored CAN messages are process data, the message failure monitoring function (P722) may respond with error F082 (DPR message failure). If the PKW CAN messages are errored, the converter does not respond. Remedial measure: • Check parameter P720 (baud rate) for every node and correct if necessary. • Check the CAN bus master. • Check the cable connection between bus nodes. • Check the cable shield. The bus cable must be shielded at both ends. • Reduce electromagnetic interference. • Replace the CBC board.	7581	communication warning 2
A085	CB alarm A "Life guarding event" has occurred. The converter has changed state according to the setting in P719. Cause: No node guarding messages have been received from the master.	8130	Life Time error
A086	CB alarm The PDO received is shorter than the parameterized PDO. Alarm A086 has been activated. The identifier to which the alarm applies is displayed in bytes 6 and 7 (fault value). The alarm will be canceled again when the next PDO is positively received.	8210	PDO not processed due to length error

CB error display

Error	Meaning
F 080	TB/CB Initialization fault: CBC is incorrectly initialized and parameterized via the dual port RAM interface (DPR interface) • Error in CBC parameter(s), cause of parameterization error in diagnostic parameter r732.01 → Correct CB parameter P711-P721 Correct CB bus address P918 • CBC defective → Replace CBC Note: If the MASTERDRIVES boots with an error F80, the parameter channel is disabled. MASTERDRIVES cannot then process any SDO tasks.
F 081	OptBrt heartbeat counter: The CBC has ceased processing the heartbeat counter. • CBC incorrectly inserted in the electronics box → Check CBC • CBC defective → Replace CBC
F 082	TB/CB message failure: The message failure monitoring time set in parameter P722 has run out. NOTE: The message failure monitoring time should be set to 0 for CANopen since data are not always exchanged cyclically and message failure is monitored by the node guarding function. • CAN bus master has failed (green LED on CBC is off; with cyclical data exchange only) • Cable connection between bus nodes interrupted (green LED on CBC is off) → Check bus cable • Electromagnetic interference on bus cable too high → Observe EMC guidelines • Message monitoring time is set too short (green LED on CBC flashing) → Increase parameter value in P722 • CBC defective → Replace CBC

Error	Meaning
F 151	Fault 4 Function blocks A normalization parameter could not be read out as the CBC was booting. Note: This error is activated only if it is connected to bit 1 of the 2nd word of the CB receive data. Bits 0-4 for each of the failed parameter tasks are set in parameter r732.19. ◆ Bit 0: P205 could not be read → Read out object 2200.01h with SDO task ◆ Bit 1: P353.01 could not be read → Read out object 2200.02h with SDO task ◆ Bit 2: P353.02 could not be read → Read out object 2200.03h with SDO task ◆ Bit 3: U857 could not be read → Read out object 2200.04h with SDO task ◆ Bit 4: U007 could not be read → Read out object 6098h with SDO task

8.5.8.2 Evaluation of CBC diagnostic parameter

The CBC stores diagnostic data in a diagnostic buffer to support commissioning and servicing activities. The diagnostic information can be read out with indexed parameter r732.i (CB/TB diagnosis). This parameter is displayed in hexadecimal notation.

The CBC diagnostics buffer is assigned as follows:

	Meaning
r 732.01	Error identifier configuration (1)
r 732.02	Not used
r 732.03	Not used
r 732.04	Not used
r 732.05	Not used
r 732.06	Not used
r 732.07	Not used
r 732.08	Not used
r 732.09	Not used
r 732.10	Not used
r 732.11	Not used
r 732.12	Not used
r 732.13	Not used
r 732.14	Counter for properly processed PKW tasks (incl. response)
r 732.15	Counter for errors in processing PKW tasks (incl. response)
r 732.16	Error type of error in processing PKW tasks
r 732.17	Error value of error in processing PKW tasks
r 732.18	Not used
r 732.19	Parameter which cannot be read out during initialization. (only MC)
r 732.20	Not used
r 732.21	Not used
r 732.22	Not used
r 732.23	Not used
r 732.24	Not used
r 732.25	Not used
r 732.26	Software version
r 732.27	Software identifier
r 732.28	Software date 'Day/Month'
r 732.29	Software date 'Year'

8.5.8.3 Meaning of CBC diagnostics

r732.01 Error identifier configuration

If the CB parameters contain an invalid value or an invalid combination of parameter values, the converter switches to fault status with error F80 and fault value 5 (r949). You can read out the cause of the fault via index 01 of CB diagnostic parameter r732.

Value (hex)	Meaning
0x0	No error
0x1	Incorrect bus address
0xC	Error in config status
0x17	Invalid baud rate
0x23	Incorrect CAN protocol type (0: Layer 2, 1: CanOpen)
0x101	Invalid mapping of 1 st RxPDO
0x102	Invalid transmission type of 1 st RxPDO
0x201	Invalid mapping of 2 nd RxPDO
0x202	Invalid transmission type of 2 nd RxPDO
0x301	Invalid mapping of 3 rd RxPDO
0x302	Invalid transmission type of 3 rd RxPDO
0x401	Invalid mapping of 4 th RxPDO
0x402	Invalid transmission type of 4 th RxPDO
0x111	Invalid mapping of 1 st TxPDO
0x112	Invalid transmission type of 1 st TxPDO
0x211	Invalid mapping of 2 nd TxPDO
0x212	Invalid transmission type of 2 nd TxPDO
0x311	Invalid mapping of 3 rd TxPDO
0x312	Invalid transmission type of 3 rd TxPDO
0x411	Invalid mapping of 4 th TxPDO
0x412	Invalid transmission type of 4 th TxPDO
0x444	Invalid device (not MCF01 or MPF01)

r732.14 Counter for correctly processed PKW tasks (incl. response) since power ON.

CAUTION

This is a counter for PKW tasks. It does not count correctly processed SDO tasks! Between 0 and 4 PKW tasks are executed depending on the SDO task (see table of objects in Subsection 8.5.1 and homing methods table, Subsection 8.5.7.4).

r732.15

Counter for errors in processing PKW tasks (incl. response) since power ON.

CAUTION

The errors that occur in processing SDO tasks are not counted!

r732.16

Error type with PKW task "processing error".

An error identifier is entered in this parameter if an error occurs in the processing of PKW tasks.

Value (hex)	Meaning
0x0	No error
0x4	DPR error: errored status byte
0x5	DPR error: errored control byte, programming error in task/response channel or an attempt has been made using object 4001 to start a parameter task which has not been implemented
0xA	Programming error in parameter status
0xB	The basic unit has not processed the transmitted parameter task within the timeout of 150 ms (or 300 ms).
PKW-PKE/IND	With response identifier 7: Task cannot be executed or with response identifier 8: No PKW control command source status

r732.17

Error value with PKW task "processing error".

Contains additional information about a particular error type.

With error types 0x4 and 0x5:

Value (hex)	Meaning
0x0	No error
0x1	Error in task channel
0x2	Error in response channel
0x66	Channel not wide enough
0x6A	Task not implemented

With error types 0xA and 0xB:

Value (hex)	Meaning
PKW-PKE/IND	Task identifier and parameter index of transmitted parameter task

Error value for a PKW response with response identifier 7 or 8:

See Subsection 8.5.3.1, Parameter Download.

8.5.9 CANopen EDS

CAN-EDS (Electronic Data Sheet) passes the IDs of the objects that are available in the device to a commissioning tool. These objects are set to their defaults.

Description

You need an EDS to be able to work with a CANopen network configuring tool.

Using this EDS, you can inform the Commissioning Tool network which objects and functions are made available by the connected CANopen device.

Application

The CANopen EDS for the MASTERDRIVES with technology option F01 or B-Pos contains all the available objects of the device.

Not all EDS objects are always available in the device. Those which are available are dependent on the PDO mapping via parameters P711 to P718. Consequently, a DCF file needs to be generated from the EDS. This DCF therefore contains the bus address setting, the baud rate and the mapped objects.

To find out which objects are mapped, read the conventions for PDO mapping in Subsection 8.5.2.4. Configure your DCF accordingly.

Example:

You have selected PDO 29 from the table of receive PDOs in parameter 712.

This contains the following objects

60FFh	target_velocity
3003h	Free object 3003h / 16-bit
3004h	Free object 3004h / 16-bit

Object 60FFh is always available in the device provided that the correct interconnection is made in the MASTERDRIVES device. This object has no effect on the EDS.

Objects 3003h and 3004h are not available if they have not been mapped via parameters P712 - P714 in a premapped PDO to the device.

This means that these objects need to be transferred to the DCF.

The three thousand objects that are not mapped to a PDO must not be transferred to the DCF.

8.5.10 Parameterization

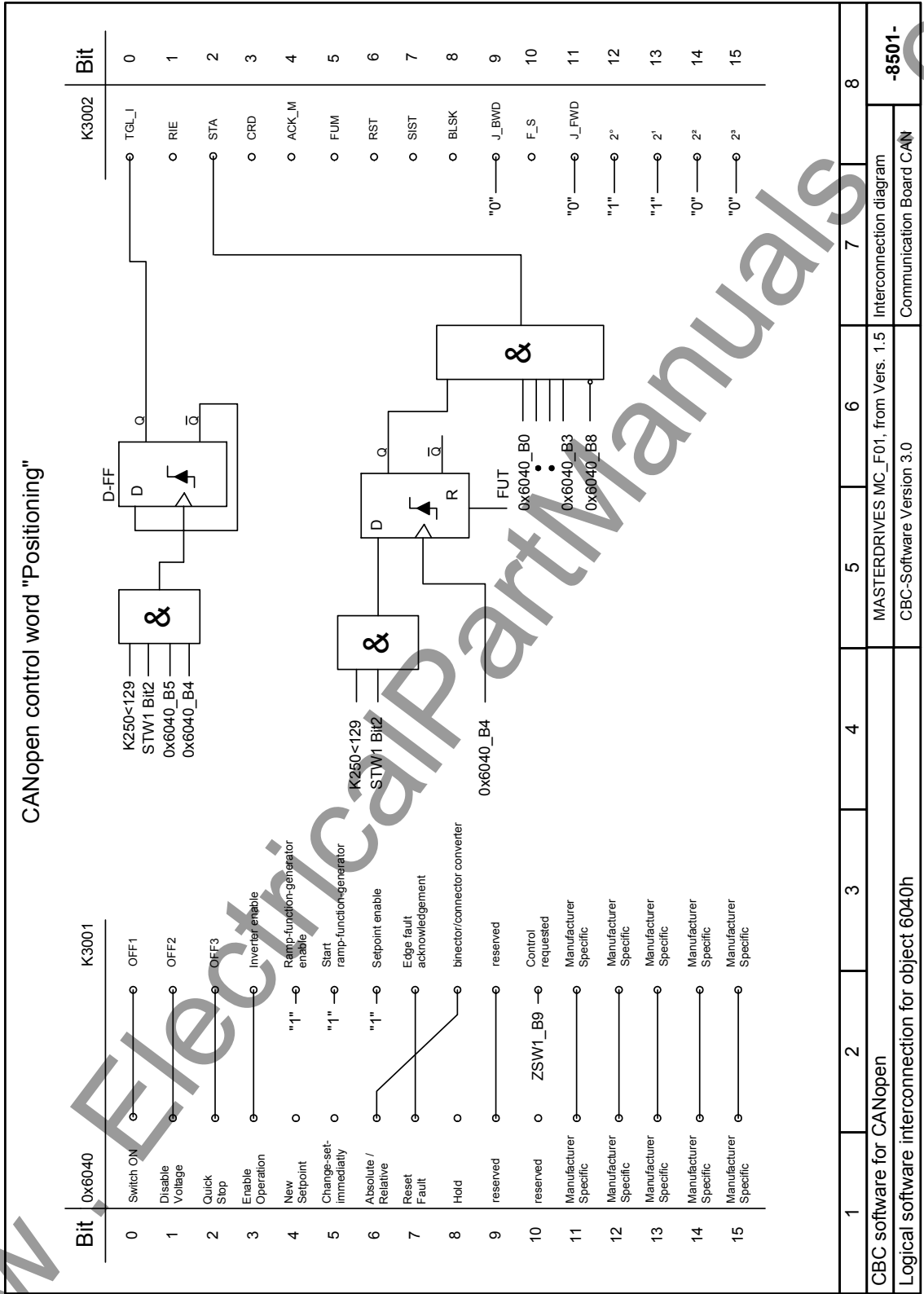
8.5.10.1 **Parameterization for the CBC CANopen with MASTERDRIVES MC_F01 and MASTERDRIVES MC_B-Pos**

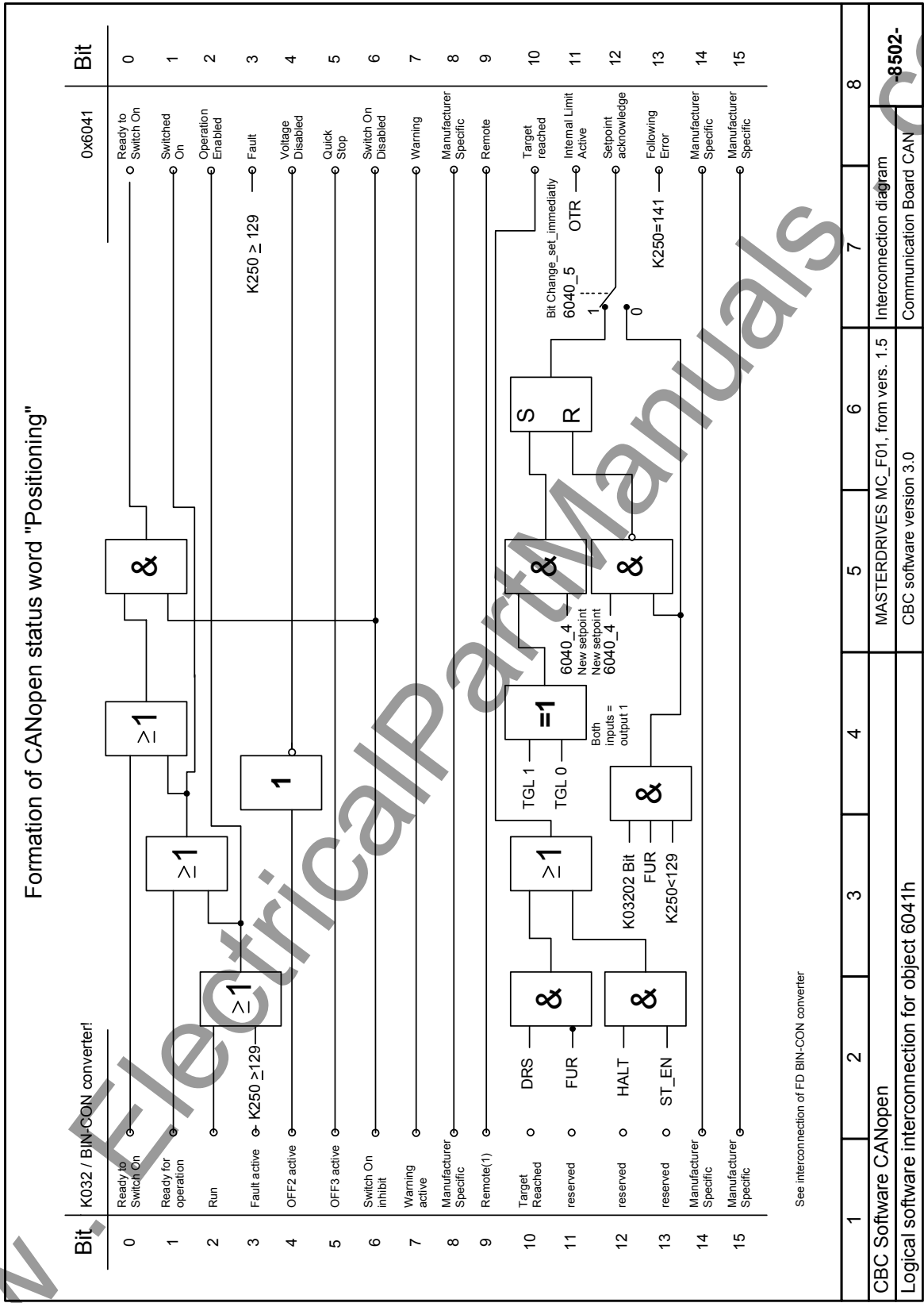
The CD "Drive Monitor" contains scriptfiles for the parameterization of MASTERDRIVES MC_F01 and MASTERDRIVES MC_B-Pos.

You must adapt these scriptfiles to suit your application.

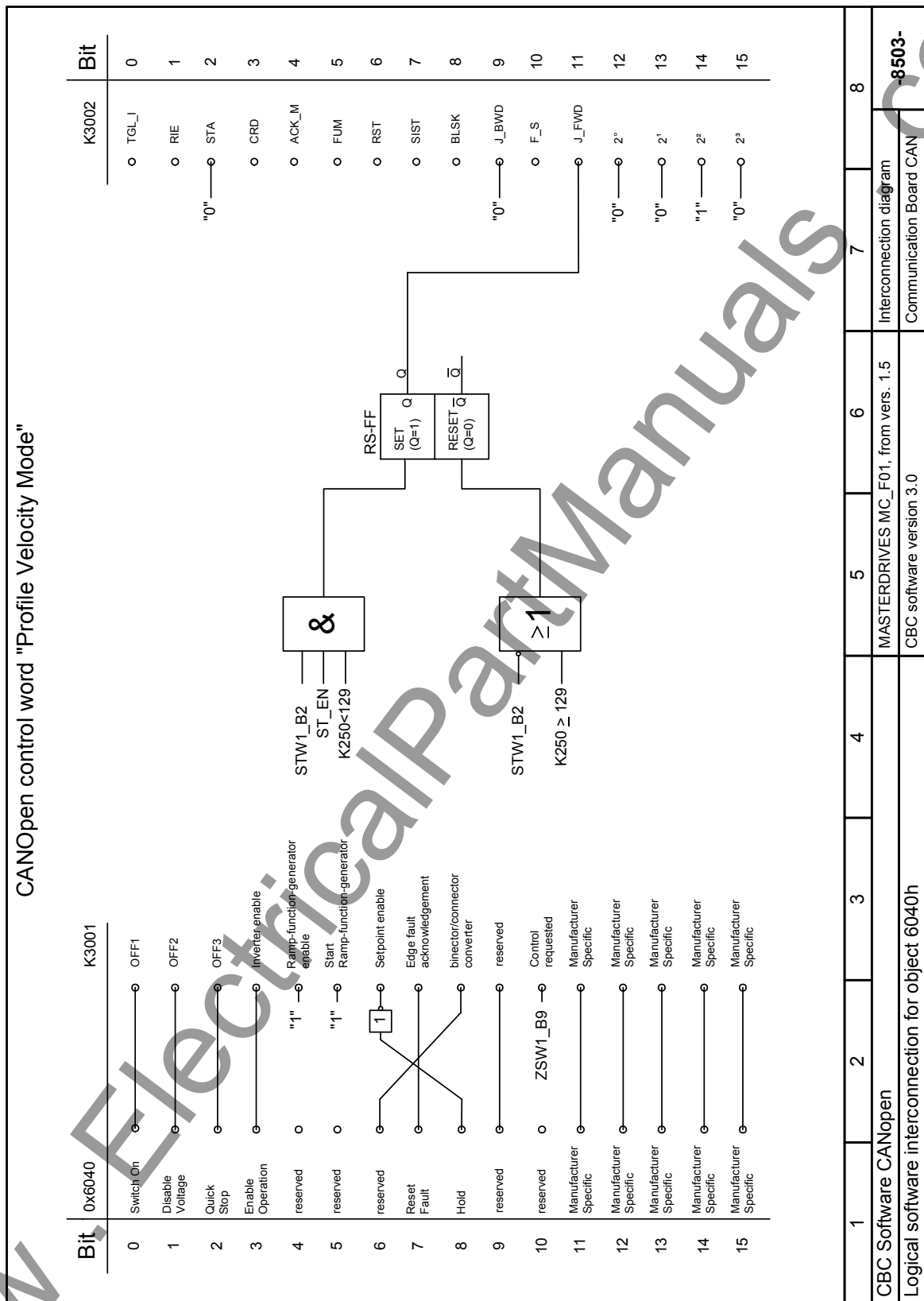
The CB parameters, receive connectors, send parameters and connectors must be "wired up" to suit your application.

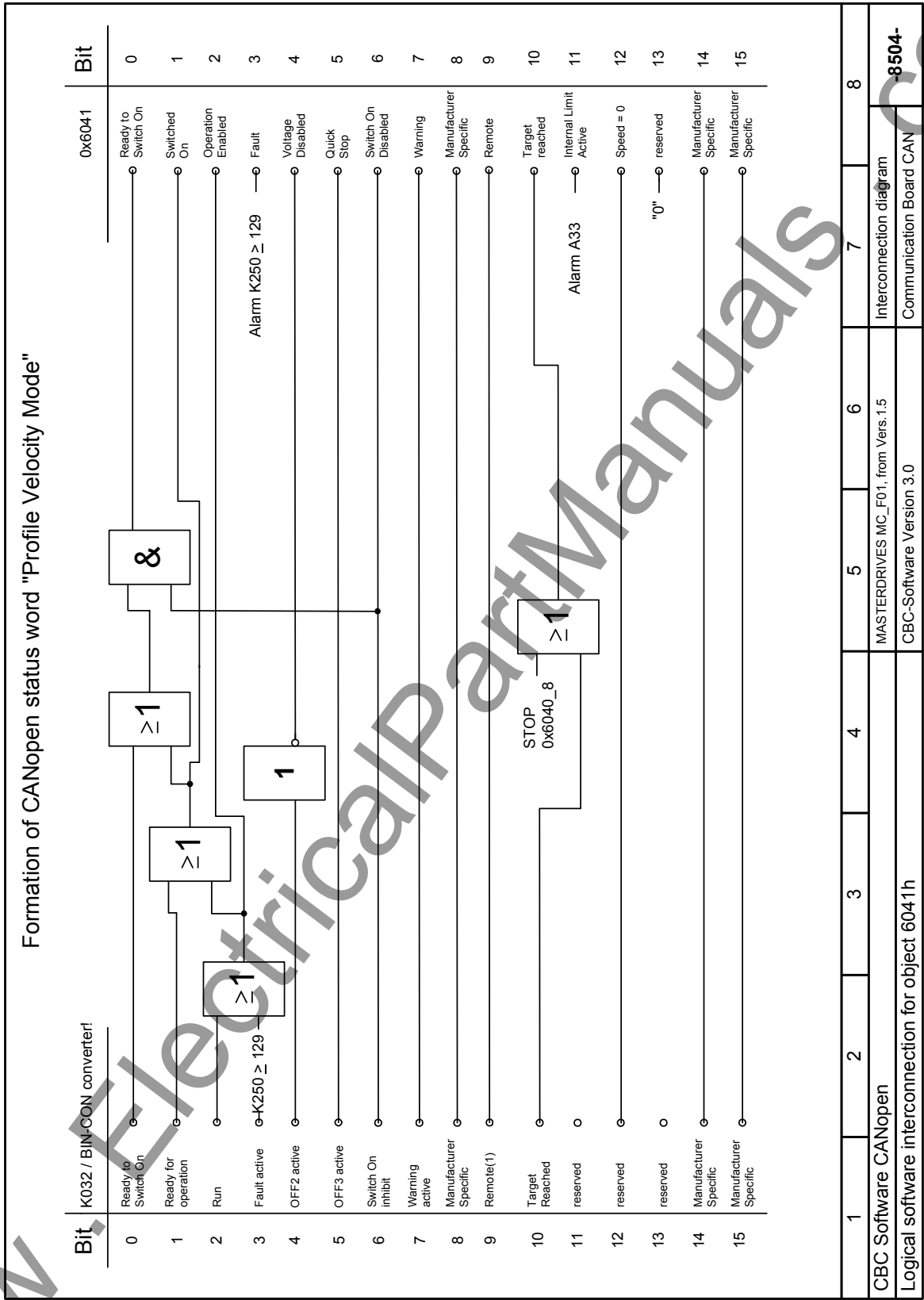
8.5.11 Logical interconnections for control and status words



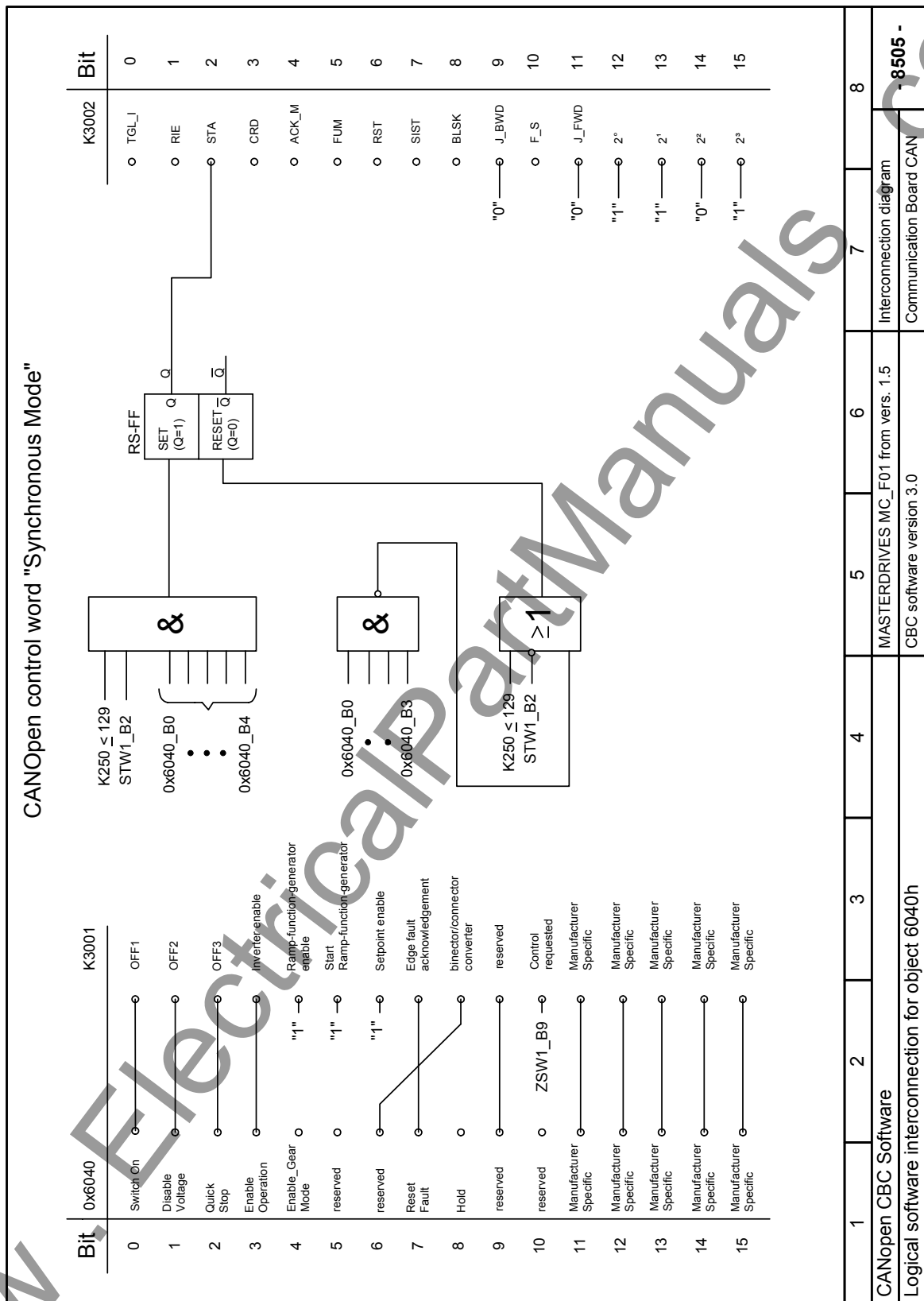


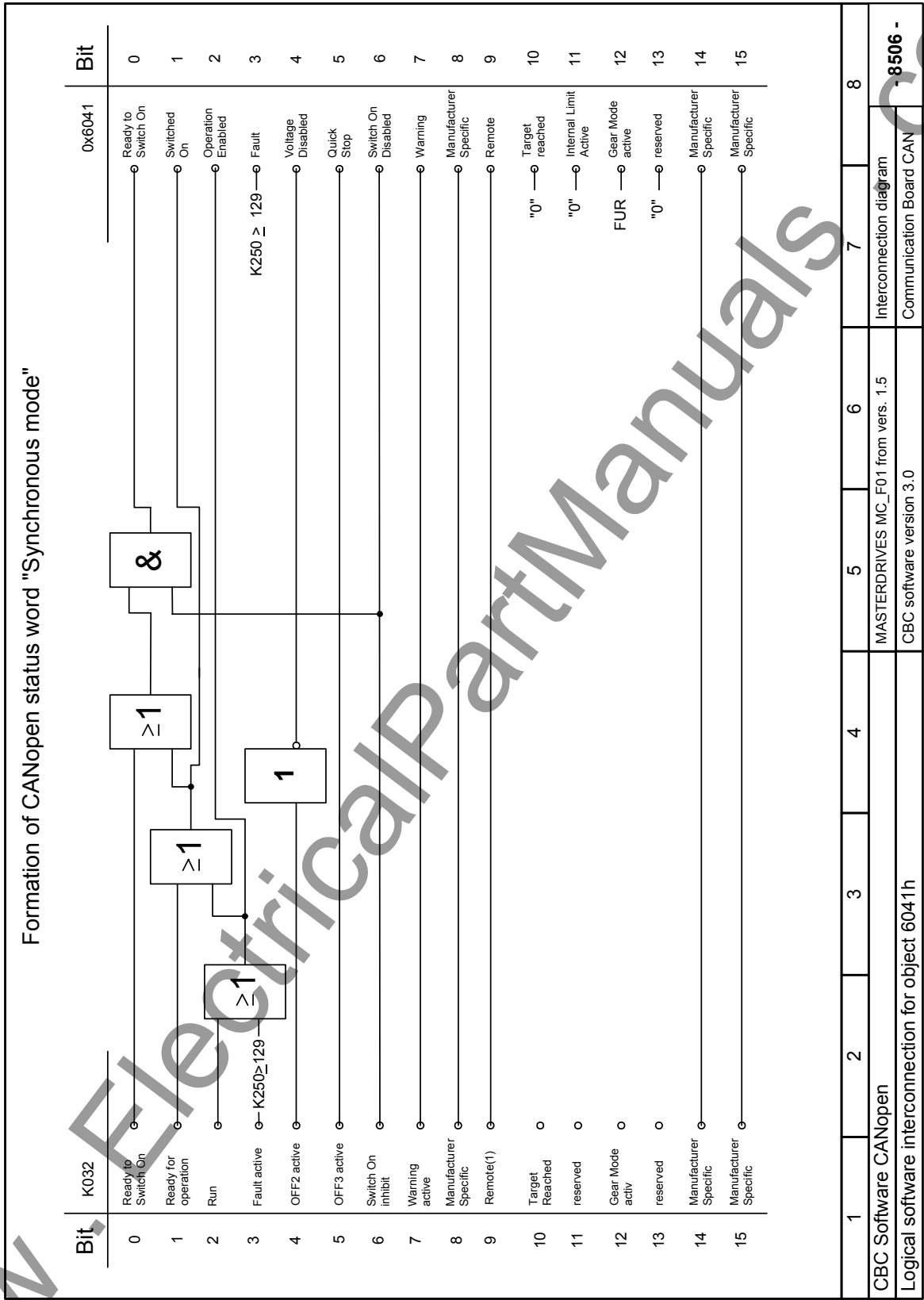
CANopen control word "Profile Velocity Mode"

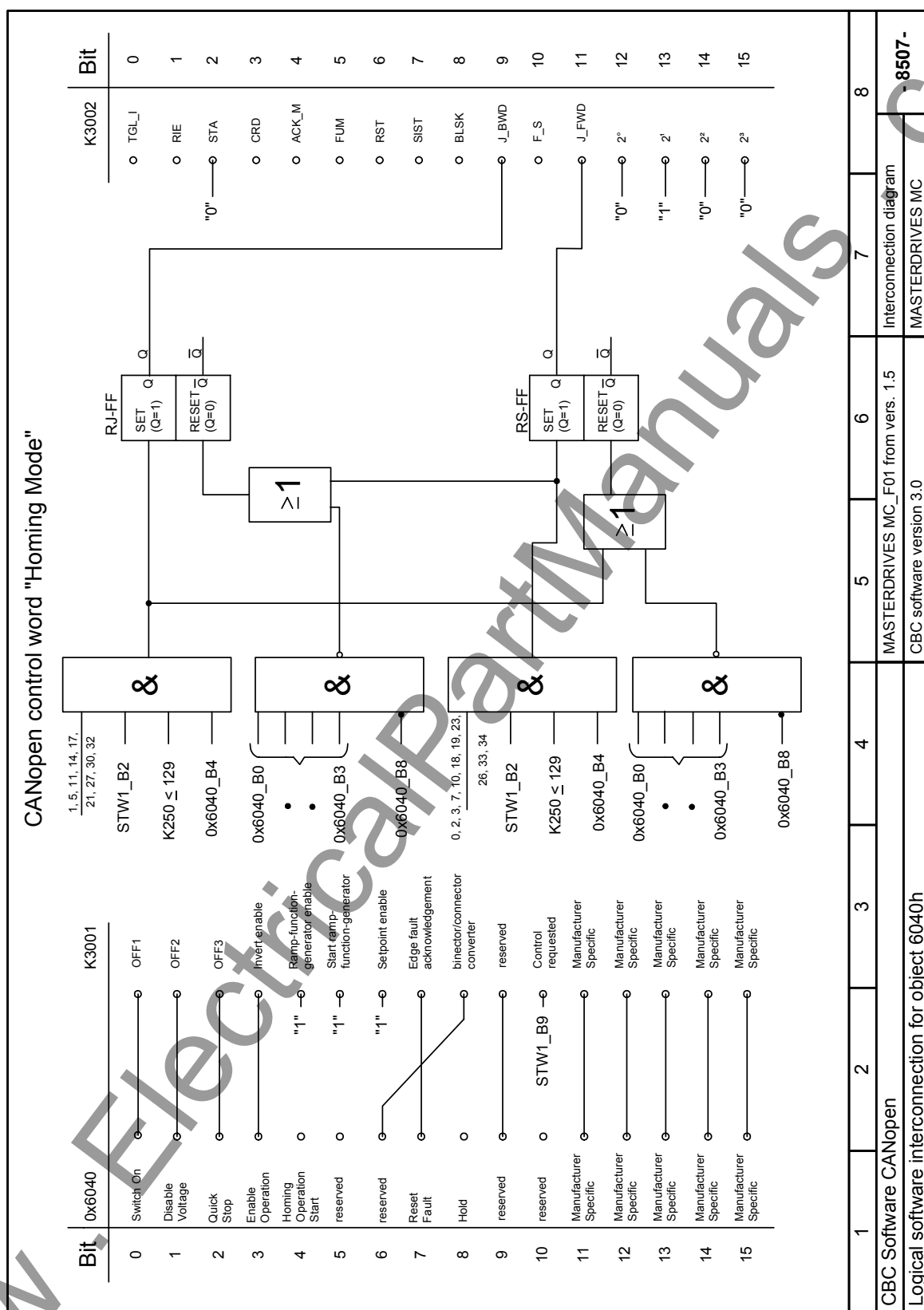


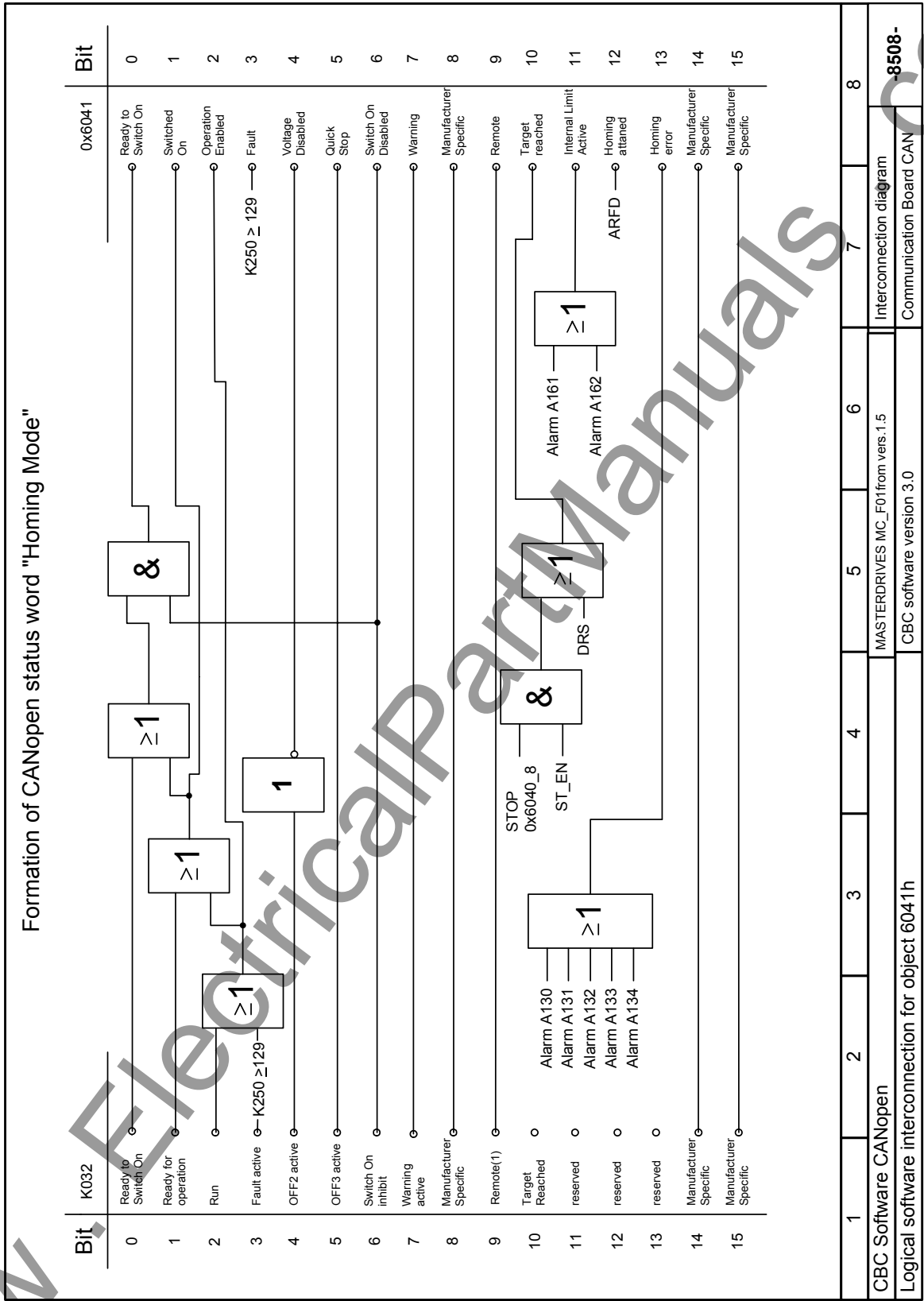


CANOpen control word "Synchronous Mode"

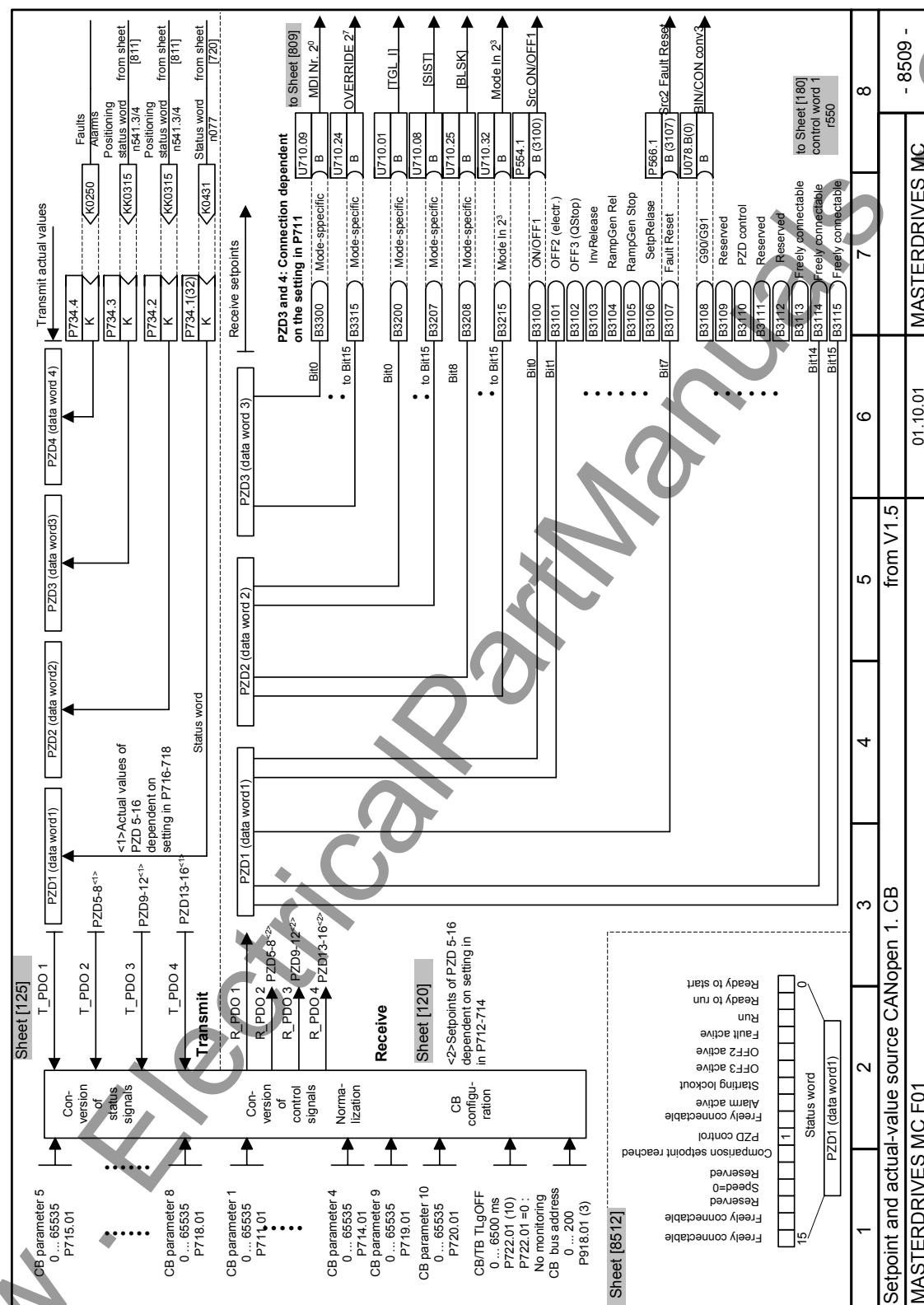


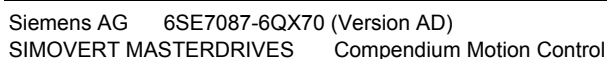


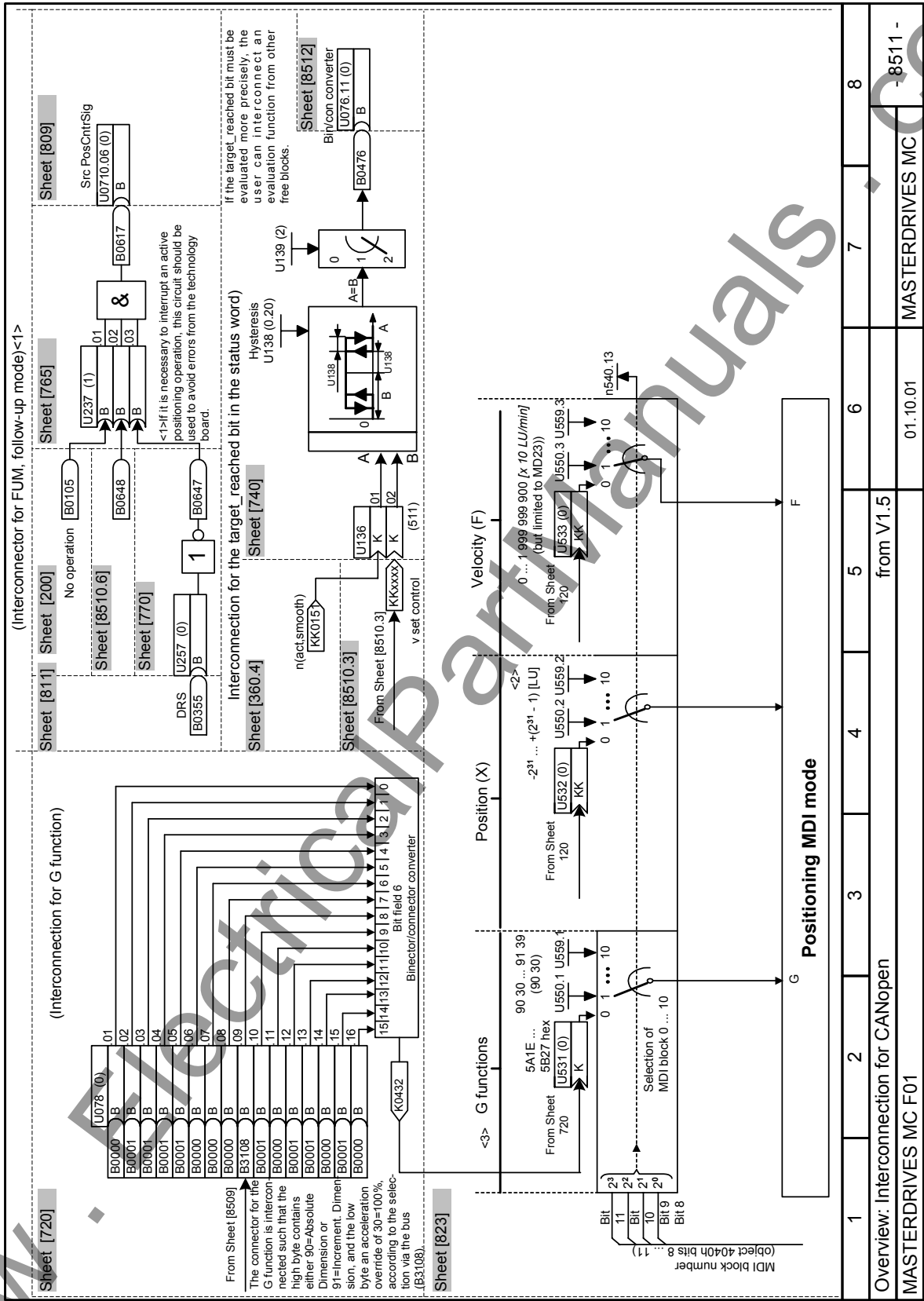


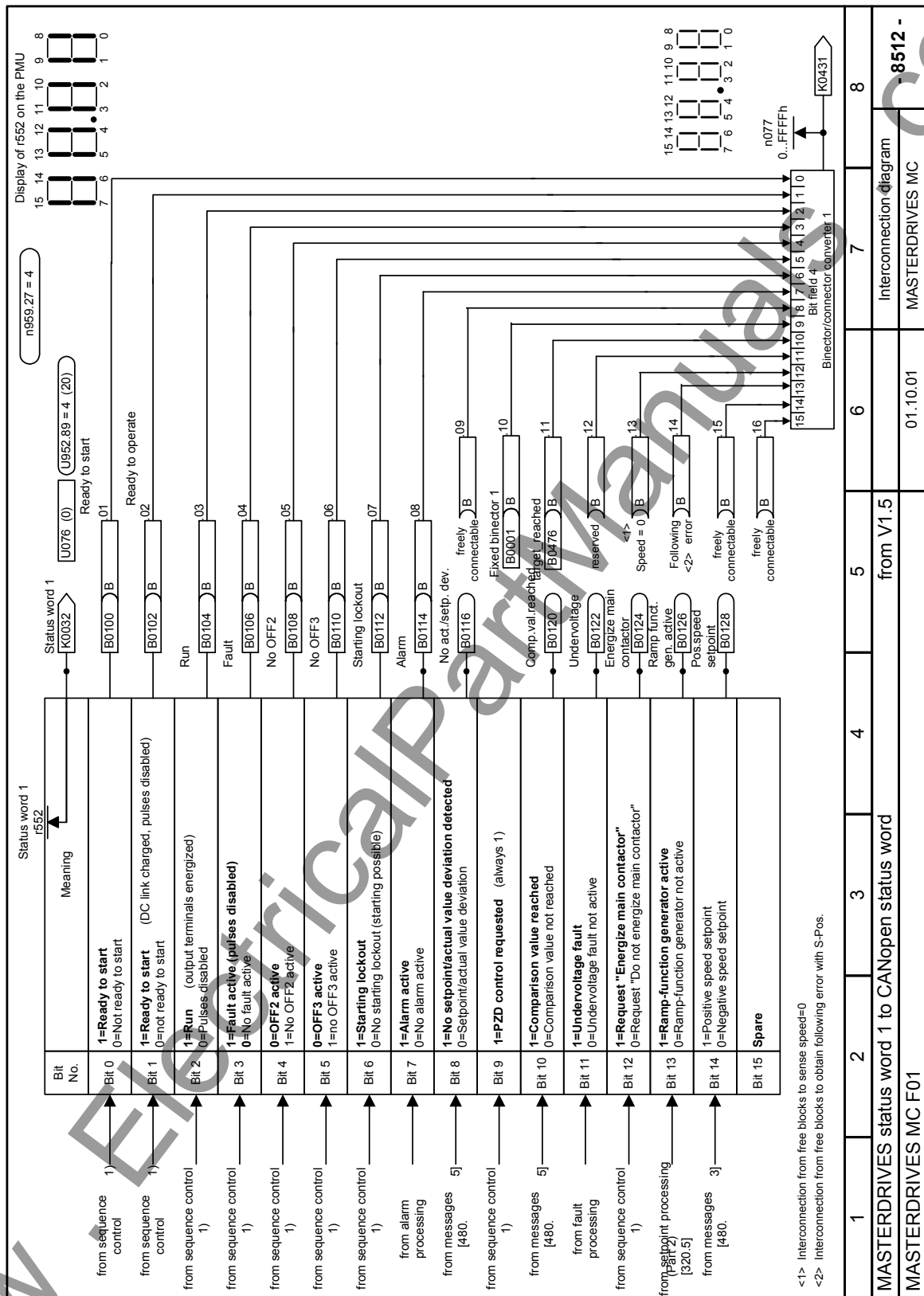


8.5.12

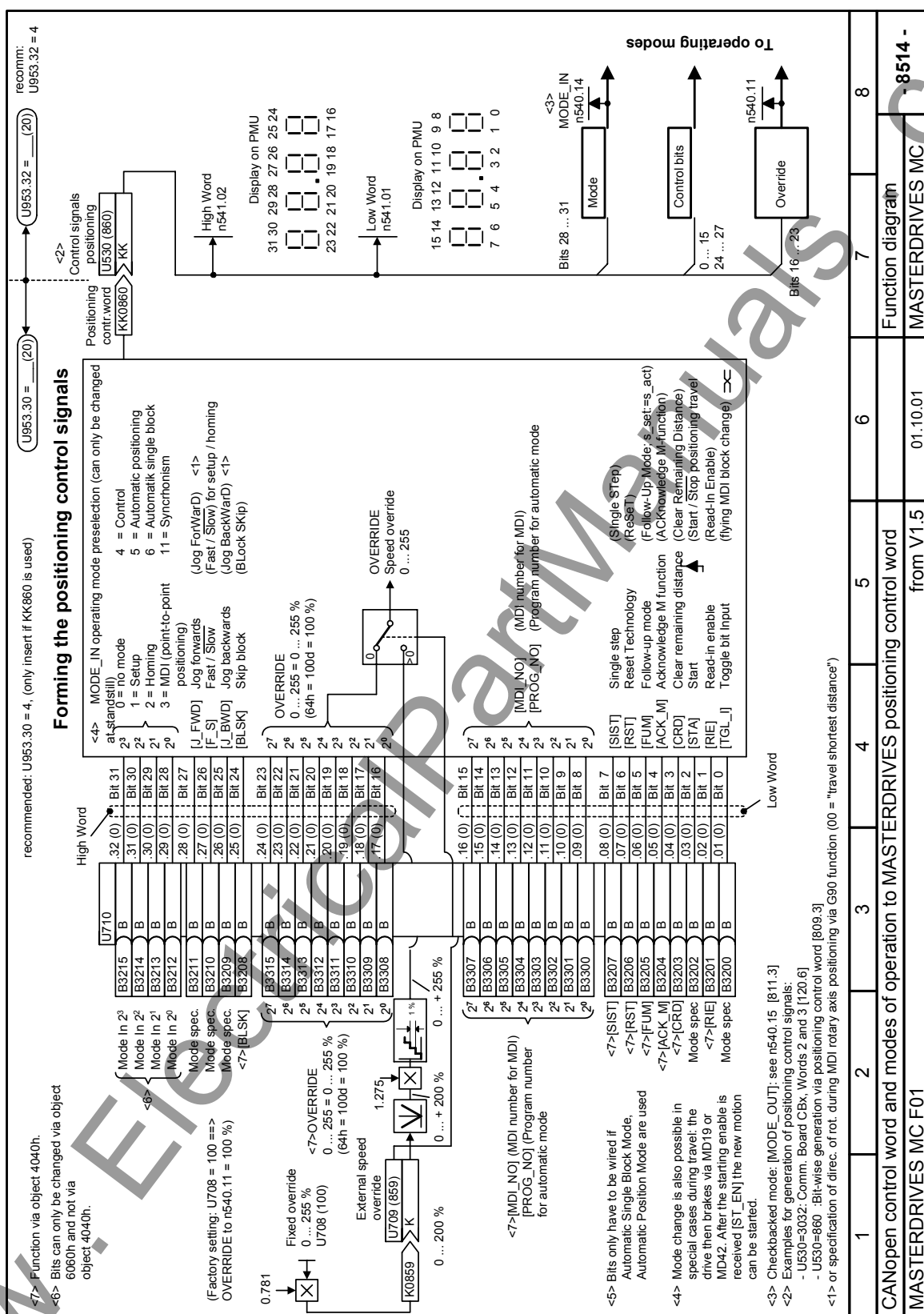








6SE7087-6QX70 (Version AD) Siemens AG
Compendium Motion Control SIMOVERT MASTERDRIVES



Parameter file for the control of positioning/synchronous operation via CBC field bus interface CANopen

Parameter file for the control of positioning/synchronous operation via CBC field bus interface CANopen							
1	2	3	4	5	6	7	8
The basic communication connections for data transfer via CANopen are made with these parameters (in shaded boxes) will need to be adapted to your application. Open the script file on your DriveMonitor CD using a text editor and enter the settings appropriate to your application. Then save the file and load it to the device using DriveMonitor. You will then need to set the motor and controller parameters. The script file is stored under the following name on your DriveMonitor CD ROM:  DriveMonitor for WINDOWS 95 and later: - MCF01.SSC (load this scriptfile, applies equally to Compact PLUS, Compact and chassis units)							
CB parameters : Receive PDOs P711.01=65281 ;RPDO 1 (RPDO 1, asynchronous) P712.01=0 ;RPDO 2 P713.01=0 ;RPDO 3 P714.01=0 ;RPDO 4 Transmit PDOs P715.01=65281 ;PDO 1 (RPDO 1, asynchronous) P716.01=0 ;PDO 2 P717.01=0 ;PDO 3 P718.01=0 ;PDO 4 CBC parameterization P719.01=4546 ;Device, device response to life guarding P720.01=4 ;CAN bus baud rate (125 kB) P721.01=1 ;CAN profile selection (CANopen) P722.01=0 ;Message failure time P918.01=5 ;CB bus address P053.00=7 ;Parameterization enable Transmit data Transmit PDO 1 P734.01=0431 ;Status word via BIN/CON converter P734.02=0315 ;Positioning status word, high P734.03=0315 ;Positioning status word, low P734.04=0250 ;Faults/alarms Transmit PDO 2 P734.05=0 ;TPDO 2 word 1 P734.06=0 ;TPDO 2 word 2 P734.07=0 ;TPDO 2 word 3 P734.08=0 ;TPDO 2 word 4 Transmit PDO 3 P734.09=0 ;TPDO 3 word 1 P734.10=0 ;TPDO 3 word 2 P734.11=0 ;TPDO 3 word 3 P734.12=0 ;TPDO 3 word 4 Transmit PDO 4 P734.13=0 ;TPDO 4 word 1 P734.14=0 ;TPDO 4 word 2 P734.15=0 ;TPDO 4 word 3 P734.16=0 ;TPDO 4 word 4 Control word 1 P554.01=3100 ;Src ON/OFF1 BICO DS1 P555.01=3101 ;Src1_OFF2(electr.) BICO DS1 P558.01=3102 ;Src1_OFF3(QStop) BICO DS1 P561.01=3103 ;Src InvRelease P562.01=3104 ;Src RampGenRel P563.01=3105 ;Src RampGenStop P564.01=3106 ;Src Selp Release P565.01=3107 ;Src Fault Reset							
Status word 1 at bin/con converter 1 U076.01=100 ;Ready to start at BIN/CON converter U076.02=102 ;Ready to operate BIN/CON converter U076.03=104 ;Run at BIN/CON converter U076.04=106 ;Fault active at BIN/CON converter U076.05=108 ;OFF2 active at BIN/CON converter U076.06=110 ;OFF3 active at BIN/CON converter U076.07=112 ;Starting lockout at BIN/CON converter U076.08=114 ;Alarm active at BIN/CON converter U076.09=0 ;Freely connectable by user U076.10=1 ;PZD control must always be 1 U076.11=476 ;target_L reached at BIN/CON converter U076.12=0 ;reserved U076.13=0 ;reserved U076.14=0 ;reserved U076.15=0 ;Freely connectable by user U076.16=0 ;Freely connectable by user U552.89=4 ;Time slot Setpoint processing {FD -320-} P443.01=131 ;Src MainSetpoint BICO DS1 (from position controller) P772.00=648 ;Src EnRGenByp Position sensing motor encoder {FD -330-} P472.00=302 ;Src Pos SetV P174.00=301 ;Src Pos CorrN P175.00=304 P178.00=16 ;Src Rough pulse BIN Input 4 Closed-loop position control {FD -340-} P210.01=1 ;Src1 RelPosReg P212.01=530 ;Src Ctrl Selp Technology option F1 U531.00=432 ;Src G-unc MDI U532.00=415 ;Src Position MDI (KK30xx) U533.00=416 ;Src Speed MDI (KK30xx) U535.00=120 ;Src PosAg V U530.00=860 ;Src Ctr Signals							
Formation of positioning signals U710.01=3200 ;TGL_1 (Toggle-Bit Input) U710.02=3201 ;RIE (Delete distance to go) U710.03=3202 ;STA (Start) U710.04=3203 ;CRD (Delete distance to go) U710.05=3204 ;ACK_M (Acknowledgement M function) U710.06=617 ;FUM (Follow-up mode) U710.07=3206 ;RST (Reset) U710.08=3207 ;SIST (Single step) U710.09=0 ;MDI_NO/PROG_NO 2'0 U710.10=0 ;MDI_NO/PROG_NO 2'1 U710.11=0 ;MDI_NO/PROG_NO 2'2 U710.12=0 ;MDI_NO/PROG_NO 2'3 U710.13=0 ;MDI_NO/PROG_NO 2'4 U710.14=0 ;MDI_NO/PROG_NO 2'5 U710.15=0 ;MDI_NO/PROG_NO 2'6 U710.16=0 ;MDI_NO/PROG_NO 2'7 U710.17=0 ;OVERRRIDE 2'0 U710.18=0 ;OVERRRIDE 2'1 U710.19=0 ;OVERRRIDE 2'2 U710.20=0 ;OVERRRIDE 2'3 U710.21=0 ;OVERRRIDE 2'4 U710.22=0 ;OVERRRIDE 2'5 U710.23=0 ;OVERRRIDE 2'6 U710.24=0 ;OVERRRIDE 2'7 U710.25=3208 ;BLSK (Skip block) U710.26=3209 ;J_BWD (Jog backwards) U710.27=3210 ;F_S (Quickslow) U710.28=3211 ;J_FWD (Jog forwards) U710.29=3212 ;MODE_IN 2'0 U710.30=3213 ;MODE_IN 2'1 U710.31=3214 ;MODE_IN 2'2 U710.32=3215 ;MODE_IN 2'3 Time slots U953.30=4 ;Control signals positioning U953.32=4 ;Positioning							
Parameter file "positioning with F01 via CANopen"						Function diagram	
MASTERDRIVES MC F01						MASTERDRIVES MC	
01.10.01						- 8515 -	

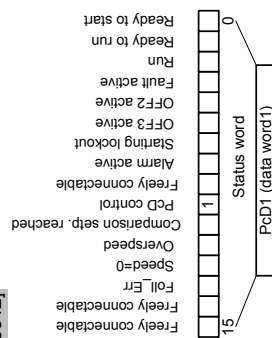
Parameter file for the control of positioning/synchronous operation via CBC field bus interface CANopen							
1	2	3	4	5	6	7	8
Free block interconnections Speed setpoint switchover for homing and control Mode-dependent speed setpoint switchover This circuit implements switchover of the speed setpoint in P212 depending on whether control or homing is selected. If this circuit is omitted, the speed setpoint is always supplied by the technology. Object 60ffh target velocity has no effect in this case. Analog signal selector switch 5 (2-word) [KK0530] Control setpoint U184.00=618 : Switchover of "Mode in control" FFP[8510.3] U185.01=311 : V-set from technology U186.02=478 : V-set fixed setpoint U018 with SDO or DPR U952.08=4 : Time slot AND element 18 [B0618] U238.01=649 : Input 1 (Mode in 2nd inverted) U238.02=650 : Input 2 (Mode in 2nd inverted) U238.03=3214 : Input 3 (Mode in 2nd) U952.92=4 : Time slot Inverter 9 [B0649] Invert Mode in 2nd U259.00=3212 : Input 2nd U952.53=4 : Time slot Inverter 10 [B0650] Invert Mode in 2nd U260.00=3213 : Input 2nd U952.55=4 : Time slot Interconnection for the target_reached bit in status word This circuit implements the target_reached bit in the status word with a single-word limit-value monitor. Limit-value monitor with smoothing (1 word) U136.01=151 : n(act.smooth) U136.02=418 : V-set via SDO (U018) or via DPR (KK30xx) U138.00=0.20 : Hysteresis U139.00=2 : Function A=B U951.18=4 : Time slot Switchover absolute/relative positioning This circuit implements the switchover between the G functions G90 (absolute positioning) and G91 (relative positioning) in technology mode MDI if the axis is traversing with MDI block 0 (fast MDI). Binector/connector converter [K0432] (G functions for MDI via CAN) Acceleration override = G30 U078.01=0 : 2nd U078.02=1 : 2nd U078.03=1 : 2nd U078.04=1 : 2nd U078.05=1 : 2nd U078.06=0 : 2nd U078.07=0 : 2nd U078.08=0 : 2nd G90 / G91 switchover U078.09=3108 : 2nd U078.10=1 : 2nd U078.11=0 : 2nd U078.12=1 : 2nd U078.13=1 : 2nd U078.14=0 : 2nd U078.15=1 : 2nd U078.16=0 : 2nd U952.90=10 : Time slot Follow-up mode - FUM - (FD[8511]) The output of this circuit switches the technology to follow-up mode if positioning is aborted. AND element 17 [B0617] (Follow-up mode ON after abort) U237.01=105 : Input 1 (no operation) U237.02=647 : Input 2 (DRS inverted) U237.03=648 : Input 3 (not in control mode I) U952.54=6 : Time slot Inverter 7 [B0647] Invert position reached (DRS) U257.00=355 : Input (DRS) U951.94=6 : Time slot Inverter 8 [B0648] Invert control mode BA U258.00=618 : Input (control) U952.41=6 : Time slot							
Parameter file "Positioning with F01 via CANopen"				from V1.5		Function diagram	
MASTERDRIVES MC F01				01.10.01		MASTERDRIVES MC	
						- 8516 -	

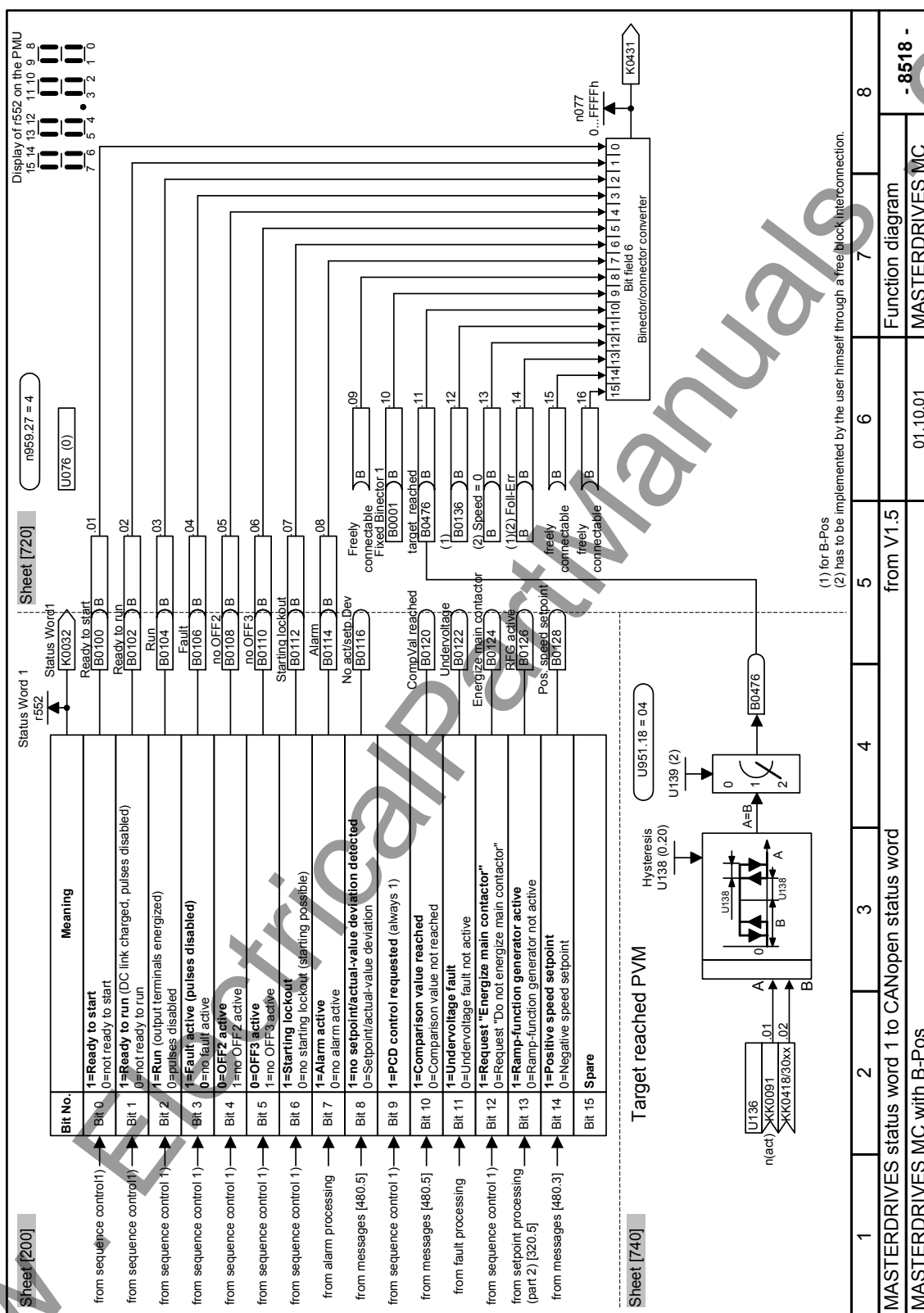
The basic communication connections for data transfer via CANopen are made with these parameter settings. Some parameters (in shaded boxes) will need to be adapted to your application. Open the script file on your DriveMonitor CD using a text editor and enter the settings appropriate to your application. Then save the file and load it to the device using DriveMonitor. You will then need to set the motor and controller parameters.

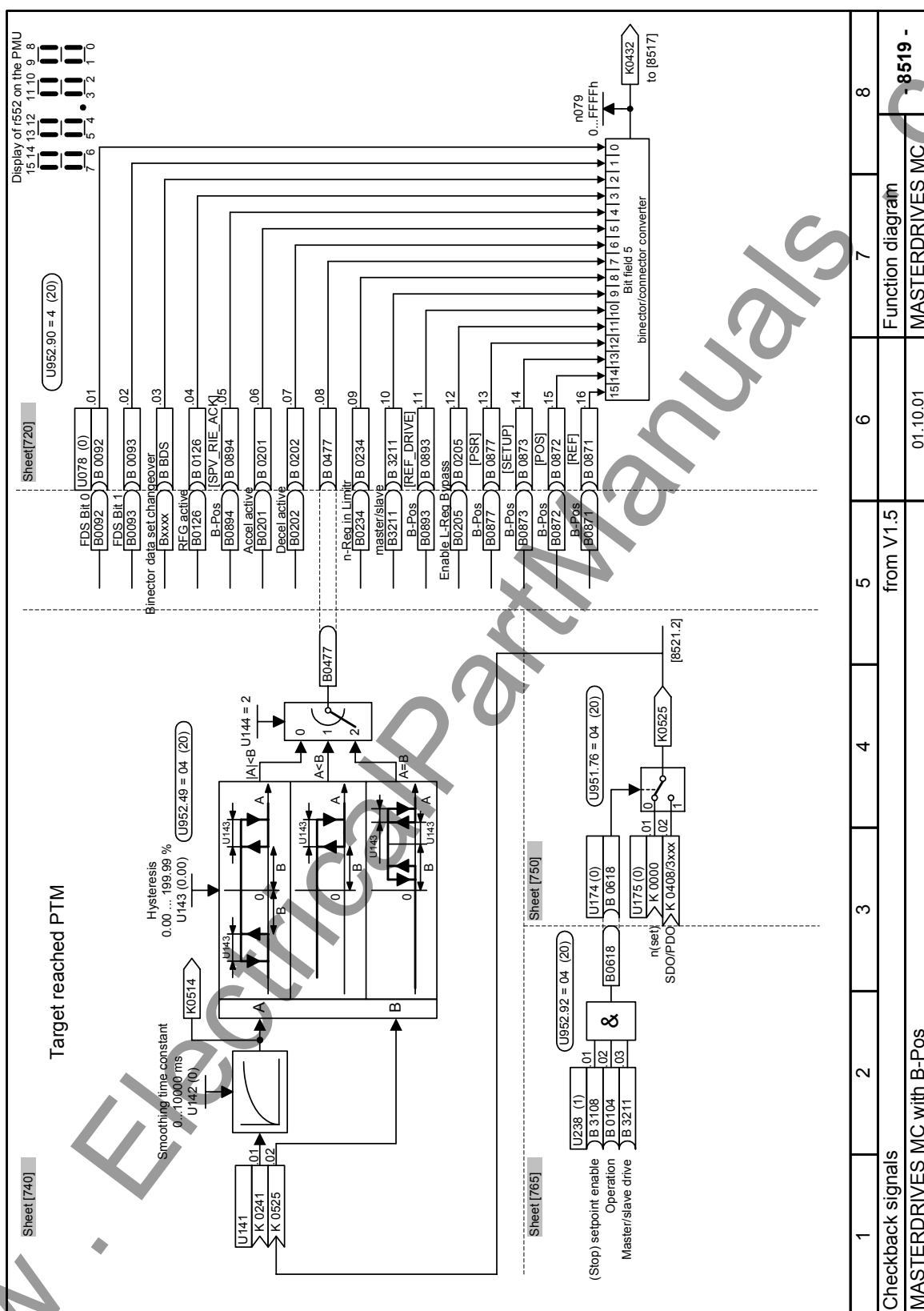
The script file is stored under the following name on your DriveMonitor CD ROM:

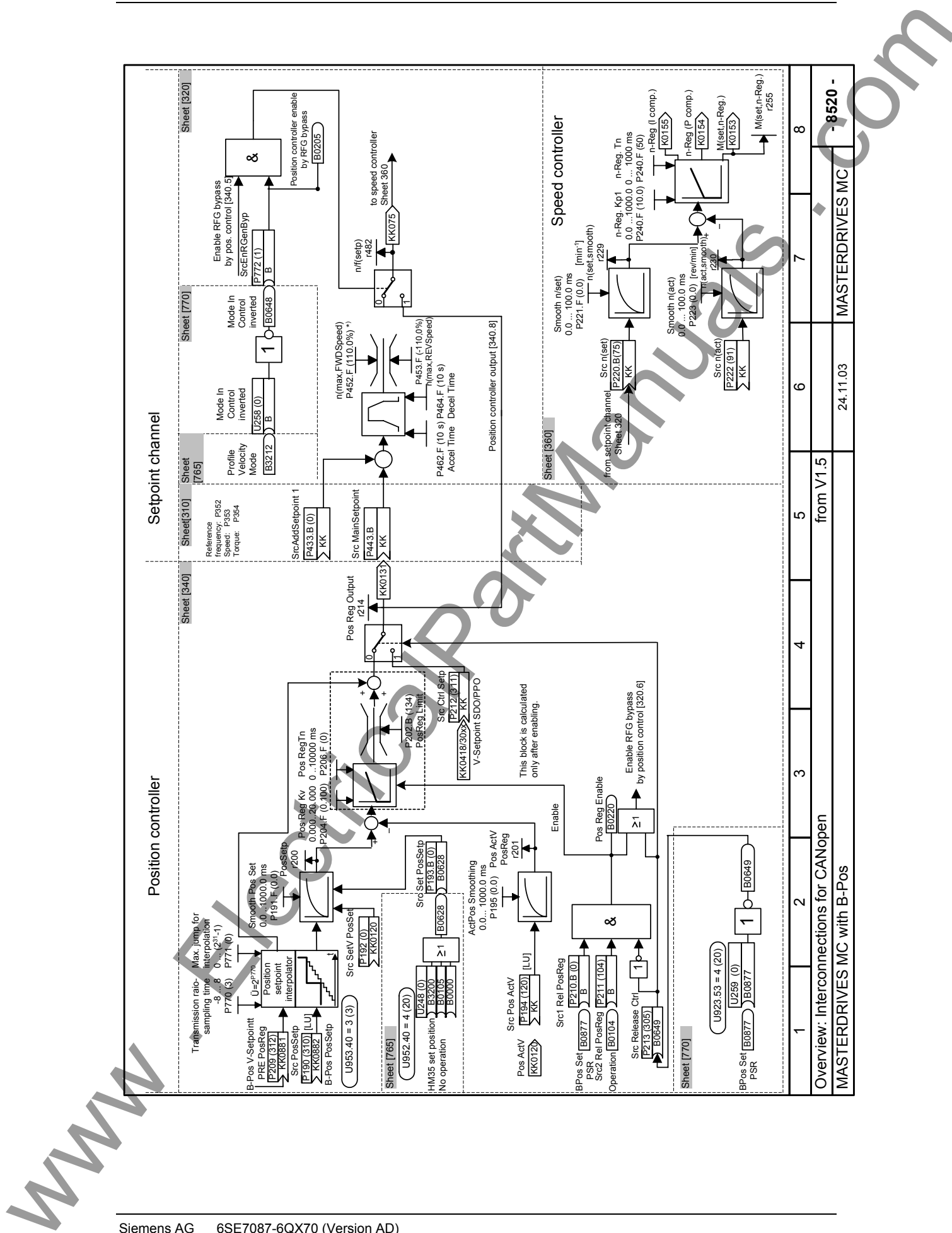


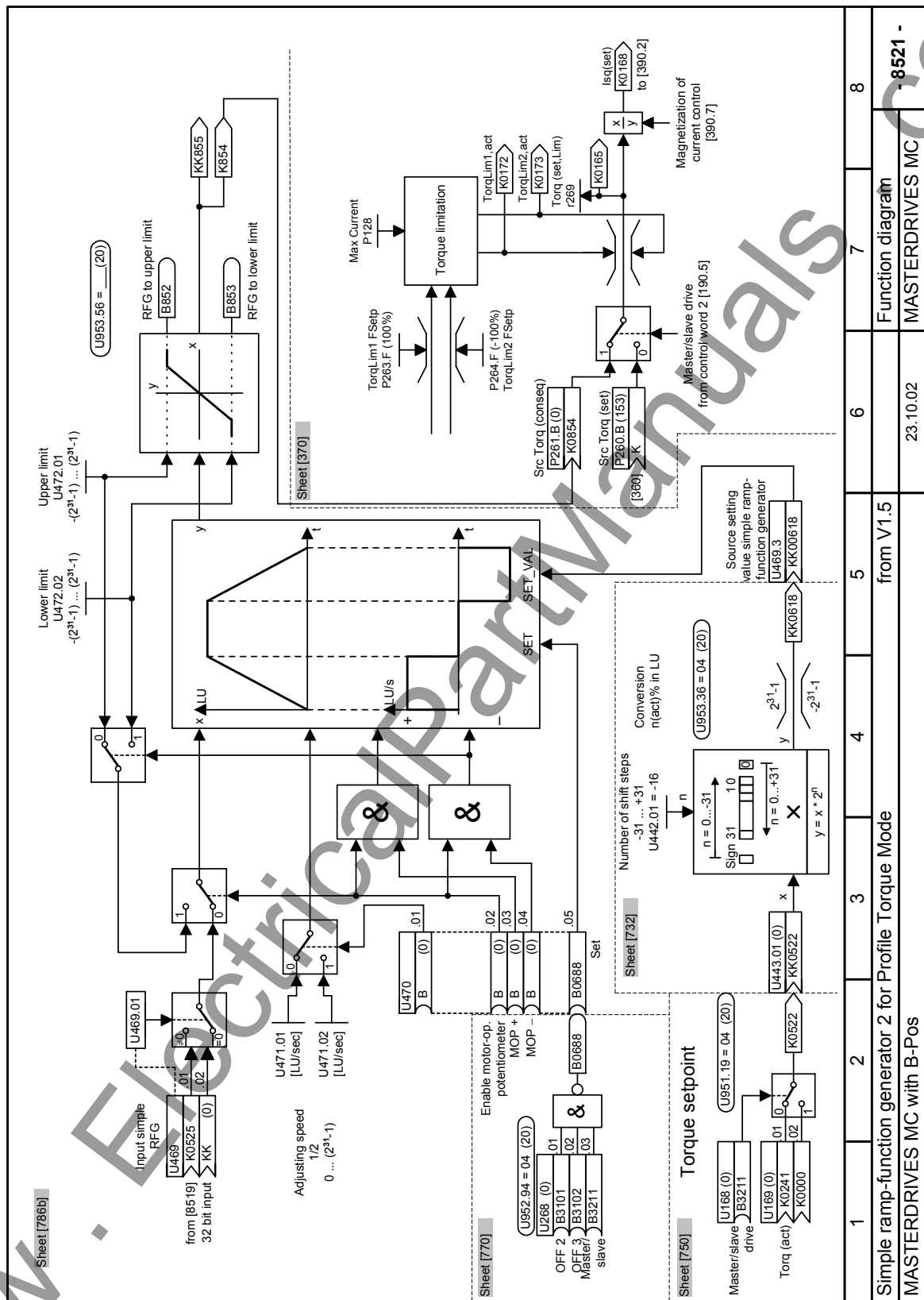
- MCF01.SSC
(load this script file, applies equally to Compact PLUS, Compact and chassis units)

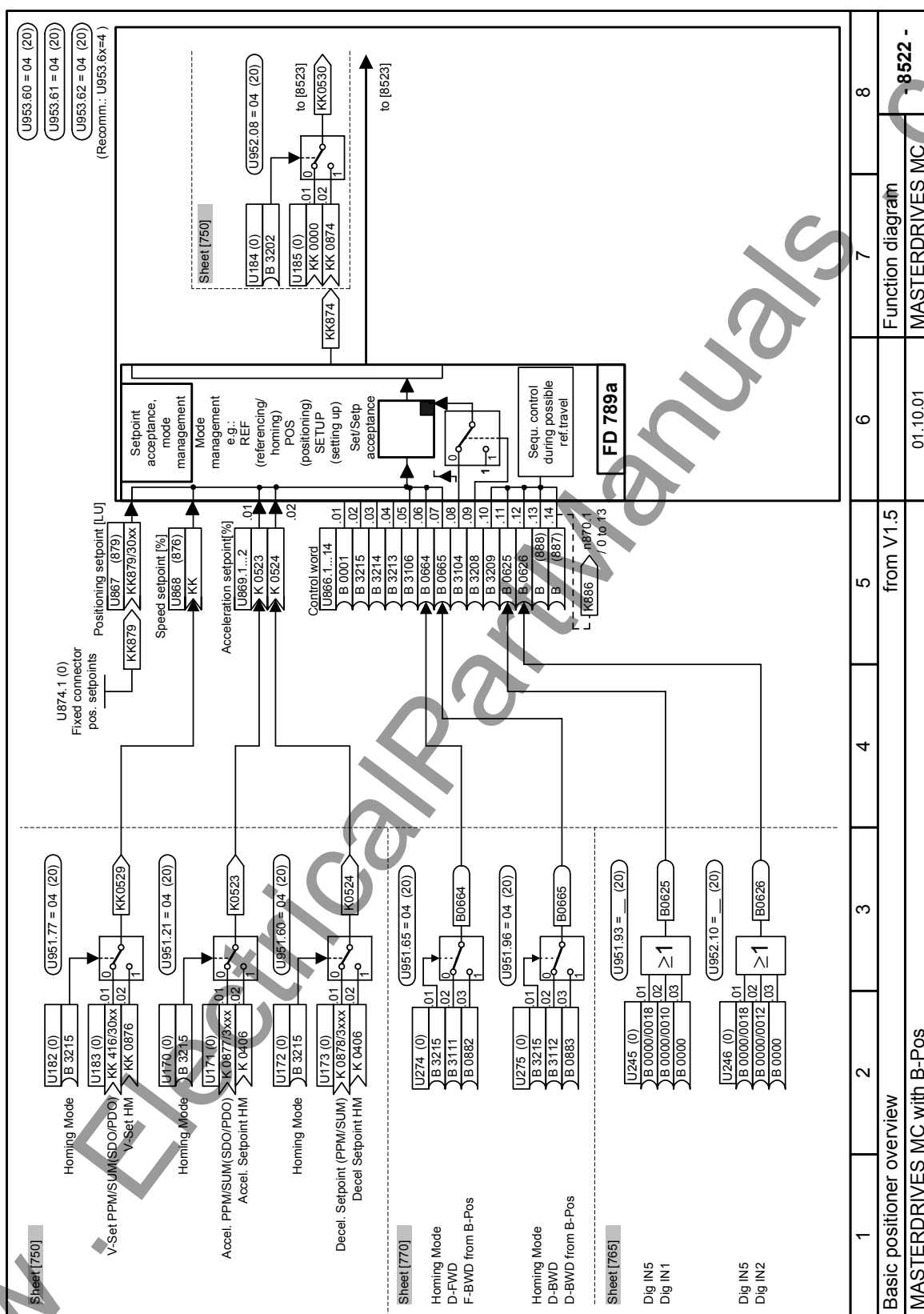


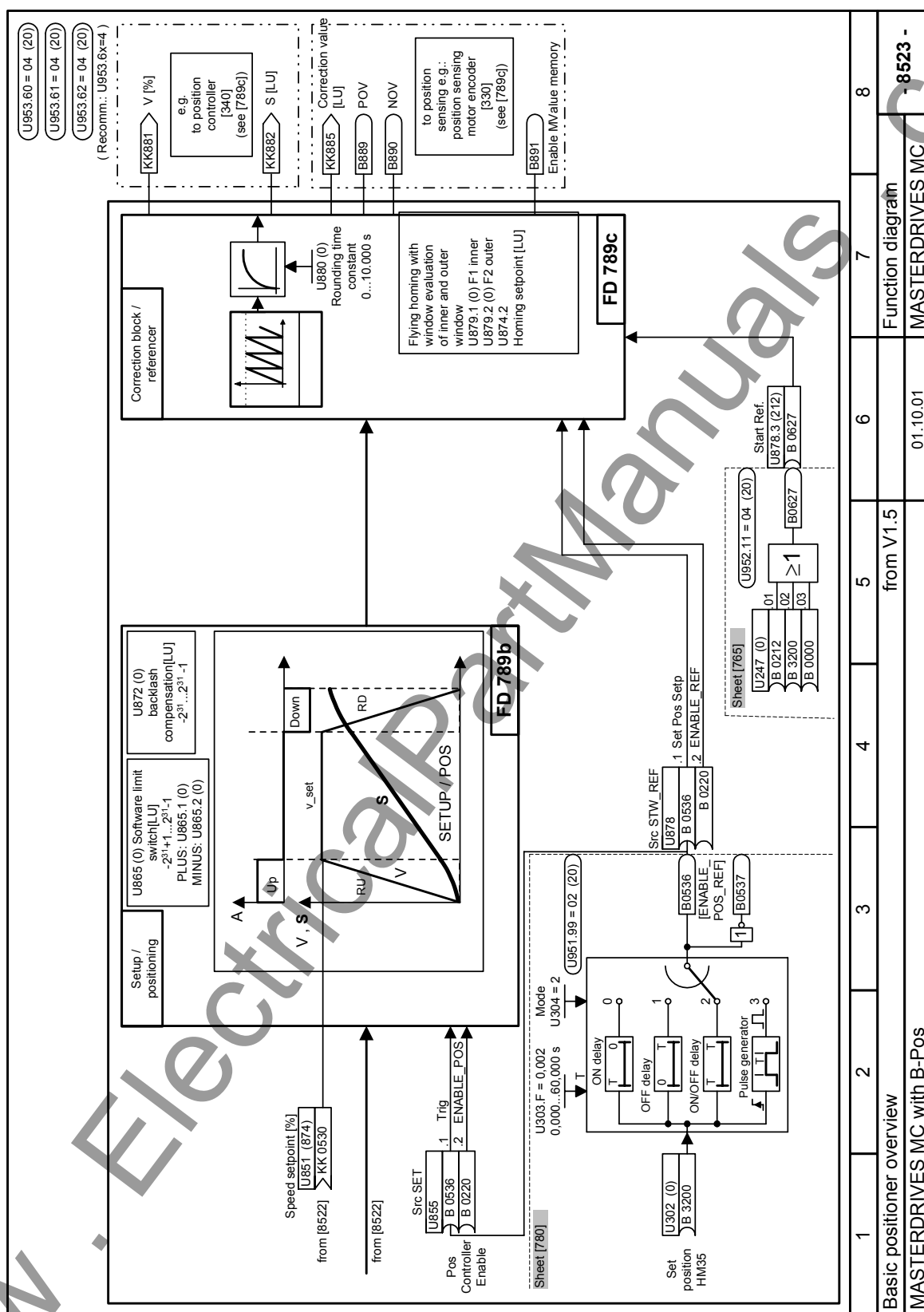












Parameter file for controlling positioning / setup via field bus interface CBC CANopen																			
Parameter file (FD-120):																			
The basic communication connections for data transfer via CANopen are made with these parameter settings. Some parameters (in shaded boxes) will need to be adapted to your application. Open the script file on your DriveMonitor CD using a text editor and enter the settings appropriate to your application. Then save the file and load it to the device using DriveMonitor. You will then need to set the motor and controller parameters. The script file is stored under the following name on your DriveMonitor CD ROM:  DriveMonitor for WINDOWS 95 and later: MCEPOS.SSC (load this script file, applies equally for Compact PLUS, Compact and Chassis units)																			
Receive PDOs P711.01=65281 :RPDO 1 (Control word, asynchronous) P712.01=0 :RPDO 2 P713.01=0 :RPDO 3 P714.01=0 :RPDO 4 Transmit PDOs P715.01=65282 :TPDO 1 (status word, modopdisplay async.) P716.01=0 :TPDO 2 P717.01=0 :TPDO 3 P718.01=0 :TPDO 4 CBC parameterization P719.01=4446 :Device, device response to life guarding P720.01=4 :CANBus baud rate (125 kB) P721.01=1 :CAN Profile selection (CANopen) P722.01=0 :Message failure time P918.01=5 :CB bus address P053.00=7 :Parameterization enable Transmit data (FD-125): Transmit PDO 1 P734.01=0431 :Status word via BIN/CON converter P734.02=889 :Status word B-Pos P734.03=432 :Status word 2 B-Pos to BIN/CON Conv 2 P734.04=0250 :Faults/alarms Transmit PDO 2 P734.05=0 :TPDO 2 word 1 P734.06=0 :TPDO 2 word 2 P734.07=0 :TPDO 2 word 3 P734.08=0 :TPDO 2 word 4 Transmit PDO 3 P734.09=0 :TPDO 3 word 1 P734.10=0 :TPDO 3 word 2 P734.11=0 :TPDO 3 word 3 P734.12=0 :TPDO 3 word 4 Transmit PDO 4 P734.13=0 :TPDO 4 word 1 P734.14=0 :TPDO 4 word 2 P734.15=0 :TPDO 4 word 3 P734.16=0 :TPDO 4 word 4 Control word 1 (FD-180): P554.01=3100 :SRC ON / OFF BICO DS1 P555.01=3101 :SRC1 OFF2 (coast) BICO DS1 P558.01=3102 :SRC1 OFF3 (QStop) BICO DS1 P561.01=3103 :SRC Inverter Enable P564.01=3108 :SRC Setpoint Enable P565.01=3107 :SRC Acknowledge Control word 2 (FD-190): P587.01=3211 :SRC Master/slave drive Digital inputs (FD-90): P648.01=4 :Accept PosMeasV with falling edge																			
Status word 1 to Bin/Con converter 1 (FD-720): U076.01=100 :Ready to start at BIN/CON converter U076.02=102 :Ready to run at BIN/CON converter U076.03=104 :Operation at BIN/CON converter U076.04=106 :Fault active at BIN/CON converter U076.05=108 :OFF 2 active at BIN/CON converter U076.06=110 :OFF 3 active at BIN/CON converter U076.07=112 :Starting lockout at BIN/CON converter U076.08=114 :Alarm active at BIN/CON converter U076.09=0 :Freely connectable by user U076.10=1 :PCD control requested must always be 1 U076.11=476 :Target reached at BIN/CON converter U076.12=136 :Overspeed: Internal Limit active PVM, SUM U076.13=0 :Speed=0, must be generated via FBS U076.14=0 :Following Error, generate via FBS U076.15=0 :Freely connectable by user U076.16=0 :Freely connectable by user U952.99=4 :Time slot Status signals B-Pos to Bin/Con converter 2 (FD-720): U078.01=92 :FDS, Bit 0 U078.02=93 :FDS, Bit 1 U078.03=0 :BICO data set via FBS or from STW2 U078.04=126 :RFG active, from RFG U078.05=894 :SPV_RIE_ACKN from B-Pos U078.06=201 :Acceleration active from RFG U078.07=202 :Deceleration active from HLG U078.08=477 :Output limit monitor, Target reached PTM U078.09=234 :T (gr) act, Limit Active PTM U078.10=3211 :Master slave fed back from CBC U078.11=893 :REF_DRIVE from B-Pos U078.12=205 :Enable pos. controller from RFG bypass U078.13=877 :PSR from B-Pos mode manager U078.14=873 :SETUP from B-Pos mode manager U078.15=872 :POS from B-Pos mode manager U078.16=871 :REF from B-Pos mode manager U952.90=4 :Time slot Setpoint processing (FD-320): P443.01=131 :Src main setpoint BICO DS1 (from pos controller) P772.00=648 :Src Enable RFG bypass from inverter U258 Position sensing (FD-330): P130.00=3 :Motor encoder Motor encoder (FD-330): P172.00=880 :Src PosSetV from Ref. Setpoint U874.02 P173.00=3200 :Src Set Pos, Bit 0 from CBC receive con. P174.00=885 :Src PosCorrV from corr'n block B-POS P175.01=889 :POV from correction block B-POS P175.01=890 :NOV from correction block B-POS P179.00=891 :Src En MeasV mem. from corr'n block B-POS P183.01=1 :PosSens for mot enc. resol or Enco enable P183.02=0 :No zero point offset from encoder																			
Torque limitation (FD-370): P261.01=854 :Src T (conseq) from simple RFG 2 Position control (FD-340): P190.01=882 :Src position setpoint from B-Pos P192.01=120 :Src SetV PosSet (PosActV) P193.01=628 :Src Set PosSet from output OR U248 P209.01=881 :PRE PosReg from B-Pos P210.01=877 :Src1 Rel Pos Reg PSR from B-Pos P212.01=418 :Src Control Setp, Target Velocity U018(SDO) P213.01=649 :Src Release Control, from Inverter U259 B-Pos control signals (FD-788a): U858.01=0 :Axis cycle length B-Pos/SETUP, 0:Linear axis U858.02=1 :Axis cycle length CORR/REF, 1:Index2=Index1 U866.01=1 :ENABLE POS/REF U866.02=3215 :POS ON from CBC, K3002, Bit 15 U866.03=3214 :POS ON from CBC, K3002, Bit 14 U866.04=3213 :SETUP ON from CBC, K3002, Bit 13 U866.05=3106 :POS TYP from CBC, K3001, Bit6 abs/rel U866.06=664 :D_BWD from binary signal changeover U274 U866.07=665 :D_FWD from binary signal changeover U275 U866.08=3104 :SPV_RIE from CBC, K3001, Bit4, New setpoint U866.09=3208 :SPV_RIE_TYP from CBC, K3002, Bit8, 1=SUM U866.10=3209 :REF_TYP from fixed binector B-Pos U866.11=625 :REF_BWD_STOP, Outp OR U245, reverse cam U866.12=626 :REF_FWD_STOP, Outp OR U246, reverse cam U867.0=879 :Position setpoint from fixed setpoint U874.01 U868.0=529 :V-set from analog changeover U182/U183 U869.01=523 :Analog signal changeover U170/U171 U869.02=524 :Analog signal changeover U172/U173 U882.0=104 :Reset Set setpoint from Status word 1 Bit 2, run U953.60=4 :Time slot (FD-789b): U855.01=536 :Set Trigger SetV from timer for HM 35 U855.02=220 :ENABLE_POS from position controller enable U851.00=530 :V-set output analog ch. U184/U185 (stop) U859.00=100 :POS_OK window width = 100LU U864.00=0.10 :POS_OK decel time = 0.1s U865.01=0 :Software limit switch -plus U865.02=0 :Software limit switch -minus U953.61=4 :Time slot (FD-789c): U878.01=536 :Set pos. setp. from timer due to HM 35 U878.02=220 :ENABLE_REF from pos. control enable U878.03=627 :Start-REF from OR U247, set pos B212 U953.62=4 :Time slot																			
Function diagram																			
MASTERDRIVES MC with B-Pos																			
17.12.03																			
from V1.5																			
6																			
7																			
8																			
- 8524 -																			

Parameter file for the control of positioning/setup via CBC field bus interface CANopen

Free block interconnections									
The basic communication interconnections for data transfer via CANOpen are made with these parameter settings. Some parameters (in shaded boxes) will need to be adapted to your application. Open the script file on your DriveMonitor CD using a text editor and enter the settings appropriate to your application. Then save the file and load it to the device using DriveMonitor. You will then need to set the motor and controller parameters. The script file is stored under the following name on your DriveMonitor CD:									
Target Reached Profile Velocity Mode(FD-740-) U136.01=151 :Speed setpoint U136.02=418 :Output of fixed setpoint U018.60FF n, mapped U137.00=500 :Smoothing time constant U168.00=2.00 :Hysteresis tolerance range 2% U139.00=2 :Mode A=B U151.18=4 :Time slot Target Reached Profile Torque Mode(FD-740-) U141.01=241 :Torque actual value U142.00=1000 :Smoothing time constant U143.00=2.00 :Hysteresis tolerance range 2% U144.00=2 :Mode A=B U152.49=4 :Time slot {FD-750-} U160.01=525 :Output of analog signal changeover U175 U160.02=1000 :Smoothing time constant U160.03=2.00 :Hysteresis tolerance range 2% U160.04=2 :Mode A=B U153.18=4 :Time slot Changeover Acceleration in PPM,SUM or HM U171.01=3215 :Bit15 of CB receive connector 2,1=HM U171.02=877 :Fixed conn. B-POS U873.2, accel. n, mapped U171.03=406 :Fixed connector U006 accel. setp. in HM U171.04=4 :Time slot Changeover Deceleration in PPM,SUM or HM U172.01=3215 :Bit15 of CB receive connector 2,1=HM U172.02=878 :Fixed conn. B-POS U873.2, accel. n, mapped U172.03=406 :Fixed connector U006 accel. setp. in HM U172.04=4 :Time slot Target Torque for PTM, Implementation of Stop U174.00=618 :Output AND element U238 U175.01=0 :Time slot U175.02=408 :Fixed connector U008, torque setp. in PTM U175.03=4 :Time slot Changeover Profile Velocity in PPM,SUM or HM U182.00=3215 :Bit15 of CB receive connector 2,1=HM U183.01=416 :Fixed connector U016,V-setp for PPM,SUM U183.02=876 :Fixed connector B-POS U873.1,V in HM U183.03=4 :Time slot Stop in PPM, SUM and HM U184.00=3202 :Bit2,2,CBC K3002.0=Stop for PPM,SUM,HM U185.01=0 :Valid speed setpoint B-Pos U185.02=874 :Time slot U185.03=4 :Time slot {FD-765-} Changeover signal for analog changeover for Stop in PTM U238.01=3108 :Stop for PPM and PTM U238.02=104 :Operation from status word 1 U238.03=3211 :Bit11 from CBC K3002.1=PTM U238.04=4 :Time slot									
REF_BWD_STOP, wiring of CBC depending on HM U245.03=0 :01,02 wired from CBC U245.04=4 :Time slot REF_FWD_STOP, wiring of CBC depending on HM U246.03=0 :01,02 wired from CBC U246.04=4 :Time slot Measured value valid for reference block B-POS (HM 35) U247.01=212 :Measured value valid from position sensing U247.02=3200 :Bit 0 from K3002, set position from HM 35 U247.03=0 :Time slot Set Pos Setp for position controller U248.01=3200 :Bit 0 from K3002, set position from HM 35 U248.02=105 :No operation from status word 1 U248.03=0 :Time slot {FD-770-} U258.00=3212 :Bit 12 from K3002, 1=PVM U258.01=4 :Time slot Inverter U259 to P213 (Src Release Ctrl) U259.00=877 :PSR from B-POS U259.01=4 :Time slot NAND U268 to U470.05 (set simple RGEN 2 for PTM) U268.01=3101 :OFF 2 from CBC, K3001 bit 1 U268.02=3102 :OFF 3 from CBC, K3001 bit 2 U268.03=3111 :Bit 11 from CBC, K3002, 1=PTM U268.04=4 :Time slot Binary changeover U274 to B-Pos D_FWD, changeover Hmand PPM,SUM U274.01=3215 :Bit 15 from CBC K3002, 1=HM U274.02=3111 :Bit 11 from CBC K3001 U274.03=882 :D_FWD from fixed binector, B-POS U274.04=4 :Time slot Binary changeover U275 to B-Pos D_FWD, changeover Hmand PPM,SUM U275.01=3215 :Bit 15 from CBC K3002, 1=HM U275.02=3112 :Bit 12 from CBC K3001 U275.03=883 :D_BWD from fixed binector, B-POS U275.04=4 :Time slot {FD-780-} Time element U302,U304 to U855.01 and U878.01 from B-POS U302.00=3200 :Set position from CBC K3002, bit 0 U303.01=0.002 :2ms time delay U304.00=2 :ON/OFF delay U304.01=4 :Time slot {FD-732-} Shift multiplier U442,U443 to U469.03,Src SetV Rgen U442.01=16 :Division by 2^16, adjustment KK->K U443.01=522 :Select torque SetV, changeover U168,169 U469.03=4 :Time slot									
{FD-786b-} Simple RGen for PTM U469,U472,K854 = Tset U469.01=525 :Analog changeover U174/U175 for stop U469.02=0 :Output shift multiplier setting value U470.01=0 :Time slot U470.03=0 :Set, output NAND element U268 U471.01=32768 :Torque Slope, 200% per second U472.01=16383 :100%, upper torque limit value U472.02=16384 :100%, lower torque limit value U472.03=0 :Time slot Display parameter R047.05: output current(FD-30-) P046.05=242 :Output current F151 Initialization fault(FD-710-) U064.00=3201 :Bit 1 from 2nd CBC receive connector U065.62=7 :Time slot									

1

2

3

4

5

6

7

8

Parameter file "Positioning with B-POS via CANOpen"

Function diagram

MASTERDRIVES MC with B-Pos

MASTERDRIVES MC

01.10.01

- 8525 -

8.5.13 Terms and abbreviations

B-Pos	<u>B</u> asic <u>P</u> ositioning
CAL	<u>C</u> AN <u>A</u> pplication <u>L</u> ayer
CAN	<u>C</u> ontroller <u>A</u> rea <u>N</u> etwork
CBC	Communication board for CAN (CANopen)
CiA DS	<u>C</u> AN in <u>A</u> utomation <u>D</u> raft <u>S</u> tandard
CiA DSP	<u>C</u> AN in <u>A</u> utomation <u>D</u> raft <u>S</u> tandard <u>P</u> roposal
F01	Technology option for MASTERDRIVES MC
HM	<u>H</u> oming <u>M</u> ode
IND	<u>P</u> arameter <u>I</u> ndex
MC	<u>M</u> otion <u>C</u> ontrol
MDx	<u>M</u> achine <u>D</u> ata
NMT	<u>N</u> etwork <u>M</u> anagement
PDO	<u>P</u> rocess <u>D</u> ata <u>O</u> bject
PFSF	<u>P</u> osition- <u>F</u> eedback <u>S</u> caling <u>F</u> actor
PKE	<u>P</u> arameter Identifier
PKW	<u>P</u> arameter Identifier Value
PNU	<u>P</u> arameter Number
PPM	<u>P</u> rofile <u>P</u> osition <u>M</u> ode
PSH	<u>P</u> ositioning with <u>S</u> etup or <u>H</u> oming
PTM	<u>P</u> rofile <u>T</u> orque <u>M</u> ode
PVM	<u>P</u> rofile <u>V</u> elocity <u>M</u> ode
PWE	<u>P</u> arameter Value
Pxxx.xx	Parameter with number
R_PDO	<u>R</u> eceive <u>P</u> rocess <u>D</u> ata <u>O</u> bject
SDO	<u>S</u> ervice <u>D</u> ata <u>O</u> bject
SPM	Toggle bit for parameter change report (not supported by the CBC)
SUM	<u>S</u> etup <u>M</u> ode
TI	<u>T</u> ask Identifier
T_PDO	<u>T</u> ransmit <u>P</u> rocess <u>D</u> ata <u>O</u> bject
Uxxx.xx	Parameter above 2000

9 Technology Option F01

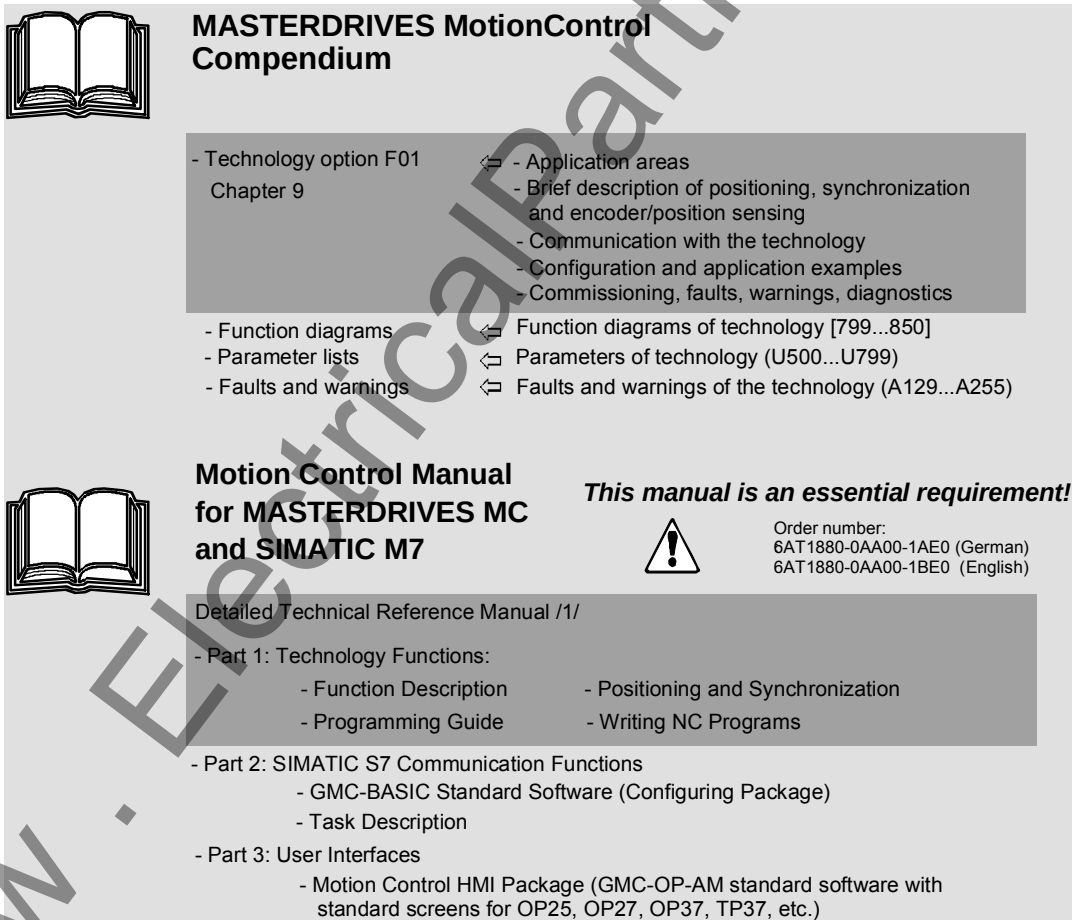
9.1 Enabling technology option F01

Technology option F01 can only be used on a MASTERDRIVES unit if the unit was supplied from the plant of manufacture with option F01 already enabled or if the option was enabled afterwards by entering a PIN number. Please consult sheet [850] of the function diagram to find out

- ◆ Whether option F01 is enabled on your MASTERDRIVES unit
- ◆ How to enable option F01 temporarily as a "demo version" for a period of 500 hours using the special PIN number
- ◆ How to enable option F01 as a "full version" after the MASTERDRIVES unit is supplied

9.2 Overview of the documentation

The figure below shows you an overview of the documentation available for technology option F01:



MASTERDRIVES MotionControl Compendium

- Technology option F01 Chapter 9
 - ⇨ - Application areas
 - ⇨ - Brief description of positioning, synchronization and encoder/position sensing
 - ⇨ - Communication with the technology
 - ⇨ - Configuration and application examples
 - ⇨ - Commissioning, faults, warnings, diagnostics
- Function diagrams ⇨ Function diagrams of technology [799...850]
- Parameter lists ⇨ Parameters of technology (U500...U799)
- Faults and warnings ⇨ Faults and warnings of the technology (A129...A255)

Motion Control Manual for MASTERDRIVES MC and SIMATIC M7

This manual is an essential requirement!

Order number:
6AT1880-0AA00-1AE0 (German)
6AT1880-0AA00-1BE0 (English)

Detailed Technical Reference Manual /1/

- Part 1: Technology Functions:
 - Function Description
 - Positioning and Synchronization
 - Programming Guide
 - Writing NC Programs
- Part 2: SIMATIC S7 Communication Functions
 - GMC-BASIC Standard Software (Configuring Package)
 - Task Description
- Part 3: User Interfaces
 - Motion Control HMI Package (GMC-OP-AM standard software with standard screens for OP25, OP27, OP37, TP37, etc.)

Fig. 9-1 Overview of the documentation for technology option F01

This chapter (Chapter 9) of the Compendium contains an overview of the position encoders that can be used for position sensing, and the systems used for their evaluation. It also describes the position controller and the technology functions for positioning and synchronization.

The "**Application areas**" section describes the positioning and synchronization functions contained in the MASTERDRIVES MC converters and the applications you can implement using these functions.

The "**Brief description of the technology functions**" section provides you with an overview of the technology used to implement the positioning and synchronization functions and the position sensing and control system. This section makes reference to the function diagram.

The "**Application examples**" section shows you how to configure the technology functions and connect them to the functions of the basic unit. In this section you will also find easy-to-understand application examples, suitable for your own studies, which will help you get started with the implementation of positioning and synchronization applications based on MASTERDRIVES MC.

In the "**Commissioning**" section you learn how to commission a positioning or synchronization axis step by step.

All technology functions are presented clearly in graphical form in the "**Function Diagrams**" chapter of the Compendium, sheets [799]...[850]. The position sensing and control systems are described in this section in [230...270, 330...340]. All references to function diagrams are quoted in brackets with the [sheet number].

The setting and monitoring parameters and the binectors and connectors for the technology functions are contained in the "**Parameter Lists**" chapter of the Compendium.

You will find detailed information on all the technology functions in Chapter 5 "Function description" of the "**Motion Control for MASTERDRIVES MC and SIMATIC M7**" manual /1/. This manual provides a complete function reference which you should consult in cases of doubt. It also contains a detailed description of the machine data, all the technological control and checkback signals and the timing diagrams for the execution of traversing movements in all operating modes. Section 6 of this manual contains a **Programming Guide** which you will need in order to write automatic NC programs.

NOTES

- In addition to the Compendium, **you need** the "Motion Control for MASTERDRIVES MC and SIMATIC M7" manual /1/ for the configuration and commissioning of technology option F01 (see the "References, software products and accessories" section).
- The  symbol refers to further information in other main chapters of the Compendium and other documents.

9.3 Application areas

The "Technology Software F01" software option contains the following functions:

- ◆ Positioning
- ◆ Angular synchronization

A MASTERDRIVES MC power converter with the "technology" software option can be ordered by specifying the MLFB extension "F01". The "technology" software option can also be enabled, using a PIN number, on a power converter which was not supplied with this option (e.g. if replacing the hardware during a service call; see the section entitled "Enabling technology option F01").

In the following section you will find a brief overview of the "Motion Control technology software" option and its possible applications. The subsequent section, "Brief description of the technology functions", provides you with more detailed information on the technical implementation of the technology functions.

NOTE

Technology functions synchronism (U953.33) and positioning (U953.32) must not be enabled simultaneously.

NOTE

If the technology functions are slotted into the time slot and the technology is not enabled with the PIN, diagnostics fault F063 occurs. The fault can only be eliminated by entering the correct PIN in U977.01 and U977.02 and subsequently switching the power supply off and then on again, or by removing the technology functions from the time slots again (by setting U953.32 = 20 and U953.33 = 20).

9.3.1 General functions

The Motion Control technology software includes the following general functions:

- ◆ **Linear axis** (with fixed stops and a maximum traversing range of 1000 m with a resolution of 1 μm); software limit switches are evaluated. A transfer carriage is an example of a linear axis:

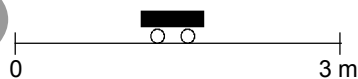


Fig. 9-2 Linear axis

- ◆ **Rotary axis** (rotating infinitely, without fixed stops, with specification of the direction or direction of the "shortest path"). A rotary table is an example of a rotary axis:

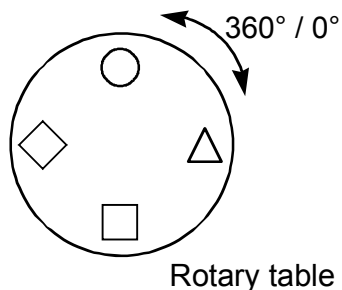


Fig. 9-3 Rotary axis

- ◆ **Roll feed** (infinitely rotating rotary axis with "cut-to-length" function). The figure shows the roll feed as used in a cutting device:

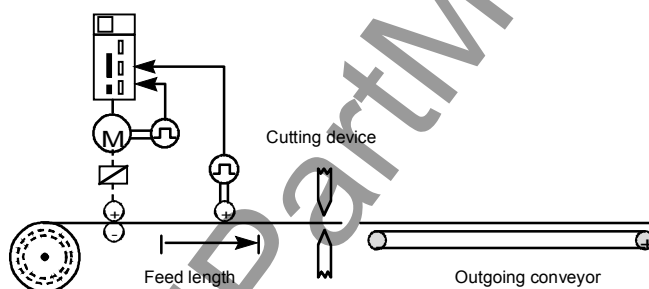


Fig. 9-4 Roll feed

- ◆ Either the **internal motor encoder** (resolver, optical encoder, absolute encoder, incremental encoder) or an external machine encoder (incremental or SSI absolute encoder) connected to the machine tool can be used as the position encoder [230..270].
- ◆ A **sophisticated precontrol** strategy is implemented in the Motion Control software: the position ramp-function generator adjusts the speed and accelerating torque continuously, bypassing the position controller, in order to achieve an optimum dynamic response and a negligible following error.
- ◆ The wear on the mechanical equipment is low, even if the full dynamic response is utilized, thanks to the position ramp-function generator with its flexible jerk limitation and acceleration adjustment.

9.3.2 Positioning

The MASTERDRIVES MC servo converter has a convenient integrated positioning control system with the following functions:

- ◆ **Setup:** Position-controlled axis travel in jog mode [819]
- ◆ **Homing:** The procedure used to zero the position measurement system when an incremental encoder is used (not usually required for absolute encoders) [821]
- ◆ **MDI:** Point-to-point positioning (**Manual Data Input**) [823]
 - Relative or absolute positioning (absolute or incremental dimensions)
 - Definition of an MDI block with position, velocity and acceleration
 - The MDI block can be defined directly from the machine control – e.g. via PROFIBUS-DP – or can be retrieved, using control commands, from a table of 10 fixed position setpoints stored on the MASTERDRIVES MC. The start command can be transmitted in the same PROFIBUS message as the MDI block, a feature that allows convenient, time-optimized control of the positioning operation from a mini PLC.
 - It is possible to change to another MDI block on-the-fly during traversing.
 - The start command (and the read-in enable for the roll feed) can be transmitted via digital inputs on the MASTERDRIVES MC or via a field bus.
- ◆ **Automatic mode:** [826...828]
 - Automatic execution of complete positioning programs
 - Single-step mode possible
 - Development of NC programs using a high-performance programming language in conformity with DIN 66025 (the standard of the German mechanical engineering industry)
 - Input of NC programs via an S7-300 (input via parameter interface and via the DriveMonitor service program under development)
 - Up to 20 programs with a total of 50 blocks (NC traversing commands) programmable
 - Program-driven output of switching functions (M functions)
 - Block change on-the-fly via digital input
 - Start and read-in enable also possible via digital input
 - Programmable zero offset, tool offset and backlash compensation
 - G function acceleration control
 - Set actual value on-the-fly
 - Start command, block change and read-in enable definable via field bus or digital inputs

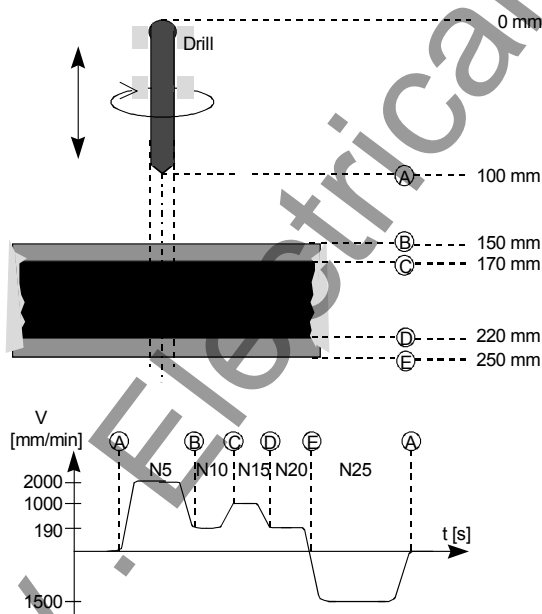
- Teach-in: allows the current position to be saved in an NC block in setup mode
 - Velocity, acceleration and time override
 - Collision monitoring via external input
 - Simulation mode for testing automatic programs without a motor, e.g. for recording the position setpoint curve with M function simulation
- ♦ **Roll feed [830]:**
- Automatic cut-to-length feature for presses, punching machines and cross-cutters in start/stop mode
 - Definable velocity/ acceleration profile of traversing curve. This achieves optimum cycle times with the lowest possible material wear and slippage.
 - Switchover possible between external machine encoder and motor encoder (at standstill)
 - The loop count (number of cut-to-length processes) can be programmed.

Applications for the positioning function

Positioning drives in the following fields are typical applications of MASTERDRIVES MC:

- ♦ Woodworking machines
- ♦ Roll feeds for presses
- ♦ Packaging machines
- ♦ Drive tasks in the glass, brick and tire industries and in general mechanical engineering applications

The following figure shows an example application for the automatic function on an automatic drilling machine in the woodworking industry:



NC-Program

N5 X150 F2000 G44 D1:	Block no. 5: move to position 150 mm at velocity 2000 mm/min, save tool offset (G44) in D1 (100 mm)
N10 X170 F190:	Block no. 10: move to position 170 mm at velocity 190 mm/min
N15 X220 F1000:	Block no. 15: move to position 220 mm at velocity 1000 mm/min
N20 X250 F190:	Block no. 20: move to position 250 mm at velocity 190 mm/min
N25 X 0 F1500 D0:	Block no. 25: return to initial position 0 and deselect tool offset (D0)

Fig. 9-5

Example of an automatic program

The figure shows a typical application for an NC program running automatically on the MASTERDRIVES MC. A chip board, which is laminated on both sides, is drilled in the following sequence of NC program steps:

- ◆ Travel A → B: The drill support travels in rapid traverse until just before the material and starts to reduce the feedrate. At exactly point B, the drill reaches the reduced feedrate for drilling through the laminated plastic.
- ◆ Travel B → C: Slow drilling of the laminate
- ◆ Travel C → D: The actual chip board is drilled at the normal feedrate.
- ◆ Travel D → E: The reduced feedrate is applied again for the lower layer of laminate.
- ◆ Travel E → A: The drill returns at increased velocity.

The NC program input on the MASTERDRIVES MC is also listed in the figure above.

9.3.3 Synchronization

General synchronization functions [831]

The following synchronization functions are included:

- ◆ Electronic shaft (angle-synchronous, long-term stable synchronization of several axes)
- ◆ Electronic gearbox (with sensitive transmission ratio adjustable via numerator and denominator; value range for numerator and denominator -32767 ... +32767)
- ◆ The transmission ratio can also be changed during operation. If necessary, the defined transmission ratio can be controlled via a free ramp-function generator [791] to prevent jumps.
- ◆ Electronic cam
 - "Table synchronization" with up to 400 interpolation points on the MASTERDRIVES MC. The 400 interpolation points can be distributed variably among one to eight tables. One table can be reloaded in the background while the other is running online. The interpolation between the points is linear.
 - The interpolation points do not have to be equidistant, but can be arranged closer to each other in critical zones and further away in linear areas.
 - Table change on-the-fly possible during operation
 - The table is scalable in the X and Y directions, and has an integrated gearbox

- ◆ The path/angle setpoint can be defined by a "real" machine axis (internal or external) or by a software "virtual" master.
- ◆ 2 interrupt-capable digital inputs for the detection of synchronization signals, e.g. printing indices

NOTE

The synchronism block should be called up in time slot T4 ($2953.33 = 4$). Calling the block in shorter times slices ($U953.33 < 4$) is not permissible.

**SIMOLINK as
synchronization
control backbone
[140...160]**

The serial SIMOLINK setpoint interface allows angle-synchronous connection of all drives involved in the angular synchronization. SIMOLINK is a high-speed fiber-optic ring which operates at 11 Mbaud and transmits the angle setpoints from drive to drive or from a host system to the drives. For example, SIMOLINK needs only 630 μ s to transfer 100 values of 32 bits each. Special SYNC messages enable quartz-accuracy, jitter-free synchronization of the sampling times of up to 200 connected converters. The result is highly dynamic, angle-synchronous drive operation. The master pulse generator is not normally required, since its function is reproduced by software and transmitted angle-synchronously via SIMOLINK (the principle of the "virtual master axis [831]"). Conventional operation with a real master, i.e. a master pulse generator, is naturally also possible [833].

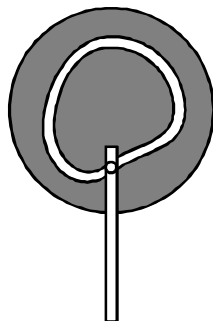
Thanks to SIMOLINK, the master drive function can be assigned to any drive or to a higher-level control system. This is particularly necessary for machines whose drives are removed from the drive train, e.g. for shaftless printing machines. The master drive function can also be performed by a drive that is removed temporarily from the drive system. SIMADYN D, SIMATIC FM458 or SICOMP SMP can be used as the host control system; SIMOLINK interfaces are available for these systems.

**Electronic gearbox
[835]**

The electronic gearbox is a simple substitute for all types of variable-speed gearbox and shafts. The exact gear factor is defined as a fraction with numerator and denominator (16 bits each). The electronic gearbox can be used with all encoders mounted on Siemens motors, including absolute encoders and encoders with SSI-standard protocols.

Electronic cam for reproducing mechanical contours [839]

The electronic cam enables angle-synchronous relative movements between a master and slave drive. It acts as a substitute for mechanical eccentric cams or cranks, as illustrated in the figure below:



Master	Slave
0 °	20 mm
5 °	100 mm
10 °	300 mm
⋮	⋮
360 °	20 mm

Fig. 9-6 Electronic cam ("table synchronization")

Up to 400 coordinate pairs describe the relative movement by means of table interpolation. These 400 interpolation points can be subdivided into one to eight curves; x and y-axis coordinates can be entered separately; the x values do not have to be equidistant. These parameter settings can naturally also be configured via PROFIBUS-DP, and the cam can therefore be modified in a matter of seconds.

Engaging/ disengaging cycle for product separation and grouping [834]

The engaging/disengaging cycle allows manipulated engaging and disengaging of angular synchronization, including the cam function, at an accurately defined coupling position for a period of one or more machine cycles. The ramp for the engaging/disengaging cycle can be defined as a path. The engaging/disengaging cycle can also be started via a digital input.

The disengaging cycle can be used in order to group products, for example if one product is missing from a continuous product stream. The disengaging cycle stops the drive (slave drive) at a parking position on request, and rejoins the master drive in angular synchronism after one or more machine cycles (product lengths).

The engaging cycle can be used, among other applications, to sort out reject products. The function operates according to the same principle as the disengaging cycle, although in this case the drive is coupled in angle synchronism with the master drive from its parking position for a period of one or more machine cycles, before returning to its exact parking position again.

The engaging/disengaging cycle can also be used in combination with the gearbox and the cam.

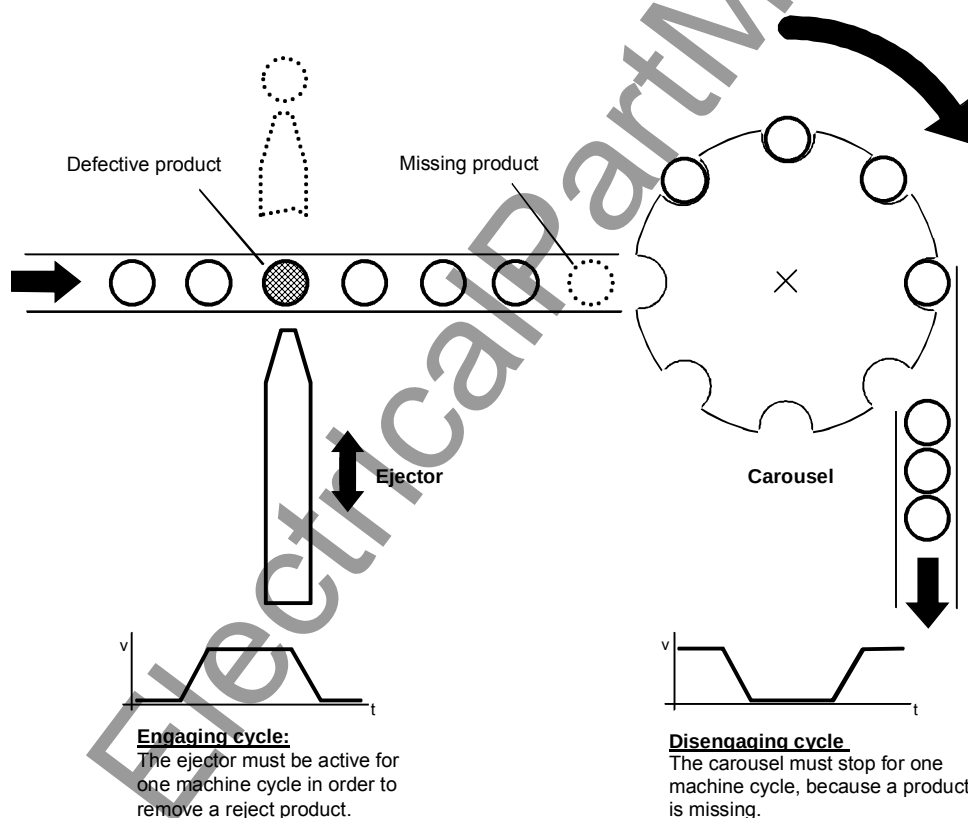


Fig. 9-7

Application example of engaging/disengaging cycle for removal of reject products in a packaging machine

Printing index controller [843]

The printing index controller (position correction) acts in combination with suitable reading devices to control the orientation of the master and slave drive in relation to each other. The synchronization signal is evaluated by a high-speed interrupt-capable digital input with a resolution of several μs . The velocity at which the orientation or correction movement is performed is adjustable.

An example of printing index evaluation is a packaging machine in which a continuous stream of products has to be packaged in film, such that the printed image on the packaging film always appears in the same position. By detecting the printing index on the film, the film expansion (or film contraction) can be measured and automatically corrected. Drift effects, which would be apparent without a printing index controller, are thus eliminated reliably.

The figure below illustrates the principle of printing index synchronization

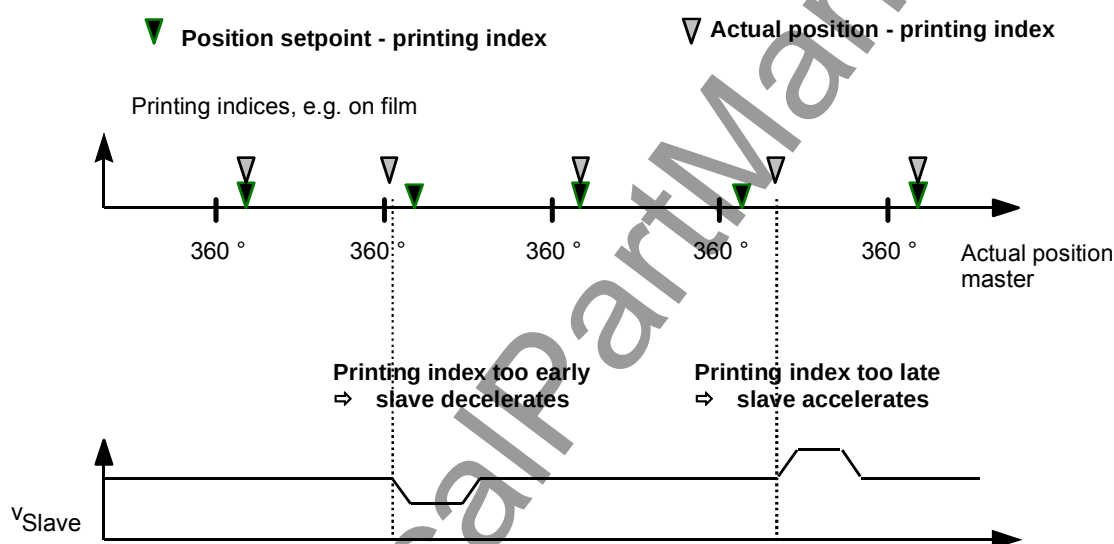


Fig. 9-8 Principle of printing index synchronization

Referencing "on the fly" [843]

The referencing "on the fly" function enables synchronization to a reference index (e.g. BERO proximity switch) to take place "on the fly" in synchronization mode.

It is no longer necessary to first approach the reference index in positioning mode and then change over to synchronization mode from standstill.

Synchronization to master value [841]

The "Synchronization to master value" function enables the position of the slave axis to be matched to the position of the master axis during synchronization. The speed and acceleration of the compensation movement required for this are adjustable.

Displacement angle setting [841]

An angle of displacement can now be conveniently set for synchronization in the following 3 alternative ways:

- ◆ Setting an absolute displacement angle via a selectable connector
- ◆ Setting a relative displacement angle via connectors or parameters which can either be supplemented in a positive or a negative direction, depending on the current zero position.
- ◆ Setting a relative displacement angle in inching mode with a selectable variable speed (similar to a motorized potentiometer).

These displacement angle settings can be made in any random size. Overflows of more than one slave axis revolution can be coped with. The displacement angle setting can be used, for example, to set up an index control system for printing machines.

Catch-up [837]

The catch-up function enables a drive to be uncoupled from an angular-locked synchronizing multi-motor system (e.g. shaftless printing press) and to be operated autonomously at its own speed setpoint ("isolated setpoint"). It also enables a drive to halt at a specified angular position.

The drive can catch up to the speed of the running machine from the halt position or from its current speed in autonomous operation. After the catch-up command has been set, the drive accelerates to the machine speed and can then be coupled back accurately into angular synchronization after reaching speed synchronization.

Applications for the synchronization function

The angular synchronization control system is a substitute for mechanical shafts, gearboxes and cams, e.g. in

- ◆ Shaftless printing presses
- ◆ Packaging and bottling machines
- ◆ Shuttle looms and other textiles machinery
- ◆ Gantry systems
- ◆ Conveyor systems

9.3.4 Technology functions already included in the standard software

Cam controller

A cam controller activates and deactivates digital outputs when certain positions, defined by parameter settings, are reached. This response is used to actuate external switching elements – such as pneumatic valves – at defined positions in a sequence of movements ("position cam"). Certain convenient cam controllers allow the operating time of the external switching elements to be controlled according to the velocity. Such systems are known as "time cams".

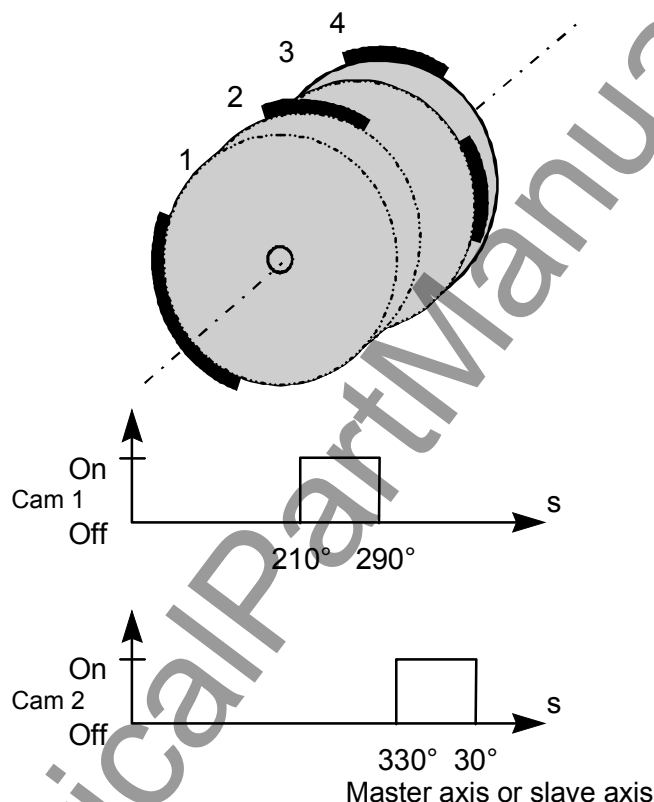


Fig. 9-9 Cam controller with MASTERDRIVES MC

The standard software of MASTERDRIVES MC contains free blocks for 2 simple cam controllers [745] and 1 expanded cam controller [745a]. Each has two position cams which can be supplied from separate input signals, e.g. a slave path setpoint and a master path setpoint. A total of 6 cams are therefore available with independently adjustable activation and deactivation positions, adjustable hysteresis for the switching instants, and a minimum time resolution of 500 μ s. If the input variable is a rotary axis (as in Fig. 9-9 above) and a cam passes the axis cycle jump (as cam 2 in Fig. 9-9 below), the axis cycle of the rotary axis (360° in Fig. 9-9) has to be entered in the corresponding cam controller (U155.2, U161.2, U436.6). If the axis cycle is equal to zero (factory setting), the cam controllers behave as previously. If the axis cycle is not equal to zero, all entered and calculated positions are mapped

within an axis cycle. To ensure that cam 1 does not overlap with itself, the following restriction must be observed:

The hysteresis must be smaller than half the difference between the axis cycle and the cam size. If this condition is not fulfilled, the output binector remains off.

The outputs of the cam controllers are binectors B480...B485, and these can be connected as desired, e.g. to digital outputs of the MASTERDRIVES unit for actuation of solenoid valves, etc.

Velocity-dependent switching time compensation and time cams are not provided, however it is often possible to implement these functions using the timers [780] in the free blocks. If you require an external high-speed cam controller with switching time compensation and additional time cams, you should use an external hardware cam controller, such as the SIMATIC S7 FM 352 module ("FM cam") or technology board T400.

9.3.5 Seamless Integration in SIMATIC Automation Solutions

The tried-and-tested standard function block packages DVA_S5 /3/ for SIMATIC S5 and Drive ES, SIMATIC /4/ for SIMATIC S7 provide convenient access to all process data and parameters of the MASTERDRIVES MC using the PROFIBUS-DP or USS protocols - not only for standard functions, but for all technology functions.

For example, a positioning NC block ("MDI block") can be defined and the movement started simultaneously in one PROFIBUS-DP message. The complete traversing process now runs without further intervention from the PLC. At the end of the movement, the axis returns a checkback message indicating that the target position has been reached. The same procedure is naturally also used with other field buses (CAN bus, USS etc.).

The following components are also available for the seamless integration of MASTERDRIVES MC in a SIMATIC S7-300 or S7-400 automation system using distributed technology (see Catalogs LS01 and /1/):

◆ **SIMATIC S7 "Motion Control Configuring Package" on CD-ROM (contained in /1/):**

Software for communication between the S7 user program and the technology via PROFIBUS-DP across a clear, easy-to-handle data interface for the following functions:

- Transfer of control/checkback signals to the technology
- Task interface for defining MDI and automatic NC blocks and programs, gear ratios, cam tables, etc.

◆ **Motion Control HMI Package for SIMATIC S7 (see /2/):**

Software for the application interface to operator panels OP25, OP27, OP37, TP27, etc. with standard screens for operating the positioning axes, including the following functions:

- MDI and automatic NC program input
- Input of machine data and cam tables
- Diagnostic screens with definition/display of control and checkback signals

You will find further information in the section entitled "Communication with the technology".

9.4 Brief description of the technology functions

9.4.1 Overview of the function diagrams

The function diagrams are an important work reference for configuring and commissioning the technology functions. The position sensing and control system and the F01 technology option are presented graphically in the following function diagrams:

- ◆ Function diagrams [230] ... [270]: Position encoder evaluation
- ◆ Function diagrams [330] and [335]: Position sensing, generation of the actual position
- ◆ Function diagram [340]: Position control
- ◆ Function diagrams [799] ... [802]: Overview of technology option F01 and the mode manager
- ◆ Function diagrams [804] ... [818]: Input/output signals of the technology
- ◆ Function diagrams [819] ... [830]: Positioning modes of the technology
- ◆ Function diagrams [831] ... [843]: Synchronization functions of the technology
- ◆ Function diagram [850]: Enabling of technology option F01 per PIN number

You will find a brief description of these functions here in section 9.4.



You will find detailed reference information on all positioning and synchronization functions in the "Motion Control for MASTERDRIVES MC and SIMATIC M7" manual /1/.

The equipment is supplied with technology option F01 not active. In order to use it, you must

- ◆ Connect the technology option to the basic unit using BICO technology and
- ◆ Nest it in the desired sampling times.

See the section entitled "Technology overview and mode manager" and [802].

The following sections provide you with a brief introduction to the technology with reference to the relevant pages of the function diagram.

9.4.2 Integrating the technology into the basic unit [801]

On sheet [801] of the function diagram, you can see how to connect technology option F01 to the basic unit functions below using BICO technology:

- ◆ Position sensing (for either motor encoder or external machine encoder)
- ◆ Position and speed controller
- ◆ Free blocks (the cam controller [745] , ramp-function generator [790 + 791] and logic blocks [765...780] are of particular relevance)
- ◆ Communication interfaces (USS, PROFIBUS, etc.)
- ◆ SIMOLINK drive coupling
- ◆ Hardware terminals (digital/analog I/O terminals of the MASTERDRIVES)

The main connections are already preset at the factory (indicated by "(WE)" in the diagram). Further information on the connections that still need to be made is provided in [815], [817], [836] and in the section entitled "Commissioning the technology".

Incidentally

The centralized technology functions for positioning and synchronization in a host control system – such as SIMATIC FM458 or SIMADYN D – are integrated via the same connection points as technology option F01.

9.4.3 General information on position encoder evaluation [230] ... [270]

You will find an overview of the position encoders that can be evaluated in MASTERDRIVES MC, together with details of their resolution and accuracy in the section entitled "Configuration".

The following position encoders can be evaluated in MASTERDRIVES MC (see also [801.1]):

One of the following encoders can be used as a **motor encoder** for position sensing by inserting a sensor board in slot C:

- ◆ Resolver [230]
 - Sensor board: SBR1/SBR2 (with/without pulse encoder simulation)
- ◆ Optical sin/cos encoder [240], e.g. ERN 1387
 - Sensor board: SBM2
- ◆ Pulse encoder [250] (for asynchronous motors; pulse encoder in V1.2 not yet released as motor encoder for positioning and synchronization)
 - Sensor board: SBP
- ◆ Multiturn absolute encoder [260], e.g. EQN 1325, EQI 1325
 - Sensor board: SBM2

The following encoders can be evaluated as **external machine encoders** for position sensing:

- ◆ Pulse encoder [255]
 - Sensor board: SBP
- ◆ Multiturn encoder [270] , e.g. encoder with EnDat or SSI protocol
 - Sensor board: SBM2
(with analog fine resolution for EQN)
- ◆ Optical sin/cos encoder
 - Sensor board: SBM2

The sensor board for the external machine encoder can be plugged into any slot except slot C. When a supplementary technology module T100, T300 or T400 is used, the evaluator module for external machine encoder must have been plugged into Slot A.

All encoder evaluation systems generate a B070 status signal (or B071 for external encoders) which returns a "1" signal whenever the measured data acquisition system is operating error-free.

NOTE

When a pulse encoder is employed as the motor encoder, the referencing modes "To Left of BERO" and "To Right of BERO" cannot be used since this type of encoder does not evaluate the zero pulse.

NOTE

If the external encoder is required for an application, the block "Position acquisition external encoder" (function diagram 335) must be slotted either in the same time slot as or a faster time slot than the technology function.

Overview

The positional information is required, in addition to the speed, for the technology functions. The MASTERDRIVES MC enables direct position sensing by the motor encoder, and so a further externally mounted encoder is not required for position control. An additional external encoder need only be used for position sensing if this is necessitated by the application technology. The encoder types can be subdivided into incremental and absolute encoders.

Incremental encoders

Incremental encoders (pulse encoders) only detect relative changes in position. The encoder system must be referenced, in order to allow absolute positioning. This is performed using a proximity switch (BERO) with a known mechanical position.

Absolute encoders

Absolute encoders can be subdivided into two groups:

- ◆ **Singleturn encoders** (two-pole resolvers, optical sin/cos encoders) indicate the absolute position within one revolution. If you need to perform absolute positioning movements over several revolutions with a singleturn encoder (this is normally the case), the system must be referenced as with an incremental encoder.
- ◆ **Multiturn encoders** detect the position over a defined range (e.g. 4096 revolutions) in addition to the position within one revolution, and they store this value when the power supply is switched off. Referencing is therefore not necessary with a multiturn encoder.

The following equipment variants of the MASTERDRIVES MC with encoder evaluation boards ("Sensor boards") are possible. A maximum of 2 encoders can be evaluated simultaneously in these configurations:

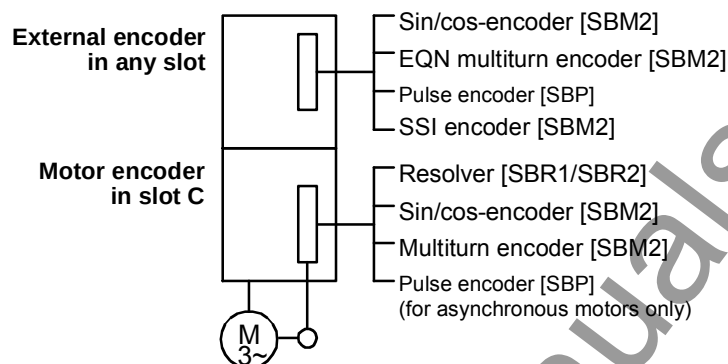


Fig. 9-10 Overview of suitable sensor boards

9.4.4 Resolver evaluation [230]

Principle

The resolver operates with an analog inductive measuring system. The resolution of the analog signals is 4096 increments per revolution. The positioning accuracy of the resolver that can be achieved under practical conditions is limited to approximately 1000 steps per motor revolution.

The two-pole resolver measures the absolute position of the rotor from 0° to 360°. In multi-pole resolvers, the measured position is not assigned to a unique mechanical rotor position.

Sensor boards SBR1 and SBR2 are available (with/without pulse encoder simulation) for the evaluation of resolver signals [230].

Cable length

Cable lengths of up to 150 m can be used for the two-pole resolver. Attention should be paid to proper EMC installation (screening, physical separation of the power cables). Please also keep in mind that – irrespective of the type of converter, pulse frequency and type of power cable between motor and converter – the permissible power cable length can be less than 150 m.

Select motor encoder P130

This parameter is automatically initialized with the value 1 (= two-pole resolver) when sensor board SBR1 or SBR2 is used. The setting must be changed for multi-pole resolvers.

Select Motor Encoder		
Par.	Value	Meaning
P130	1	2-pole resolver as motor encoder
P130	2	Resolver with motor pole pair number as motor encoder

Angle offset P132

The absolute position of the resolver must match the mechanical position of the flow axis. The alignment of the resolver is performed at the factory and must not be modified. If you use a third-party motor with a different alignment specification to a SIEMENS motor, the displacement angle can be corrected with parameter P132.

Pulse encoder simulation P134

If you use the SBR2 board, pulse encoder simulation is available on the front connector. The number of pulses/revolution is configurable. Depending on the parameter setting, the simulation produces 512 or 1024 pulses plus one zero pulse per revolution. This applies to the two-pole resolver. Resolvers with larger numbers of poles output a correspondingly higher number of pulses per revolution. The signals are output as differential signals in pulse tracks displaced by 90° (RS422) and are **not** potentially isolated.

Pulse Encoder Simulation		
Par.	Value	Meaning
P134	0	512 pulses per revolution
P134	1	1024 pulses per revolution

Standard encoder

The two-pole resolver is the standard encoder.

NOTE

If a multi-pole resolver has to be referenced, connector KK96 has to be used according to Chapter 9.4.8 "Position sensing system for motor encoder [330]".

9.4.5 Optical sin/cos encoder [240]**Principle**

The resolution with an ERN1387 optical sin/cos encoder is represented by the positional information of $2^{24} = 16\,777\,216$ graduations per encoder revolution: for each 2048 sine and cosine periods per revolution this yields a "rough digital resolution" of 8192 steps per motor revolution after "pulse quadrupling" (evaluation of the zero crossings). Each quarter period is resolved again into 2048 steps through the fine analog amplitude evaluation of the sine/cosine signals.

The relative accuracy (repeatability) that can be achieved under practical conditions is approximately 4 000 000 steps per revolution. The system accomplishes high-precision position sensing with an absolute accuracy of approximately 100 000...1 000 000 steps per encoder revolution.

The sin/cos encoder uses a special sine/cosine track, each with one period per revolution, to return the absolute position of the rotor from 0° to 360°. Initial rough acquisition of the rotor position is undertaken when the power is switched on, when moving out of the drive position (P60 = 5) or when fault F051 is acknowledged. This initial value is corrected the first time the zero pulse is crossed, after which the full precision of the encoder is available to the user.

The signals of the optical sin/cos encoder are evaluated on an SBM or SBM2 sensor board [240]. In addition to the signal processing circuits, the sensor board is equipped with the encoder power supply. The SBM2 sensor board with extended functionality has meanwhile taken the place of the SBM sensor board.

Encoder power supply

The SBM board can provide both 5 V and 15 V as the encoder power supply. The correct supply voltage for the encoder must be set on the board. The encoder might sustain damage if the supply is incorrectly set. The standard ERN 1387 encoder operates on 5 V.

SBM board

The power supply is set using two hook switches on the SBM sensor board. An incorrect setting can result in damage to the encoder.

Both switches open → 5 V encoder supply

Both switches closed → 15 V encoder supply

SBM2 board

The encoder supply voltage is set directly in volts via parameter P145 on the SBM2 board. The value in Index 1 defines the supply voltage for the motor encoder, while that in Index 2 defines the supply voltage for the external encoder. The maximum supply voltage for Compact units is 15 V and for Compact Plus units 24 V. Setting examples:

Encoder supply		
Par.	Value	
P145	5	5 V supply voltage for encoder
P145	15	15 V supply voltage for encoder

Cable length

The maximum cable length for the sin/cos encoder is 100 m.

NOTE

An ERN1387 encoder as a motor encoder requires a 6FX_002-2AC31-____ connecting cable.

CAUTION

The encoder cable must not be pulled out or inserted in a live state as otherwise the encoder may be destroyed!

Select motor encoder P130

The automatic board identification system automatically initializes the parameter for the sin/cos encoder.

Select Motor Encoder		
Par.	Value	
P130	3	Sin/cos encoder as motor encoder

Number of lines P136

The resolution of the encoder must be stored in parameter P136. The resolution is entered in increments per revolution. The input is graduated as 2P136.

Standard encoder

The ERN 1387 sin/cos encoder manufactured by Heidenhain is used as the standard encoder. The following parameters are initialized for this encoder:

Number of Lines for Sin/Cos Encoder ERN1387				
Par.	Value	Signal Periods/ Revolution	Incr./ Revolution	
P136	11	$2^{11} = 2048$	8192	Encoder resolution

For induction machines 1PH4, 1PH7(=1PA6) and 1PL6, the ERN1381 type of encoder made by the Heidenhain company is frequently used. The ERN1381 does not have any C/D tracks for detecting the initial position. P130 = 7 must be used for this type of encoder.

**Encoder selection
P130**

Par.	Value	
P130	7	Encoder without C/D track

The absolute initial position is not set in the case of encoders without C/D track. These encoders can be used with induction machines only. The position is corrected by a zero pulse where applicable.

9.4.6 Multiturn encoder evaluation [260, 270]**CAUTION**

The encoder cable must not be pulled out or inserted in a live state as otherwise the encoder may be destroyed!

Principle

The multiturn encoder is an absolute encoder. It detects the number of revolutions, in addition to the rotor position from 0° to 360°. On initialization, the start position is transmitted to the converter using a serial protocol. The revolution information is always stored on the encoder, i.e. even after a power failure, wire break or component replacement. Referencing is therefore not required.

In addition to the signal processing circuits, the sensor board is equipped with the encoder power supply.

**Encoder supply
voltage**

The supply voltage for the encoder is set by the same method used for the sin/cos encoder.

**Cable length
P149.01, P149.07**

The serial protocol of the coded rotary encoder is implemented as a synchronous protocol. Communication takes place according to the master (converter)/slave principle. The cable length is limited by the transmission times from converter to encoder and back.

Possible baud rate depending on the cable length:

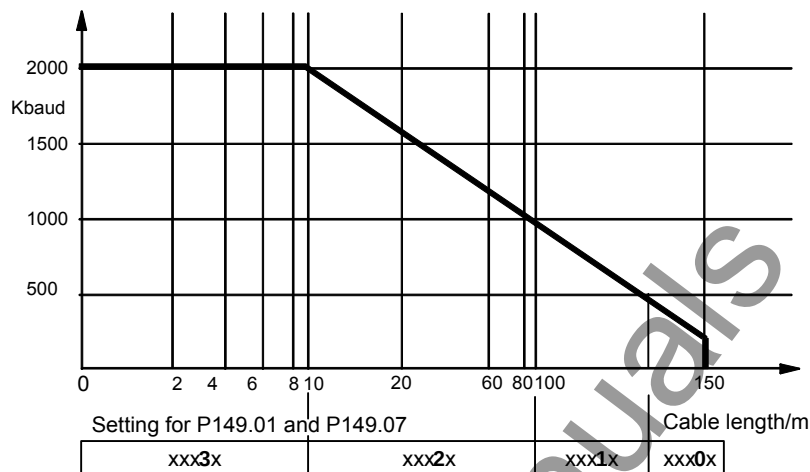


Fig. 9-11 Possible baud rate depending on the cable length

NOTE

The EQN1325, ECN1313 and EQI1325 encoders require a 6FX_002-2EQ10-____ connecting cable.

NOTE

Not all measuring system manufacturers support baud rates up to 2 Mbaud. The serial protocol for the motor encoder is only required for initialization and redundancy monitoring. These functions are non-time-critical. A baud rate of 100 kHz (standard setting) is recommended for immunity to interference.

Encoder types

Only encoders which generate incremental tracks, in addition to transmitting the position using the serial protocol, are suitable as motor encoders. The sensor board supports the serial **SSI** and **EnDat** protocols.

Recommended coded rotary encoders:

Coded Rotary Encoder				
Name	Res./ Rev.	Rev.	Protocol	
EQN 1325	8192	4096	EnDat	Multiturn Standard
ECN1 313	8192	-	EnDat	Singleturn
EQI 1325	256	4096	EnDat	Multiturn

Select motor encoder P130

The sensor boards for sin/cos encoders and multiturn encoders are identical. For this reason, the encoder type must be entered specifically in spite of the automatic board identification (the default setting is sin/cos encoder).

Select Motor Encoder		
	Value	
P130	4	Coded rotary encoder as motor encoder

Default setting P147.1

Parameter **P147.1** initializes the necessary settings. If one of the listed standard encoders is used, no further settings are necessary.

Select Multiturn				
P147.1	Name	Res./ Rev.	Rev.	
1	EQN1325	8192	4096	
2	ECN1313	8192	-	Singleturn
3	SSI 25	8192	4096	
4	SSI 21	8192	256	
5	SSI 13	8192	-	Singleturn
6	EnDat	auto	auto	
7	EQI1325	128	4096	
8	EQN1125	8192	4096	
9	ECN1113	8192	-	Singleturn
10	ROQ 424 SSI	4096	4096	Multiturn

Number of lines P148

The resolution of the encoder must be stored in parameter P148. The signal periods per revolution are set for Heidenhain encoders with EnDat (see the nameplate). The measuring steps per revolution are specified for SSI encoders. Consequently, you need a different setting for SSI and EnDat encoders with the same resolution. The parameter is input as powers of 2.

This parameter is initialized when P147 ≠ 0.

Example:

Number of Lines for Multiturn EnDat EQN1325				
	Setting		Resolution/ Revolution	
P148.1	11	$2^{11} = 2048$	8192	Signal periods/ revolution EnDat encoder
P148.2	12	$2^{12} = 4096$	-	Number of revolutions that can be distinguished
Number of Lines for Multiturn SSI				
	Setting		Resolution/ Revolution	
P148.1	13	$2^{13} = 8192$	8192	Measuring steps/ revolution SSI encoder
P148.2	12	$2^{12} = 4096$	-	Number of revolutions that can be distinguished
P148.7	3	$2^3 = 8$		Ratio between signal periods and measuring steps for SSI motor encoder
P148.8	3	$2^3 = 8$		Ratio between signal periods and measuring steps for SSI technology encoder

Configuration of protocol P149

The configuration of the serial protocol is stored in parameter P149. It is possible to use additional functions with EnDat protocols, e.g. for saving a zero offset (only permissible if an EQN1325 encoder is used) or storing customer parameters on the encoder. It is also possible to configure SSI protocols that differ from the standard configuration, e.g. with parity bit or binary coding.

This parameter is initialized when P147 ≠ 0.

Index 1 to 6 are applicable for the motor encoder, index 7 to 12 for the external machine encoder.

P149.1				SSI/EnDat Default Setting
T	H	Z	E	Thousands, Hundreds, Tens, Ones
X	X	X	0	SSI Protocol
X	X	X	1	EnDat Protocol
X	X	0	X	Baud rate 100 kHz
X	X	1	X	Baud rate 500 kHz
X	X	2	X	Baud rate 1000 kHz
X	X	3	X	Baud rate 2000 kHz
X	0	X	X	Serial protocol for initialization only
X	1	X	X	Serial protocol corrects pulse counter
0	X	X	X	Rotary encoder
1	X	X	X	Linear scale Disabled for motor encoders

NOTE

Parameter P149.1

Motor encoder, hundreds digit = 0:

Without the incremental position sensing being checked by the serial protocol

Motor encoder, hundreds digit = 1:

With the incremental position sensing being checked by the serial protocol

Parameter P149.7

External machine encoder, hundreds digit = 0:

Encoder without incremental signals, position sensing by the serial protocol

External machine encoder, hundreds digit = 1:

With the incremental position sensing being checked by the serial protocol

P149.2				EnDat Configuration
T	H	Z	E	Thousands, Hundreds, Tens, Ones
X	X	Z	Z	Number of data bits (EnDat) (e.g. 25 for EQN 1325)
X	0	X	X	Read measured values (EnDat)
X	3	0	X	Write parameters into encoder EEPROM EQN1325 (EnDat) (address in P149.4 and P149.5 ; value in P149.6)
X	4	X	X	Read parameters from encoder EEPROM EQN1325 (EnDat) (address in P149.4 and P149.5 ; value in P149.6)
X	A	X	X	Self-startup (EnDat) (protocol length; read encoder type and number of lines from encoder EEPROM and assign P148 ; P149 accordingly => only if EnDat protocol selected in P149.1 !)
X	B	X	X	Save zero offset on encoder (EnDat) (save the zero offset from P146.1 in the encoder EEPROM and clear P146.1)

P149.3				SSI Configuration
T	H	Z	E	Thousands, Hundreds, Tens, Ones
X	X	X	Z	Number of non-significant leading zero bits in the protocol (SSI)
X	X	0	X	Data of encoder in binary format (SSI)
X	X	1	X	Gray-coded encoder data (SSI)
X	0	X	X	No alarm bit (SSI)
X	Z	X	X	Position of alarm bit after last protocol bit (SSI)
0	X	X	X	No parity bit (SSI)
1	X	X	X	Parity check (last protocol bit) (SSI)

NOTE

In addition to the serial protocol, the motor encoder must also have a 1 Vpp output, since the motor controller requires the motor position in real time. The serial protocol can only achieve very low sampling rates and is thus not suitable at present for motor control. The standard encoder is the Heidenhain EQN1325 with the EnDat protocol.

P149.4				EnDat MRS-Code (Memory Range Select)
T	H	Z	E	Thousands, Hundreds, Tens, Ones
Z	Z	Z	Z	EnDat Memory Range Select - address selection for memory accesses to the encoder EEPROM in conformity with the EnDat specification (hexadecimal)
P149.5				EnDat Address
T	H	Z	E	Thousands, Hundreds, Tens, Ones
Z	Z	Z	Z	EnDat address in specified memory range in conformity with EnDat specification (hexadecimal)
P149.6				EnDat Data
E	H	Z	E	Thousands, Hundreds, Tens, Ones
Z	Z	Z	Z	EnDat data at address defined in P149.4 and P149.5, if read or write data was selected in P149.2 (hexadecimal) in conformity with EnDat specification

Encoder monitoring

The pulse inhibits are compared with the serial protocol of the encoder with P149.1 = x1xx and corrected if necessary. In the event of frequent deviations a fault message is tripped.

The monitoring time slot can be set in U950.19.

Parameter of Measuring System Manufacturer	EnDat Addresses (Selection in Conformity with EnDat Specification V2.0)			
	Linear	Rot.	MRS-Code	Add.
Operating state			B9	0 - 3
Masks			A1	4 - 7
Version of EnDat Interface			A1	8
Memory allocation for OEM parameters			A1	9 - A
Memory allocation for offsets			A1	B - C
Number of cycles for transmission of position			A1	D
Measuring system type			A1	E
Signal period or signal periods/rev.	nm		A1	F
Signal period or signal periods/rev.	nm		A3	0
Distinguishable revolutions			A3	1
(Basic) distance of reference marker	mm		A3	2
Position of first reference marker	mm		A3	3
Measuring step/measuring steps/rev. for protocol	nm		A3	4 - 6
Zero offset of measuring system manufacturer	per		A3	6 - 7
Identification number			A3	8 - A

	EnDat Addresses (Selection in Conformity with EnDat Specification V2.0)			
Parameter of Measuring System Manufacturer	Linear	Rot.	MRS- Code	Add.
Serial number			A3	B - D
Direction of rotation			A3	E
Startup diagnostics			A3	F
Maximum velocity/speed	m/min	min ⁻¹	A5	0
Accuracy in range I	LSB	LSB	A5	1
Accuracy in range II	LSB	LSB		2
Alarm support			A5	3
Warning support			A5	4
CHECKSUM			A5	0F
Operating Parameters	Linear	Rot.	MRS- Code	Add.
Zero offset in signal periods			A7	0 - 1
OEM Parameters	Linear	Rot.	MRS- Code	Add.
User-assignable			A9. AD	0 - F
Customer Parameters	Linear	Rot.	MRS- Code	Add.
User-assignable			AF	0 - F
Offset Values, Not Yet Defined	Linear	Rot.	MRS- Code	
			B1..B7	0 - F

Abbreviations:

add. = address

linear = linear scale

rev. = revolution

rot. = rotary encoder

NOTE

The zero offset for the motor encoder must **only** be displaced using parameter **P149.2**! Otherwise serious damage to the motor control system can result!

**Zero offset for
encoder P146.1**

The zero offset for the motor encoder can be entered in parameter **P146.1**. The zero offset is entered here in revolutions, since the rotor position is not otherwise allowed to be displaced in case of damage to the control system.

P146.1				Zero Offset
T	H	Z	E	Thousands, Hundreds, Tens, Ones
Z	Z	Z	Z	Zero offset in revolutions (decimal)

Setting the zero offset and saving to EEPROM

NOTE

In certain applications, it is necessary to save the zero offset directly on the encoder (in order to modify the position overflow).

It is **only** permissible to store the zero offset in the encoder EEPROM if the EQN1325 is used.

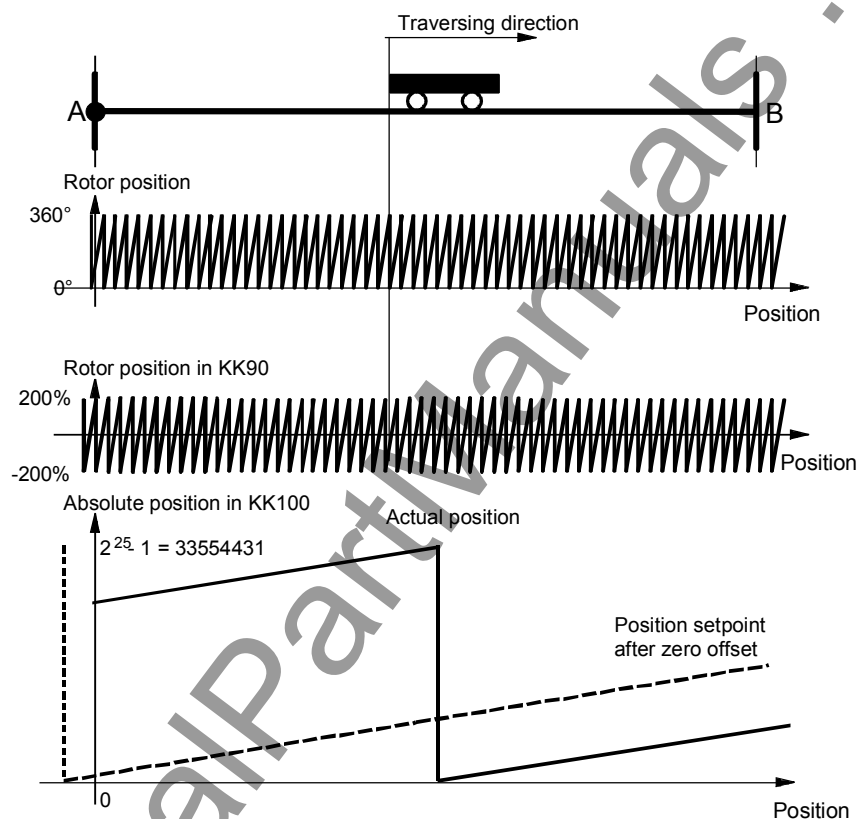


Fig. 9-12

Example for setting the offset on EQN encoders:

Encoder:	EQN1325, pulse number 2048
Parameter setting:	Standard (P147.1 = 1)
KK100 at point A is	27962026, EnDat protocol
Setpoint at point A	10 revolutions

How many increments per revolution are generated by the encoder?

Encoder EQN1325 generates 2048 signal periods per revolution. The maximum resolution is calculated with quadruple evaluation for position sensing → this yields 2 to the power of (number of lines + 2) increments per revolution.

$$2^{11+2} \text{ inc / rev} = 8192 \text{ inc / rev}$$

By how many increments must the absolute position be corrected?

$$\Delta = 27962026 \text{ inc} - 10 \text{ rev} \times 8192 \frac{\text{inc}}{\text{rev}} = 27880106 \text{ inc}$$

How many revolutions does that represent and what is the remainder?

$$\Delta_{\text{rev}} = \frac{27880106}{8192} \text{ rev} = 3403 \text{ rev}$$

$$\Delta_{\text{rem}} = (27880106 \text{ inc} - 8192 \frac{\text{inc}}{\text{rev}} \times \Delta_{\text{rev}}) = 2730 \text{ inc}$$

Set Zero Offset	
Revolutions P146.1	-3403

The setting is added on allowing for the overflows of the representable value range. The value range is between "zero" and the maximum encoder resolution even after a zero offset is set.

When **EnDat** encoders are used, the zero offset can be transferred from parameter **P146.1** into the encoder EEPROM.

Save Zero Offset in Encoder EEPROM		
Par.	Setting	Description
P60	5	Switch to drive settings
P149.2	B25	Save zero offset on encoder
P60	1	Return to ready for operation

NOTE

The offset setting in **P146.1** is cleared and stored on the encoder. It is important that the correct number of lines is specified in **P148.1** → if you do not use a default encoder, we recommend that you call up the EnDat self-startup function before saving the zero offset (**P149.2** xAxx and **P149.1** xxx1)

WARNING



If you store an offset in the encoder while an incorrect number of pulses per revolution is being entered in P148.1 the motor encoder will become incorrectly oriented and the motor may overspeed.

The fine offset within one revolution can be set with parameter P184 [330.7]. If the technology is used, the fine offset must be defined in **machine data MD10** [815.4].

9.4.7 Pulse encoder evaluation [250, 255]

Principle

The pulse encoder generates two pulse tracks displaced by 90° and one zero pulse per revolution. Because of the operating principle of the pulse encoder, it only reproduces changes in position. In order to determine the absolute position from 0° to 360°, it is necessary to reference the system (e.g. by crossing the zero pulse). These characteristics mean that the pulse encoder can only be configured for asynchronous machines.

Evaluation of the pulse encoder is handled by a sensor board (SBP) [250, 255].

Cable length

The permissible cable length depends on the encoder selected. Various charts are produced according to the interface. Bipolar encoders allow the longest cables to be used. Unipolar encoders restrict the length of cable which may be used. In the case of HTL encoders, the maximum output current of the encoder has a major impact on the transmission distance that can be achieved. The higher the maximum output current, the greater the distance possible (the encoder must reverse the cable capacitance charge on every pulse).

With the **SBP** sensor board, the terminating resistors can also be used for HTL signals (this is a "dynamic cable terminator" which keeps power loss to a minimum).

In the chart below, the maximum permissible cable lengths for TTL/RS422 encoders are plotted against the pulse frequency:

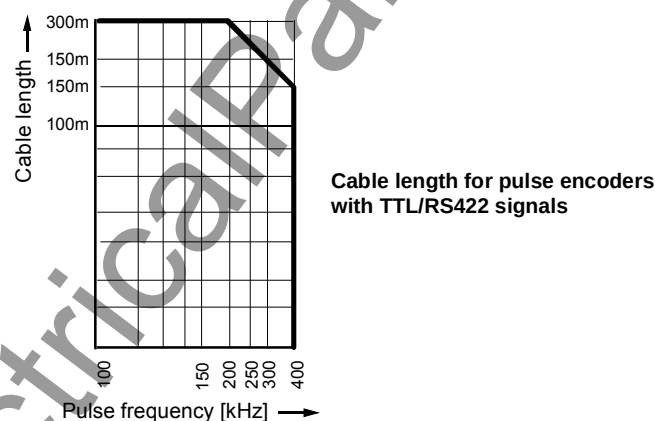


Fig. 9-13 Maximum cable lengths for pulse encoders with TTL/RS422 signals

Select motor encoder P130

Slot C is the preferred location for installation of a pulse encoder used as a motor encoder. If the encoder is not installed in slot C, the parameter should be configured as shown below:

Select Motor Encoder		
Par.	Value	
P130	5	Pulse encoder in slot C as motor encoder
P130	6	Pulse encoder not in slot C as motor encoder

Number of lines P151

The resolution of the encoder is determined by its number of lines. The number of lines per revolution is entered in parameter **P151**. This parameter appears on the product nameplate and in the corresponding data sheet.

Example for a pulse encoder with 2048 pulses per revolution:

Select Motor Encoder			
Par.	Number of Lines	Increments/ Revolution	
P151.1	2048	8192	Pulses per revolution for motor encoder

Configuration P150

The signal level of the pulse encoder can be modified according to the following table.

P150				Pulse Encoder Configuration		
Th	H	T	O	Low Level	High Level	Meaning
			↓			Signal Level, A/B Track
X	X	X	0	< 3 V	> 8 V	HTL unipolar (inverted inputs to ground)
X	X	X	1	< 1 V	> 4 V	TTL (inverted inputs to ground)
X	X	X	2	< -3 V	> 3 V	HTL differential signal
X	X	X	3	< -0,2 V	> 0,2 V	TTL/RS422 differential
X	X	↓	X			Signal Level, Zero Track
X	X	0	X	< 3 V	> 8 V	HTL unipolar (inverted inputs to ground)
X	X	1	X	< 1 V	> 4 V	TTL (inverted inputs to ground)
X	X	2	X	< -3 V	> 3 V	HTL differential signal
X	X	3	X	< -0,2 V	> 0,2 V	TTL/RS422 differential
X	0	X	X			5 V supply for encoder
X	1	X	X			15 V supply for encoder

NOTE

Incorrect setting of the power supply parameters can result in damage to the encoder.

Four switches are installed on the SBP sensor board. Switches 1 to 3 activate the bus terminating resistors (the sensor board is supplied with the switches closed), while switch 4 deactivates the power supply in the closed state (the sensor board is supplied with the switch open).

9.4.8 Position sensing system for motor encoder [330]

Overview

The position sensing system for the motor encoder is shown in [330]. The motor encoder generates a rotor position signal KK090 "Theta(mech)" [500] with a resolution of 2^{32} increments per encoder revolution, from which the position sensing system [330] generates the actual position value KK120.

In the factory setting ($32 - 12 = 20$ shift steps), the shift division block [330.4] generates a raw positional value with 4096 increments per encoder revolution, which is suitable for most applications using a resolver. The shift division is used to ensure optimum resolution of the actual position value and to prevent the numeric range of 32 bits from being exceeded during extremely long traversing movements (or the range from -999 999 999 to +999 999 999 when using technology option F01; see [815.4]).

After shift division, the actual position value is normalized by applying the actual value weighting factor AVWF such that an increment at the actual position output KK120 matches the length unit LU in which the target positions are to be specified. You will find specifications for setting the AVWF below and in "Defining the actual value weighting factor" in the section entitled "Commissioning the technology".

The position sensing system includes the following additional functions:

- ◆ Position correction, which can be used to trap overflows on rotary axes and roll feeds (is activated accordingly by the technology [815.5 and 836.8])
- ◆ Basic control system for homing with incremental encoders (resolvers, sin/cos encoders, pulse encoders)
- ◆ Position memory for storing the present actual position if one of the two interrupt-capable digital inputs on the converter terminal strip is actuated by a signal edge (terminals X101.6 and X101.7). Printing index signals from optical sensors or other synchronization signals can be connected to the digital inputs. This measured position is processed subsequently on the technology [815 and 836].

Principle

Resolvers and sin/cos encoders register the absolute rotor position within one revolution from 0° to 360°. The number of motor revolutions is counted in addition to this rotor position, in order to measure the position. If a multiturn encoder is used, the number of revolutions is determined during initialization. The sum of the motor revolutions and the rotor position determines the overall position. The transition from encoder increments to a physical unit such as μm or degrees is performed by applying the actual value weighting factor (AVWF). The physical unit of position is referred to below as LU (Length Unit).

The position sensing system operates with a data width of 32 bits and thus has a value range of:

	Minimum Value	Maximum Value
	-2^{31}	$+2^{31}-1$
Increments * AVWF	-2.147.483.648	2.147.483.647
Length units [LU]	-2.147.483.648	2.147.483.647
Example: 1 LU = 1 μm	-2.147.483.648 μm	2.147.483.647 μm

Please note that the value range is limited to $\pm 999\,999\,999$ when technology option F01 is used [815.4].

Enable position sensing/encoder type P183

A value of 0 in the ones digit of parameter P183 disables position sensing. The function of the position sensing system is inhibited and all output connectors have the value zero.

P183				Meaning
Th	H	T	O	Thousands, Hundreds, Tens, Ones
X	X	X	0	Position sensing disabled → no actual position calculation
X	X	X	1	Enable position sensing with resolver, pulse encoder or encoder
X	X	X	2	Enable position sensing with multiturn encoder

x = not relevant for enabling position sensing.

NOTE

With a two-pole resolver or encoder, the output of the position sensing system is set to the current rotor position when the unit is powered up. The position sensing system thus detects the absolute position within one revolution of the motor.

If a multiturn encoder is used, the number of revolutions is also detected.

Position resolution P171

Parameter P171 defines the resolution of the encoder system used to generate the overall position. The value should not be greater than the practical maximum resolution of the encoder. It is important that the complete traversing range be capable of representation in a 32-bit double word. If this is not the case, the resolution must be reduced by shift division.

The table below provides an overview of practical encoder resolutions:

Encoder system	Practical max. resolution [increments/revolution]
Resolver	$2^{12} = 4096$
Encoder	$2^{24} = 16777216$
Multiturn encoder	$2^{20} = 1048576$

The resolution in P171 must be selected such that the positioning range can be represented in a 32-bit double word.

The default of 4096 increments/revolution is adequate for most positioning tasks.

Actual value weighting factor P169/P170 P180.01 / 02

The actual value weighting factor (AVWF) is used to transpose the encoder increments into a physical unit. The unit can be defined freely, and is referred to as the LU (Length Unit). LU is the length unit in which the user prefers to specify his target positions. The weighting factor specifies the traversing distance in the "number of length units LU" corresponding to one position increment (after shift division) – including all gear ratios, roll feed diameter, etc.

The preferred reference for the length unit LU in association with positioning tasks is μm for linear axes and 0.001 degrees for rotary axes.

Recommended Position Normalization for Positioning Axes		
Recommended for linear axis: 1 LU = $1 \mu\text{m}$	AVWF =	traversing distance in μm per increment [LU/inc]
Recommended for rotary axis: 1 LU = 0.001°	AVWF =	traversing distance in 1/1000ths of one degree per increment [LU/inc]

The actual value weighting factor (AVWF) can be input by 2 different methods:

- Directly as a decimal number with 3 integer places and 8 decimal places
 - As a fraction with a 20 bit numerator and a 20 bit denominator
- Variant (b) will need to be used if the AVWF cannot be represented by 8 decimal places and where cumulative errors have to be avoided. This applies in the case of rotary axes. The actual value weighting factor (AVWF) must therefore be specified as a fraction for rotary axes if it cannot be represented by 8 decimal places.

Example:

The position sensing motor encoder is parameterized with P171=18 such that one revolution of the rotary axis equals 2^{18} increments per revolution. This must correspond to numerical value 360000.

The actual value weighting factor (AVWF) is therefore as follows:

$$\text{AVWF} = \frac{360000}{2^{18}} = 1.373291015625$$

The calculation produces a number with 12 decimal places which can only be accurately represented by specifying the factor as a fraction.

AVWF factor with integer/decimal places

The actual value weighting factor has 3 integer places and 8 decimal places in two separate components.

Actual Value Weighting Factor	
Integer places 3-digit P169	Decimal places 8-digit P170
0 to 999	0 to 99999999

NOTE

If you want to work in encoder increments instead of physical units, the actual value weighting factor (AVWF) should be set to 1.0. This is recommended, for example, for pure synchronization axes.

Example: Determining the AVWF for a linear axis

The following example calculates the actual value weighting factor (AVWF) for an application in which a motor drives a toothed belt via a gearbox and a drive roller.

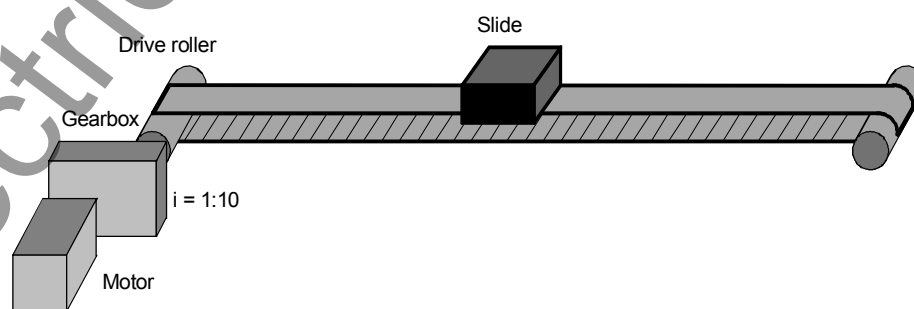


Fig. 9-14

CAUTION

After the AVWF factor has been changed, the converter has to be re-initialized (switch the unit off and on again).

Encoder: Resolver
(P171=12 $\approx$ 4096
increments/revolution)

Gear ratio: 1:10

Diameter of drive roller: 300 mm

How many μm does the slide travel in one encoder increment?

AVWF = number of LUs per position increment

$$\frac{\text{AVWF}}{\left[\frac{\mu\text{m}}{\text{inc}}\right]} = \frac{1}{\text{Gear ratio}} \times \text{Diameter} \times \pi \times \frac{1}{\text{inc / revolution}}$$

$$\frac{\text{AVWF}}{\left[\frac{\mu\text{m}}{\text{inc}}\right]} = \frac{1}{i} \times \left(\pi \times \frac{D}{\left[\mu\text{m}\right]} \right) \times \frac{1}{\frac{2^{P171}}{\left[\text{inc}\right]}}$$

$$\frac{\text{AVWF}}{\left[\frac{\mu\text{m}}{\text{inc}}\right]} = \frac{1}{10} \times 300000 \mu\text{m} \times \pi \times \frac{1}{4096 \text{ inc}}$$

$$\frac{\text{AVWF}}{\left[\frac{\mu\text{m}}{\text{inc}}\right]} = 23,00971181828 \mu\text{m}$$

Resulting Actual Value Weighting Factor (AVWF)	
Integer places 3-digit P169	Decimal places 8-digit P170
23	(00)971181

NOTE

Trailing zeroes must be entered with the decimal place component; leading zeroes can be omitted.

Examples:

AVWF = 12.3 $\rightarrow$ P169 = 12, P170 = 30000000

AVWF = 12.00000003 $\rightarrow$ P169 = 12, P170 = 3

AVWF as numerator/ denominator

When the actual value weighting factor (AVWF) is specified as a numerator / denominator, the length units LU are entered in the numerator and the encoder increments in the denominator.

Example:

A rotary axis is driven via a gear unit with a gear ratio of 1:3. The actual position sensing function is parameterized such that 2^{16} increments correspond to one motor revolution. One revolution on the load side must equal 360000LU.

$$\text{AVWF} = \frac{360000}{2^{16} \cdot 3} = \frac{360000}{196608}$$

Direction of rotation bit P595

The direction of rotation bit (right/left) can be used to invert the direction of rotation of the motor.

Direction of Rotation Bit	
P595	Meaning
0	Right (Clockwise looking towards the output)
1	Left (Counterclockwise looking towards the output)

With resolvers, encoders and pulse encoders, the leading sign and counting direction of the position are inverted. With an absolute encoder (Multiturn or Singleturn encoder), the maximum traversing range is also added.

Example:

Qualitative progression of the absolute position with voltage initialization at the zero point of the encoder and for clockwise rotation looking towards the output.

Clockwise (**P595** = 0) → no difference between encoder and absolute encoder

Counterclockwise (**P595** = 1) → different curves for encoder and absolute encoder

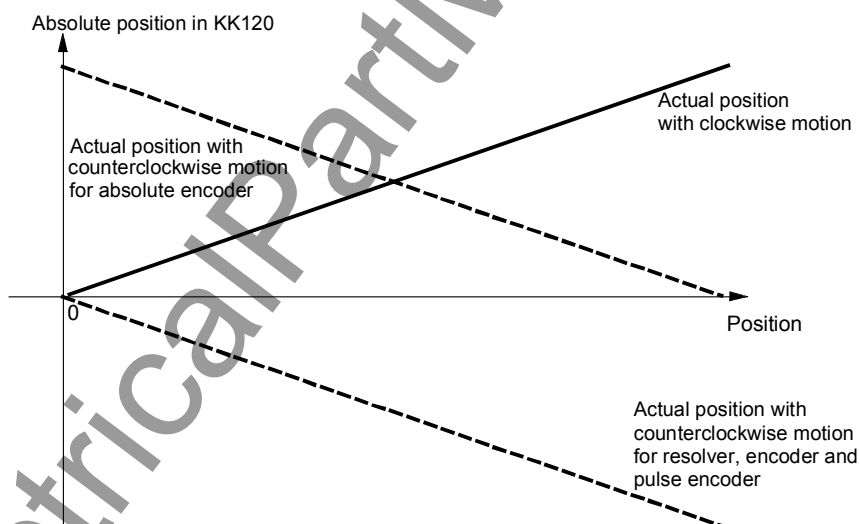


Fig. 9-15

Position correction P174/P175

The position correction is used to correct the actual position by a certain value. [330.5], [335.5]

The position correction is used mainly for:

- ◆ Rotary axis operation with angular synchronization [836.7] and positioning [815.5]. The correction is performed with an overflow from 360° to 0°.
- ◆ The tool offset during positioning

The control signals for adding and subtracting the position correction operate as follows:

Position correction:	Actual position
→ Added	Actual position = actual position + pos. correction
→ Subtracted	Actual position = actual position - pos. correction

The position correction value can be positive or negative.

The time chart below illustrates the signal sequence during position correction.

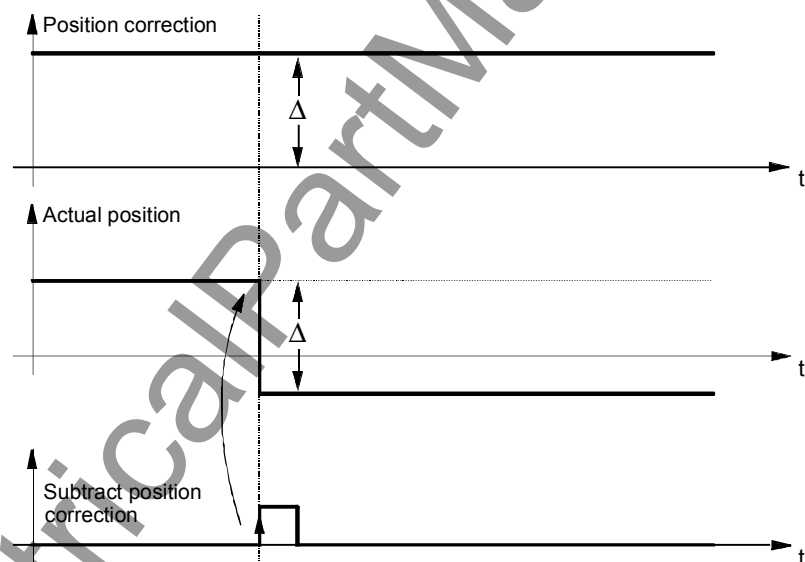


Fig. 9-16

Homing P183

The motor encoder indicates the absolute position within one motor revolution. If the motor covers more than one revolution during a positioning movement, the position sensing system must be referenced with an external rough pulse signal.

NOTE

If a multi-pole resolver has to be referenced, the resolver angle KK96 (available from software version 1.6) has to be connected up to the motor encoder position detection instead of the rotor position KK90 (P182=96). If KK90 were used, the zero pulse would always be detected in the pole pitch in which the resolver happens to be positioned when switched on.

In one mechanical motor revolution, the resolver angle makes the same number of revolutions as the number of pole pairs of the resolver. The number of pole pairs of the resolver must therefore be taken into account in the denominator of the AVWF factor (P180.2).

Position detection uses the zero passage of the connected position angle as a substitute for a zero pulse. It thus detects the same number of zero pulses as the resolver's number of pole pairs. The desired zero pulse is selected with the rough pulse.

The following table provides an overview of the homing modes:

Homing Mode	
To right of BERO P183 = xx11	The home position is the first rotor zero position after the negative edge of the rough pulse. The traversing direction must be positive.
To left of BERO P183 = xx21	The home position is the first rotor zero position after the negative edge of the rough pulse. The traversing direction must be negative

NOTE

When using homing mode in technology option F01 or in SIMATIC M7, the machine data MD5 setting must be identical to P183 [821.3].

Signal sequence for homing P177

A positive edge on the "enable homing" control signal enables the homing logic for one cycle. When the home position is detected, the position is set to the set value and the "home position detected" checkback signal is output [330.7 and 335.7]. The checkback signal remains active until the "enable homing" control signal is canceled again. The time chart below illustrates the sequence.

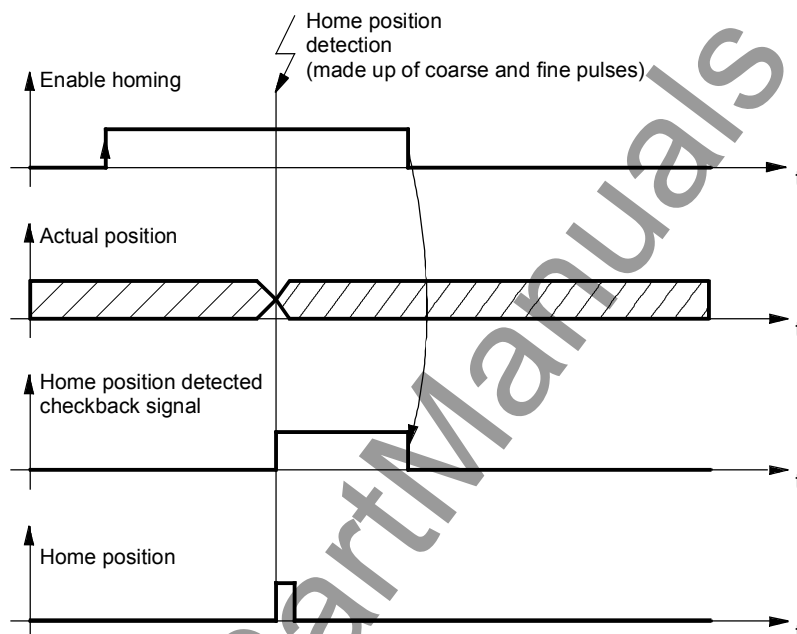


Fig. 9-17

Homing mode to right of proximity switch (BERO)

A rough pulse (proximity switch) signal is required for this homing mode. The home position is the first rotor zero position after the negative edge at the rough pulse input for a positive traversing direction (direction A → B).

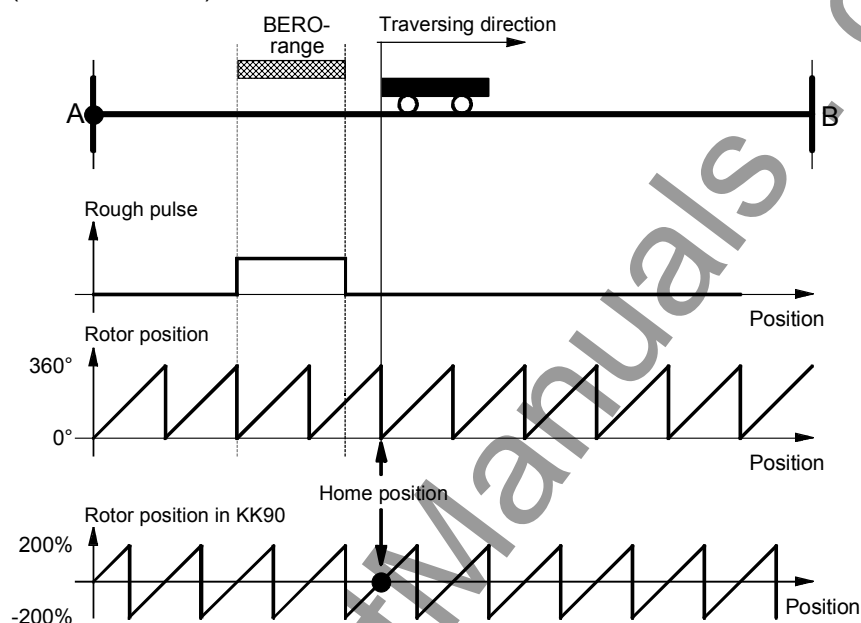


Fig. 9-18

Homing mode to left of proximity switch (BERO)

A rough pulse (proximity switch) signal is required for this homing mode. The home position is the first rotor zero position after the negative edge at the rough pulse input for a negative traversing direction (direction B → A).

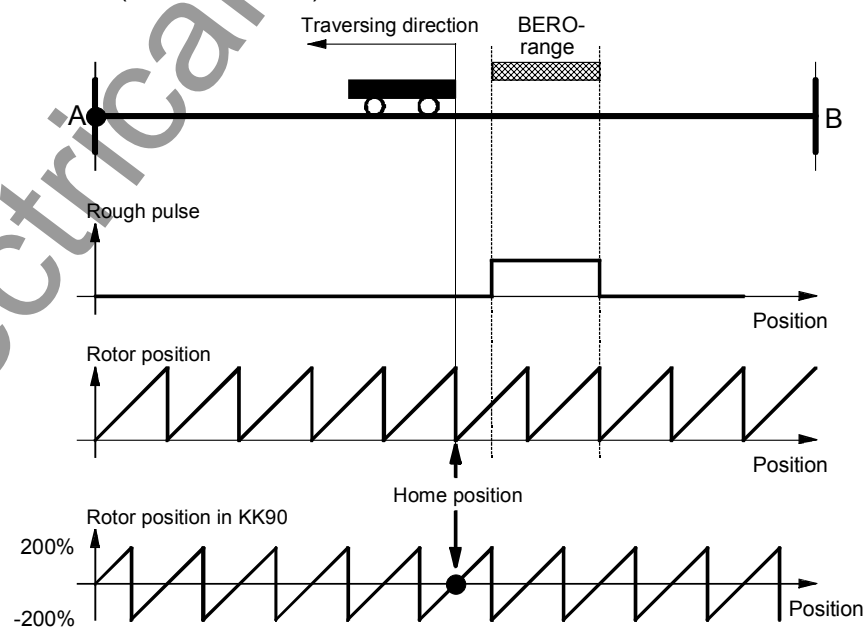


Fig. 9-19

Proximity switch alignment

Since the rough pulse is read in via a digital input, the signal is evaluated in the sampling time of the digital inputs. If the negative edge of the rough pulse is located directly above the rotor zero position, the detection of the home position may be incorrect, because the signal is detected with the inaccuracy of a sampling period.

Example:

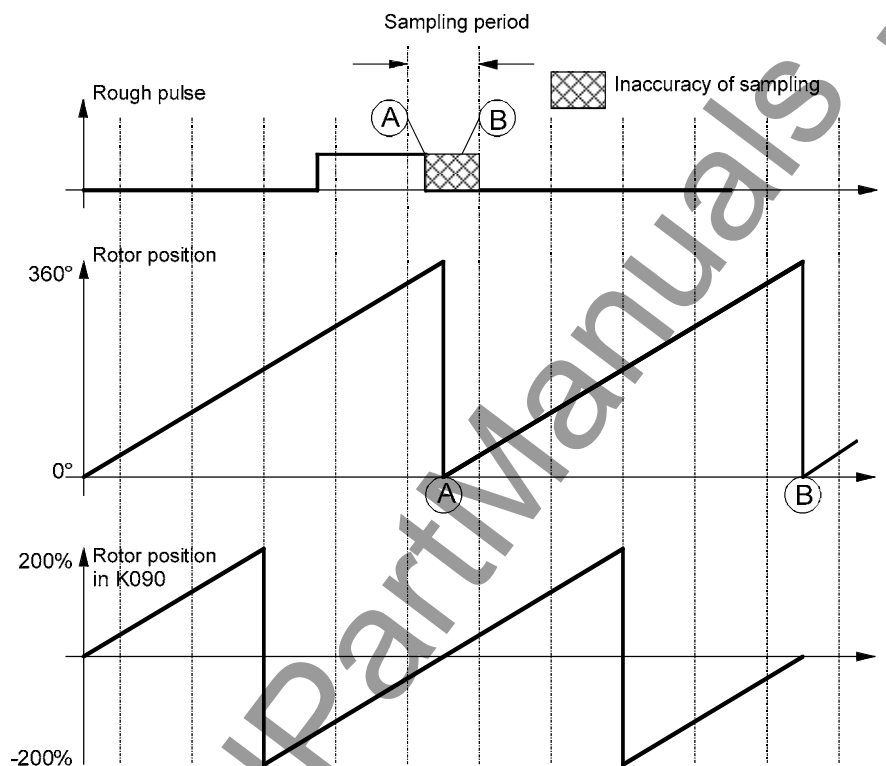


Fig. 9-20

In the configuration shown in the graphic, the negative edge of the rough signal can be detected in front of the rotor zero position (sample A), resulting in the detection of the home position at point A. If the negative edge is not detected until after the rotor zero position (sample B), the home position is located at point B.

To prevent incorrect detection of the home position, the proximity switch must be aligned such that the falling edge does not coincide with the rotor zero, but occurs in the most central position possible between two rotor zero crossings. The rotor zero position can be monitored in KK090 (e.g. using display parameter r033.1, if P032.1 = 90 is set [30.2]).

**Rotor position offset
P188 / r189**

An alternative to mechanical alignment of the proximity switch is to use parameter P188 to define an offset to the measured rotor position. This has the same effect as mechanical alignment of the proximity switch. The offset to be entered in P188 is determined as follows:

Step 1: Perform a homing procedure. If the home position is found, the rotor position measured at the negative edge of the rough pulse is output in parameter r189.

Step 2: The measured value in r189 must be less than -100 % or greater than +100 %. If the value is outside this range, an offset must be specified for the rotor position. The offset value is calculated as follows:

Measured rotor position r189	Offset in P188
Positive, >100 %	No correction necessary
Positive, <100 %	$P188 = 200 \% - r189$ see example in Fig. 9-21 $r189 = 20 \%$ $\rightarrow P188 = 200 \% - 20 \% = 180 \%$
Negative, >-100 %	$P188 = -200 \% - r189$ Example: $r189 = -80 \%$ $P188 = -200 \% - (-80 \%) = -120 \%$
Negative, <-100 %	No correction necessary

The graphic below illustrates the procedure.

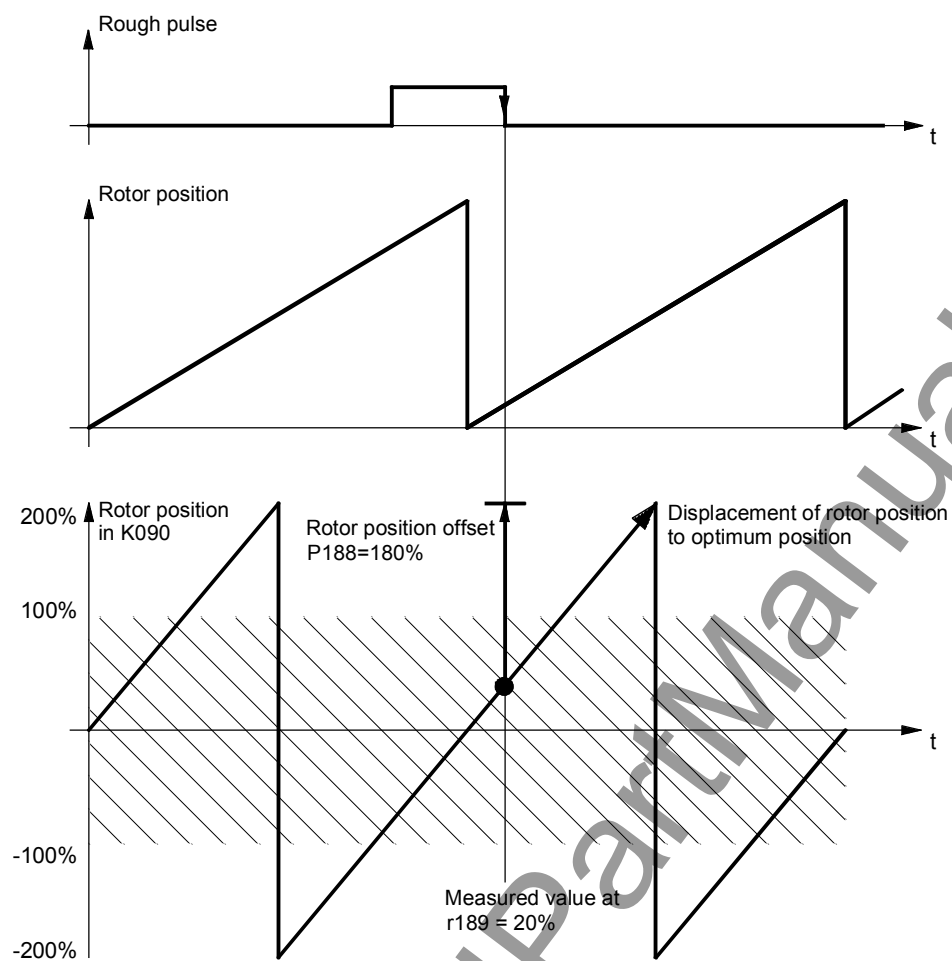


Fig. 9-21

9.4.9 Using absolute encoders for positioning of motors with load-side gearing and rotary axis

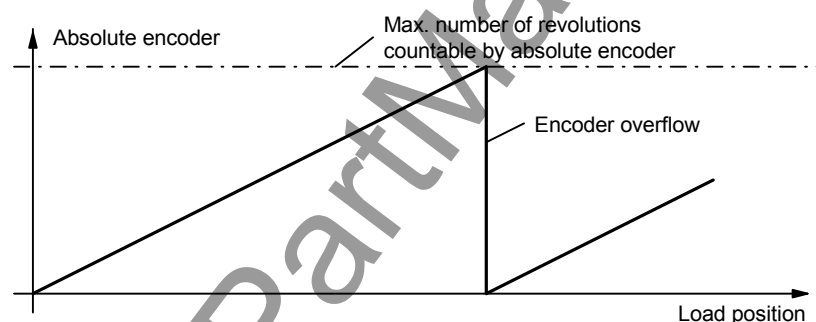
Basic problems

This chapter describes the procedure required when **mechanical gearing** is interposed between a **rotary axis** and the motor and position control is to be performed by an **absolute value encoder** located on the motor. In this case, an additional function block is required which is shown in function diagram 327 for the motor encoder and function diagram 333 for the external encoder.

Absolute encoders capable of counting a definite number of encoder revolutions (such as 4096) are used for angular synchronism and in order to avoid having to start from a home position.

As a rotary encoder turns endlessly in one particular direction, the representation range of the encoder is exceeded. The result is an encoder overflow which means that the encoder starts counting from 0 again after, for example, 4096 rotations.

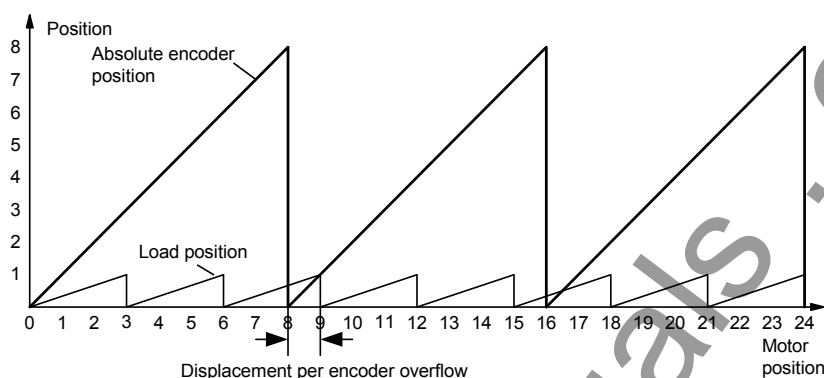
This is illustrated by the diagram below.



To save costs, an absolute encoder is fitted on the motor and used both for torque and speed control and for position control (EQN 1325). One advantage of this is that fitting the encoder on the motor is much less critical but much more precise than fitting it on the load side.

Mechanical gearing is usually located between the motor and the load for speed adjustment. Depending on the gear ratio, an offset between the zero position of the load and the motor occurs on every encoder overflow.

Example: Gear ratio 1:3, absolute encoder can count 8 revolutions



In this case, a load-side offset of 1/3 of a load revolution occurs on each encoder overflow, after three encoder overflows the motor and load zero position coincide again. The load position can no longer be reproduced with certainty after an encoder overflow.

When torque motors are employed, a gear unit may be installed between the encoder axis and the motor axis. This gear ratio can be parameterized in P116. This parameter is included in the calculation of the load offset mentioned above.

Motor encoder position tracking block

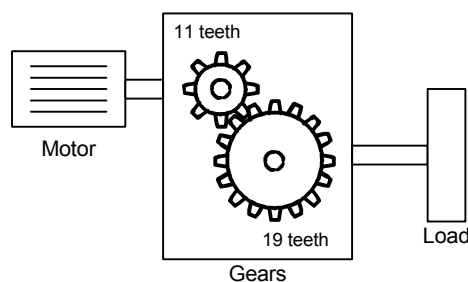
In order additionally to reproduce the position of the load for any gear ratios, the free block "Start position absolute encoder" is used with mechanical gear ratios (function diagram 327 for motor encoder, function diagram 333 for external encoder). The block counts the encoder overflows using the absolute position. The overflow and revolution counter for retentive storage in an tracking memory element is accessed via connector KK625 (KK628). When the module is powered up, the overflow and revolution counter is accessed from the tracking memory element. The starting position for position acquisition is calculated from the absolute position using this information.

NOTE

The visualization parameters in the position tracking block refer exclusively to the encoder data.

The mechanical gear ratio is given by parameter U810 (U795). U810.01 (U795.01) gives the number of gear teeth on the motor side, and U810.02 (U795.02) the number of gear teeth of the load side. It is important to give the numbers of gear teeth, not the circumferences.

Example:



In the example, the motor makes 19 revolutions for 11 revolutions of the load. The value 11 must be entered in U810.01, and the value 19 in U810.02.

NOTE

The ratio given on the gearbox rating plate is often only a rounded value (e.g. 1:7,34). To prevent long-term drift on a rotary axis, the actual ratio of gear teeth must be obtained from the gear manufacturer.

Linking the block

The principal configuration of the block for the motor encoder is shown in function diagram 327. Slotting the block into a time slot automatically ensures that the position acquisition for the motor encoder is placed at the correct starting position. The revolution/overflow counter must be connected with a tracking memory element parameterized for non-volatile data storage. Storage is enabled when valid values are transferred from the encoder evaluator (B070 on TRACK input). After parameterization, the overflow counter should be reset once only and the unit switched OFF and ON again. Thereafter, the overflow counter must not be reset again. Depending on the tracking memory element, the following wiring is necessary.

Tracking memory element 1	Tracking memory element 2
U950.76 = 4	U952.69 = 4
U203.01 = B070	U206.01 = B070
U204 = 625	U207 = 625
U205 = 1	U208 = 1
U811.01 = 551	U811.01 = 552

The same function is provided by software version V1.50 and higher for external encoders, too. The function is shown on function diagram 333. The FD333 and FD327 function blocks differ only in terms of the inclusion of parameter P116 (gear unit encoder-motor) in the position tracking block for motor encoders.

Rotation with the power off

Besides tracking encoder overflows, the block monitors whether the drive is rotated or runs down while no power is supplied to the electronics.

NOTE

A position can be reproduced only if the number of revolutions made with the power off corresponds to less than half of the encoder range. For example, in the case of standard encoder EQN 1325 this is **2048** motor revolutions.

NOTE

The "Position tracking for motor encoder" (function diagram 327) and "Position tracking for external encoder" (function diagram 333) blocks are released only for EQN1325 encoders with 2048 lines. With software version 2.30 and higher, all parameters available for selection in parameter P147.1 or P147.02 can be used.

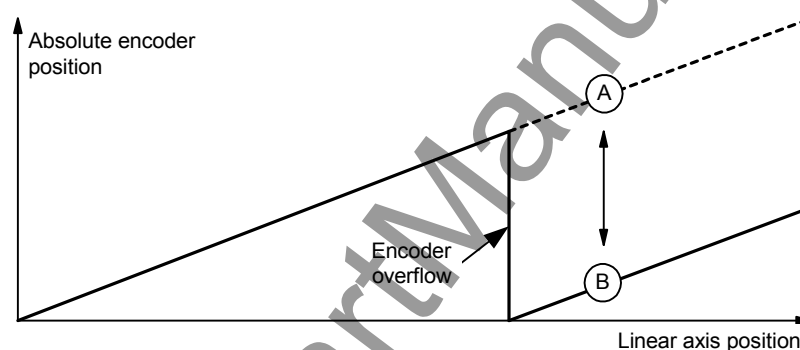
9.4.10 Linear axis with absolute encoder when the traversing range is greater than the display range of the encoder.

The following chapter describes the procedure when the traversing range of a linear axis is greater than the traversing range of the absolute encoder.

Basic problems

Absolute encoders have a limited display range. For example, on multiturn encoder EQN1325 up to 4096 encoder revolutions can be counted. This is sufficient for most applications. However, if the traversing range of the linear axis is greater than the display range of the encoder, an encoder overflow occurs. The position of the axis can no longer be uniquely determined.

This problem is explained in the diagram below:



When the encoder overflows, the absolute encoder starts counting from zero again. The position of the linear axis is position A, but position B is output by the absolute encoder.

Function block "Starting position absolute encoder" function diagram [327] ([333]) ensures the correct functioning of the position acquisition even if the encoder overflows.

The function block is integrated as shown in the function diagram. The block is handled in exactly the same way as that of the rotary encoder (section "9.4.9"), only configuration U813 (U798) has to be parameterized to xxx1 = linear axis.

NOTE

Maximum 15 encoder overflows can be tracked. When the range is exceeded, a fault is displayed at binector B565 (B566).

Handling

Position tracking is set such that the overflow counter lies within the valid range 0 to 15. Underflows below zero must be prevented. The following commissioning procedure must therefore be followed:
The linear axis is traversed to the end stop, so that the smallest possible position actual value results. The overflow counter is then set to zero via U812 (U797) and the converter is switched off and on again.

9.4.11 Position sensing system for external machine encoder [335]

The position sensing system for external machine encoders is shown in [335] and has the same function as the position sensing system for motor encoders [330].

However, connector KK0105 [335.2] has a different scaling to rotor position KK090: Whereas for "position acquisition motor encoder" a connector on which a revolution to 2^{32} is mapped is wired as the source, "position acquisition external encoder" evaluates the increments that the encoder module produces without rescaling. By "increment" we mean the smallest digital unit that the encoder produces:

- ◆ On pulse encoders with two pulse tracks offset by 90° , the pulse edges of both tracks are evaluated. With this type of evaluation known as "pulse edge evaluation", the pulse encoder produces four times as many "increments" per revolution as it has marks (1024 marks = 4096 increments per revolution).
- ◆ On encoders that provide a sine and a cosine track (A/B track), the passages through zero of both tracks are evaluated much the same way as for a pulse encoder. Here, too, four times as many increments per revolution are produced as the number of sine/cosine periods per revolution (2048 periods = 8192 increments). The resolution can be increased additionally with the fine resolution (see below).
- ◆ On SSI encoders or EnDat encoders, which only transfer their positional value to the encoder module via a serial protocol, one increment corresponds to the lowest value bit in the protocol.

Connector KK105 outputs the position actual value in increments. The encoder with the sine and cosine tracks (A/B track) is the exception among the listed encoders. If this encoder is operated on an SBM2 encoder module, the analog value of the A/B track can be evaluated over and beyond simple acquisition of the passages through zero of the A and B tracks, as this module also contains A/D converters with 12-bit resolution. The resolution that can be produced by evaluating the analog signals is called "fine resolution".

On the external encoder the degree to which the resolution of the positional value is to be increased can be selected via parameter 154. In the binary number, the increments are moved to the left by the number of positions parameterized in P154 and the lower bits that are released as a result are filled up with the fine resolution. One increment is divided into 2^{P154} steps. A sensible value for P154 is between 7 and 10.

Please note that the entire position with fine resolution must still fit in a 32-bit number! (example: multiturn encoder EQN1325: Revolutions 12 bits + increments 13 bits + fine resolution 7 bits = 32 bits).

9.4.12 Position control system [340]

The position controller is shown in [340]. The method used to connect the position controller to the technology is described in [801 + 817] and in the section entitled "Commissioning the technology".

The position control system [340] is implemented using a PI controller with a deactivatable I component.

Actual position smoothing P195

If the actual position signal is very unstable, it can be stabilized using the actual value smoothing parameter. It should be remembered, however, that smoothing the signal reduces the potential dynamic response of the system. The set input is used to synchronize the output of the smoothing element with the actual value of the position sensing system during setting or correction processes, e.g. on a rotary axis with tool compensation. Synchronization is only necessary if a smoothing time constant has been entered in P195.

Position setpoint smoothing P191

Position setpoint smoothing is only appropriate if the speed precontrol feature of the position control system is used. In this case, the smoothing time constant should be set to the equivalent time constant of the speed control loop. Position setpoint smoothing is not usually required. Position setpoint smoothing must also be set when the position setpoint is set.

Position difference smoothing P199

Position difference smoothing is preferred for rotary axes and angular synchronization, since it overcomes the problems associated with setting events. Like actual position smoothing, position difference smoothing reduces the potential dynamic response of the drive.

Position setpoint interpolator P770/P771

If the position setpoint (e.g. for synchronization or positioning) is generated in a slower sampling time than that of the position controller, the consequence is jumps in the setpoint for the position controller. This results in unstable operation and reduces the level of accuracy that can be achieved. In order to optimize the transition between sampling times, it is possible to convert the rough graduation of the setpoint commands into a fine graduation for the position controller. This task is performed by the position setpoint interpolator, the operation of which is defined by two parameters:

P770 defines the ratio between the sampling time of the position setpoint generator to the sampling time of the position controller in steps of 2^{P770} . Example: time slot of the position setpoint generator = T_4 , sampling time of the position controller = T_1 , $P770 = 3$.

If P770 is set to a positive value, the position setpoint is extrapolated (lookahead calculation). If P770 is negative, the setpoint is interpolated. The extrapolation option must be used when no speed precontrol is active on the position controller. If the precontrol function is active, interpolation of the position setpoint should be selected instead.

P771 defines the limit for the setpoint change, referred to the sampling time of the prior position setpoint detection.

The following setting rule applies for P771:

$$P771[\text{LU}] = \frac{2 \cdot P205 [1000 \text{ LU/min}] \cdot \left(\frac{\text{Scanning time of the prior position setpoint detection [ms]}}{60} \right)}{60}$$

If the setpoint change is below the limit, the interpolation is performed. If the change is above the limit, the position setpoint is transmitted immediately. This function is required in order to disable interpolation during setting events.

The graphic below illustrates the response of the interpolator:

P770 = -2

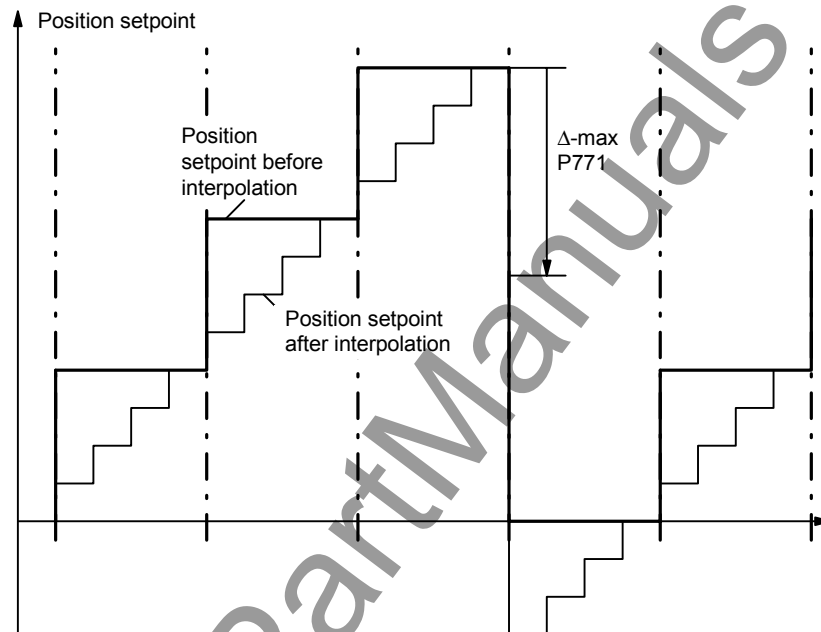


Fig. 9-22

When the extrapolation option is used, the setpoint is conditioned as follows:

P770 = +2

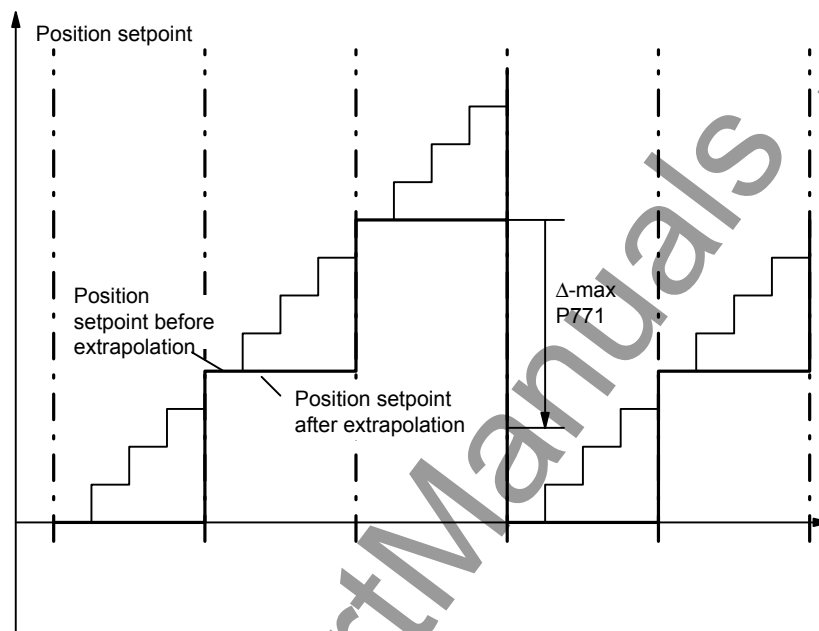


Fig. 9-23

Loop gain factor P204

The loop gain factor represents the proportional gain of the position controller. It is defined such that the setting is independent of the encoder resolution and the traversing velocity from the perspective of the user.

V_{rated} P205

It is important that the rated velocity in P205 actually represents the velocity of the drive at 100 % speed setpoint (defined in P353).

Example:

Reference speed of motor: 3000 rpm (P353)

Gear factor: 1:10

Diameter: 300 mm

$$V = \text{Reference speed} \times \frac{1}{i} \times \text{Diameter} \times \pi$$

$$V = 3000 \text{ rev / min} \times \frac{1}{10} \times 300 \text{ mm} \times \pi$$

$$V = 282743 \frac{\text{mm}}{\text{min}}$$

This rated velocity must be entered in P205 and must also be entered in MD23 [804] if the technology option is used.

The rated velocity can also be derived from the hardware parameters.
The following example refers to the use of the motor encoder:

AVWF	P169 / P170
Actual pos. resolution	P171
Reference velocity	P353

$$\left[\frac{V}{\frac{1000 \text{ LU}}{\text{min}}} \right] = \left[\frac{P353}{\frac{1}{\text{min}}} \right] \times \left[\frac{\text{AVWF}}{\frac{\text{LU}}{\text{inc}}} \right] \times \frac{2^{P171}}{[\text{inc}]} \times 10^{-3}$$

9.4.13 Technology overview and mode manager [802]

[802] provides you with a general overview of the technology functions with references to all relevant pages of the function diagram. Sheet [802] thus represents a "graphical table of contents" for all of the technology functions. A rough sketch is also provided of the signal exchange between the technology and the basic unit functions of the position controller, speed controller and position sensing system.

The mode manager connects the input signals to the current operating mode selected by [MODE_IN].

Input signals include machine data MD1 to MD50, positioning control signals, the special "digital inputs for positioning" and the position signals from the position sensing system.

As you can see in the graphic below, 7 modes are available for selection: positioning modes 1 to 6 and synchronization mode 11.

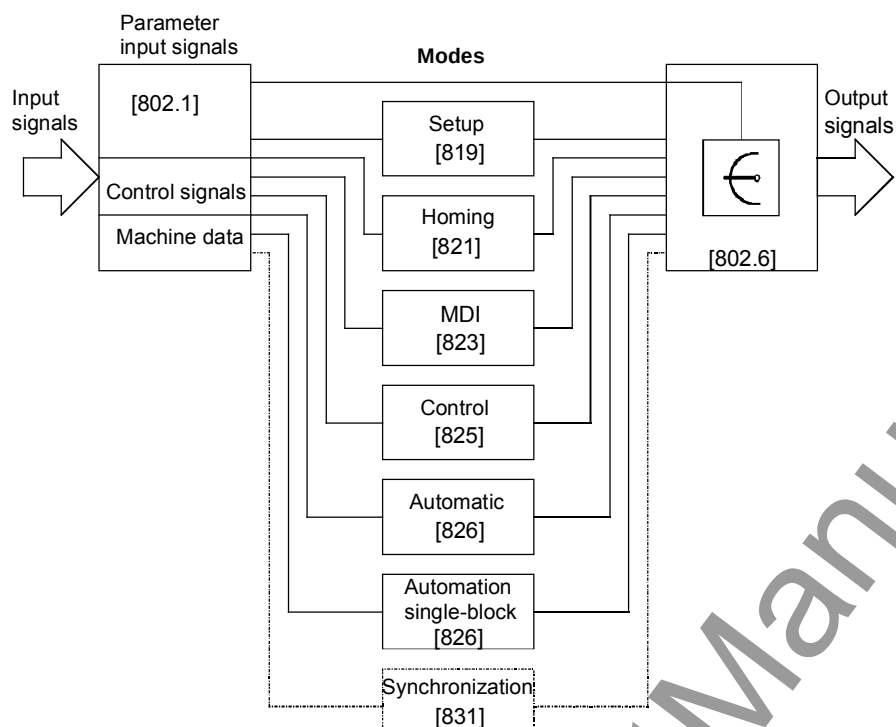


Fig. 9-24

Mode	Use
Setup	Position-controlled movement of drive at constant velocity
Homing	Used for referencing with incremental encoder types
MDI	Used to define and execute an NC data block for a point-to-point positioning operation
Control	Speed-controlled mode
Automatic	Automatic execution of NC programs
Automatic single-block	Block-by-block execution of NC programs for test purposes, etc.

Nesting positioning and synchronization in a sampling time

The technology functions are not calculated until they have been nested in a sampling time. One parameter each is provided for nesting the following functions in a sampling time:

- ◆ Positioning modes (including synchronization)
Parameter U953.32 [802.8]
- ◆ Synchronization as independent free block
Parameter U953.33 [802.8]
- ◆ Virtual master axis (can also be used without F01)
Parameter U953.34 [832.8]
- ◆ Generation of positioning control signals
Parameter U953.30 [809.5]

Please refer to [702] and the notes in [802.8] for more information on nesting of technology functions in a sampling time.

Positioning including synchronization can be nested in a sampling time with parameter U953.32. The value 4 is the preferred value for this parameter ($= 16 \cdot T_0 = 3.2 \text{ ms}$ at 5 kHz converter clock frequency).

Synchronization can also be activated as an independent block, preferably in T4 [U953.33 = 4]; in this case the mode manager is not used and the positioning modes must remain deactivated with U953.32 = 20 (see the section below entitled "Synchronization mode - overview" for information on the differences in synchronization).

The mode manager switches the output signals from the active mode through to the signal outputs [802.5].

9.4.14 Machine data [804]

The machine data are used to define centralized settings required from the perspective of the working machine and the mechanical transfer elements for positioning and synchronization. The machine data are abbreviated with "MD..." in all documents. They have the same meaning for technology option F01 and for the centralized technology in SIMATIC M7.

MD1 to MD50 are listed in a brief overview in [804]. They are mapped onto MASTERDRIVES parameters U501.01 to 501.50.



You will find detailed information on all machine data in the "Machine data" section of the Function Description in manual /1/. Please note that the decimal point has been omitted from all machine data on sheet [804], since they appear in this format in the MASTERDRIVES parameter display. In manual /1/, however, the machine data are presented in the format used in the OP screens (see also /2/, i.e. occasionally with decimal points).

Example for presentation of decimal points in machine data:

- ◆ Value range for MD14
 - in manual /1/: 0.001...99.999 [1000*LU]
 - in MASTERDRIVES MC: 1...99 999 [LU]
- ◆ Input value for following error of 300 LU in MD14:
 - in manual /1/: 0.300
 - in MASTERDRIVES MC: 300

Changes to machine data must be transferred with U502 = 2 [804.3]. This is only possible at a standstill. A machine data transfer is also initiated when the electronic power supply is switched on/off.

After one or more machine data have been changed, U502 changes automatically from value "0" to value "1". After transfer of the machine data by U652 = 2, U652 automatically changes to value "0" if no errors were detected in the machine data.

If the machine data contain an error, the changes are not accepted, U502 is set to 1 and an error message is output to n500. At present only one error is possible, i.e. "negative limit switch is located to right of positive limit switch", i.e. MD12 > MD13.

CAUTION

If the machine data are modified with a DriveMonitor download file, the MASTERDRIVES electronic power supply must be switched off and on again, in order to activate the new machine data.

NOTE**Machine data for synchronization**

If you use synchronization, but no positioning functions, only machine data MD11 and MD49 are relevant [836.4 + 836.7]; if synchronization is nested as a positioning mode, MD12, MD13 and MD15 are also referenced. See also the section below entitled "Synchronization mode - overview".

9.4.15 Parameter download file POS_1_1 [806]

DriveMonitor download file POS_1_1 is used to set up the message configuration of the 10 process data words in the send and receive direction for the S7 software "Motion Control Configuring Package" /1/. This assignment is described in the "Control and checkback signals" chapter of the Function Description in manual /1/. See also the section below entitled "Procedure for using the GMC-BASIC S7 software" in "Commissioning the technology" and the section entitled "Communication with the technology".

9.4.16 Positioning control signals [809]

There are two ways to define the positioning control signals:

- ◆ U530 can be used to select any double-connector as the source of the positioning control word. When defining the control signals via PROFIBUS-DP, for example, U530 = 3032 would assign this function to receive words 2 and 3 on the communication board [120] (double-connector KK3032).
- ◆ With the factory setting U530 = 860, the control signals are defined by binector with U710. In this case, you must remember to nest the "control signal generation" block in a sampling time using U953.30 (recommended setting: U953.30 = 4). Any binectors can then be used as the source of the individual control commands.



The positioning control signals are described in detail in the "Control and checkback signals" chapter of the Function Description in manual /1/.

9.4.17 Positioning status signals [811]

The status signals are routed via various binectors and display parameters to double-connector KK315, the positioning status word. For example, you can connect the positioning status word to send words 3 and 4 [125] of the communication board (e.g. PROFIBUS-DP interface) with P734.3 = 315 and P734.4 = 315. The status bits available at binectors B351...B361 can be wired in any configuration using BICO technology.



The positioning status signals are described in detail in the "Control and checkback signals" chapter of the Function Description in manual /1/.

9.4.18 Digital I/Os for positioning [813]

Digital I/Os for positioning

U536 and MD45/MD46 allow you to use any binectors of the MASTERDRIVES MC for special positioning control functions. The digital inputs of the converter terminal strip X101 or the terminal expansion boards EB1/EB2 can be selected as binectors. You can also connect binectors, which are generated by logic circuits using the free blocks [765...780], to this point.

Digital outputs for positioning

MD47/MD48 can be used to assign special positioning status functions to binectors B311...B316. These binectors can be wired in any configuration using BICO technology, e.g. to PROFIBUS-DP or to digital outputs on the converter terminal strip or the terminal expansion boards EB1/EB2.



You will find detailed information on the digital I/Os for positioning under MD45 to MD47 in the "Machine data" chapter of manual /1/. Please note that this documentation is based on a special digital I/O assignment for positioning with converter terminal strip X101, however this configuration is not a mandatory requirement for general applications.

9.4.19 Evaluation and control of the position sensing system, simulation mode [815]

Position sensing

Sheet [815] depicts the interconnection of the technology with the position sensing system for the motor encoder [330] and the external machine encoder [335].

The upper section details the measured values and status signals that the technology requires from the position sensing system. The lower section shows the control signals and set/offset values transmitted from the technology to the position sensing system.

Each section has 2 columns specifying the parameters required for the connection between the technology and the position sensing system for the motor encoder or external machine encoder. With the factory setting, the connection to the motor encoder is mostly intact, and so only a small number of parameters needs to be modified in this case. The parameter settings are listed in "Connection and parameters of the position sensing system" in the section entitled "Commissioning the technology".

Simulation mode

General information on simulation mode

During simulation mode, the position actual-value is simulated by the position encoder, i.e. all the functions of the axis, including setpoint output (at parameters n540.01, n540.10 and n540.37 [817]), automatic mode and the M functions can be tested without a position encoder and drive. Even if a motor is connected up, no axis movement takes place. This is achieved by setting the position setpoint KK310 to the current position actual-value and by setting the speed and acceleration precontrol KK312 and KK312 to "0" [817].

Via the simulation mode it is also possible to test the interaction of a higher-level control system with the positioning functions in the drive.

An axis can be activated via U503 independent of the selected operating mode of simulation (U503 = 1) and switched back to normal operation (U503 = 2).

If the SIMATIC M7 standard software GMC BASIC /1/ is used, simulation can be selected or deselected via the "Simulation input" task. The selection is stored in the EEPROM.

Activating simulation mode

After "Simulation ON", the technology must be reset via the control signal [RST] (Reset technology) or the drive must be re-energized (Power OFF/ON). Simulation is not activated until this has taken place.

De-activating simulation mode

After "Simulation OFF" the technology must be reset via the control signal [RST] (Reset technology) or the drive must be re-energized (Power OFF/ON). Simulation is not de-activated until this has taken place.

9.4.20 Setpoint output and enabling [817]

Sheet [817] shows the output of the following setpoints to the basic unit:

- ◆ **Position setpoint** (with jerk limiting)
- ◆ **Speed setpoint** for the speed control modes (homing and control)
- ◆ **Speed precontrol value** for the position control modes (setup, MDI, automatic, synchronization)
- ◆ **Acceleration precontrol value** (not yet implemented in V1.2)

Binector B305 is used to switch between the position control modes (B305 = 0) and the speed control modes (B305 = 1).

At the right-hand border of [817] you will find the parameters required in order to connect these signals to the position, speed and torque control systems.

9.4.21 Faults, warnings, diagnostics [818]

The main faults and warnings generated by the technology are shown on sheet [818] together with diagnostic parameter U540 of the technology.

You will find further information on faults, warnings and diagnostics in the section of the same name at the end of this chapter.

9.4.22 Setup mode [819]



You will find detailed information on "Setup mode" in the chapter entitled "Setup mode" in the Function Description of manual /1/.

Mode 1 "setup" allows the axis to be moved in position control mode using the direction commands "jog forwards" [J_FWD] and "jog backwards" [J_BWD].

The "fast/slow" command [F_S] can be used to switch between two velocity levels, which can be set in U510.1 and U510.2. The two velocity levels are multiplied by the override.

In order to prevent abrupt changes in velocity, the output setpoint is controlled by a ramp-function generator, the ramps of which can be adjusted with MD18 and MD19.

Software limit switches MD12 and MD13 are evaluated, however this is only the case with an incremental position encoder if the axis has already been referenced (status bit [ARFD] = 1). A start command [STA] is not required for setup mode.

Setup mode is useful for commissioning and maintenance work and machine setup, for example. Setup mode also includes the teach-in feature, which allows you to enter the current position in an automatic NC program block.

The control signal commands for the positive direction of movement are shown in the diagram below.

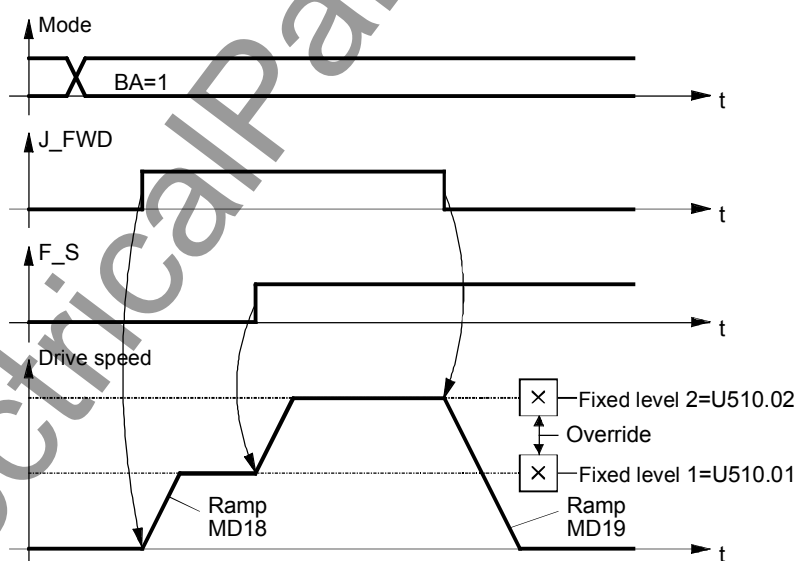


Fig. 9-25

9.4.23 Homing mode [821]

Mode 2 "homing" is only necessary with incremental encoders, i.e. when using a resolver, optical sin/cos encoder or pulse encoder. Homing is not necessary when using an absolute encoder or roll feed. With incremental encoders, the axis must be moved to the home position before a position control mode (setup, MDI, automatic) can be started.

WARNING



- Automatic reversal of the traversing direction does not take place during the homing procedure if a hardware limit switch is reached. Hardware limit switches must be evaluated by the external machine control system and additionally - if they are safety-related - by the hardware (see also the danger notice in "Commissioning the technology").
- No monitoring takes place when a movement is started, although the "axis referenced" status signal [ARFD] is active; this status bit must be evaluated by the external machine control system.



You will find detailed information on homing mode in the chapter entitled "Reference point approach" in the Function Description of manual /1/.

When incremental position encoders are used, there is no relationship between the measuring system (incremental encoder) and the mechanical position of the axis when the controller is switched on. For this reason, the axis must be moved to a defined home position each time the system is switched on, i.e. in order to reference the axis.

There are two ways to reference the axis:

- ◆ With homing, the axis travels across a proximity switch (BERO) (rough pulse) to the zero pulse (fine pulse) of the incremental encoder. When the fine pulse is detected, the measuring system is set to a defined coordinate, thereby establishing the absolute position reference to the mechanical system.
- ◆ With "set home position", the coordinate is set immediately on activation by the user program. The home position thus depends on the mechanical position of the axis at the time the home position is set.

In most cases, the homing procedure is used to reference the measuring system, since this method is accurate to the nearest increment.

"Set home position" is used if neither a rough pulse (proximity switch) nor a fine pulse is available or if the application requires the axis to be referenced at different positions.

Parameter settings The diagram below shows an overview of the parameter settings.

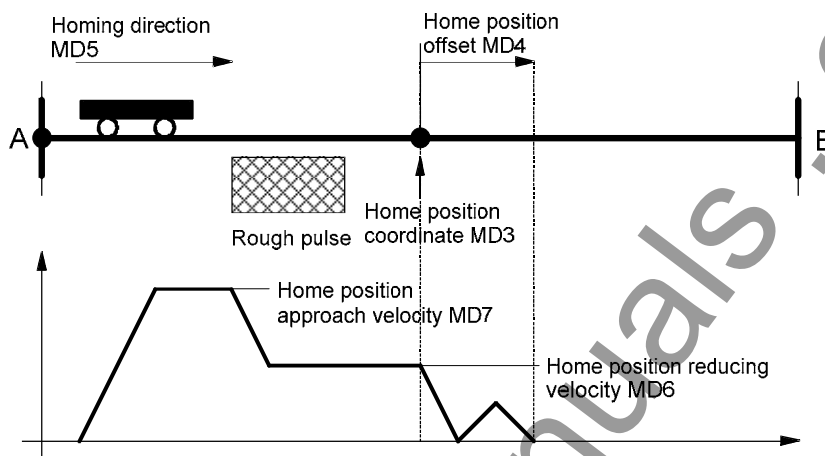


Fig. 9-26

Example

The following example shows the referencing sequence.

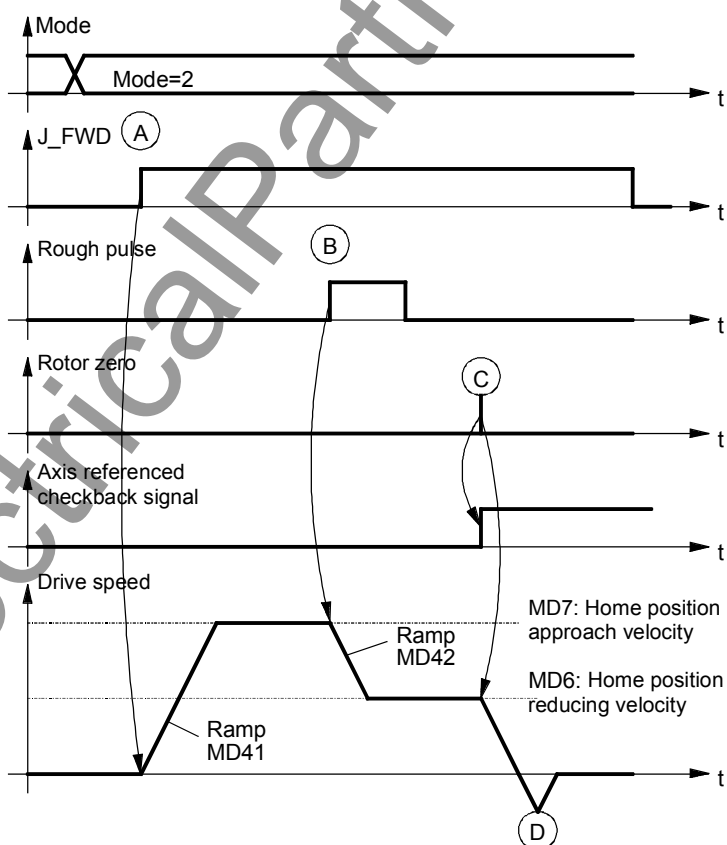


Fig. 9-27

- (A) On activation of the mode 2 command, the axis is started with jog forwards or jog backwards. The drive accelerates to the home position approach velocity MD7. The user must ensure that the home position is crossed in the correct direction. The homing direction is defined in MD5 and must match the setting on the position sensing system (motor encoder P183). The limit switches are not evaluated.
- (B) When the rough pulse signal is detected, the drive decelerates to the home position reducing velocity MD6.
- (C) When the next rotor zero is detected, the drive is brought to a standstill. The axis referenced checkback signal (ARFD) is output.
- (D) The drive realigns itself at the home position.

Connection of the rough pulse

The rough pulse must be wired both via parameter P178 (for motor encoders) to the position sensing system and to the positioning system. This is performed via one of the digital inputs of the positioning system, connected to the digital inputs with parameter U536. The function of the digital input is defined with MD45.

Example 1: Motor encoder with resolver, rough pulse connected to digital input 4 (terminal X101.6, see [90.5]).

Par.	Value	Meaning
P178	16	Rough pulse for position sensing from digital input terminal 6 [330.5]
U536.4	16	Digital input I4 for positioning from digital input terminal 6 [813.1]
U501.45	xx7xxx	Function of digital input I4 for positioning input is proximity switch for homing. MD45 [813.4]

Example 2: Machine encoder with incremental encoder, rough pulse connected to rough pulse 1 of pulse encoder evaluation for machine encoder (function diagram 255.3, connector X400/64).

Par.	Value	Meaning
U536.4	66	Digital input 4 of positioning system from rough pulse 1 of pulse encoder evaluation for external machine encoder [255]
U501.45	xx7xxx	Function of high-speed input is proximity switch for homing

The homing procedures (with homing switch and encoder zero mark) implemented up to V.1.32 are only partly suitable for rotary axis applications. Elaborate adjustments (adapter gears, etc.) have to be performed in order to make use of the present homing procedures.

New homing procedures have therefore been implemented:

1. Homing with homing switch only
2. Homing with encoder zero mark only
3. Use of a reversing switch during homing.

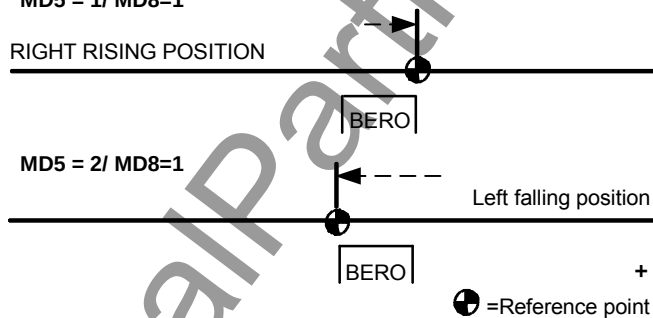
9.4.23.1 Homing with homing switch only

Homing and referencing are governed solely by the homing switch. The zero mark of the encoder is disregarded.

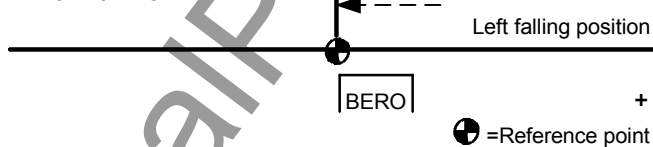
1. New machine datum MD8 for defining referencing
2. 0 = Homing with zero and zero mark (<V1.4x)
 - 1 = Homing with zero only
 - 2 = Homing with zero mark only

MD5 = 1/ MD8=1

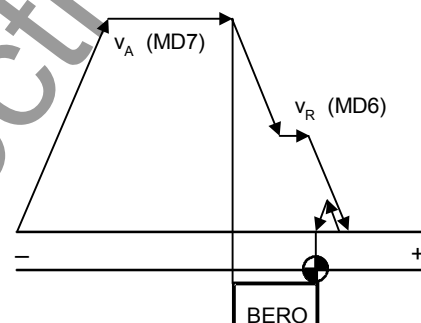
RIGHT RISING POSITION



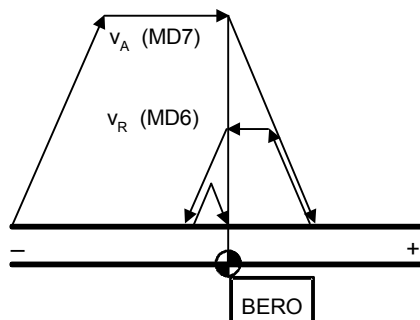
MD5 = 2/ MD8=1



With homing switch only, reference point at right



With homing switch only, reference point at left



NOTICE

If the axis is already on the switch at the beginning of homing, this fact must be taken into account.

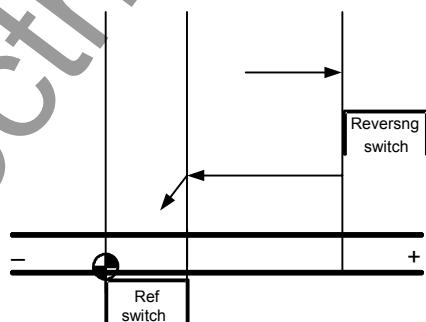
9.4.23.2 Homing with encoder zero mark only

This function is implemented analogously to the description given in section 9.4.23.1. However, only the zero mark of the encoder is used as reference signal. For reasons of accuracy, starting should be at reduced speed.

9.4.23.3 Use of a reversing switch during homing

It was previously necessary during homing to ensure that the reference point was pointing in the right direction when the axis was at rest. If not, the axis traveled to the limit switch.

When a reversing switch is also evaluated, the direction of travel is given either by the homing (same procedure as before) or by the reversing switch (in which case the axis reverses and looks for the homing switch in the other direction).



The reversing switch is always active in homing mode. A digital output (MD45 = 8) can be used for connecting the reversing switch.

9.4.24 MDI mode [823]

Point-to-point positioning - it's easy

Mode 3 "MDI" allows easy point-to-point positioning on the external control system. The abbreviation "MDI" is derived from NC technology and stands for "Manual Data Input".

A simple MDI positioning operation requires the following steps [823.5]:

- ◆ **Step 1:** Define an MDI block using 5 words (8 bytes) over the field bus or select an MDI block, which is permanently stored in 3 indices of a parameter. An MDI block comprises:
 - G functions (1 word specifying whether positioning is absolute or relative and – if desired – an acceleration factor)
 - Position (1 double word, target position for absolute positioning or distance to be crossed for relative positioning)
 - Velocity (1 double word)
- ◆ **Step 2:** Define a start command [STA]
- ◆ **Step 3:** Wait until the "destination reached, axis stationary" status bit [DRS] changes to "1"
→ the traversing operation is complete, the axis is in position.

These steps are described in detail below:

Define the "MDI block" [823.4...6]

The first step is to define the desired MDI block. An MDI block describes the reference data of a positioning operation and comprises 3 components:

- ◆ Two "G functions" (this expression is also derived from NC technology):
The **first G function** defines whether the traversing operation is to be performed in absolute or incremental (relative) dimensions, i.e. whether the specified target position refers to the home position or the current position. On incremental measuring systems, the home position is defined by the home position coordinate MD3 [823.4]; on absolute encoder systems, it is defined by the zero point of the position encoder. Only relative positioning is used on roll feeds.

The first G function can accept two values:

- 90 = Positioning in absolute dimensions
- 91 = Positioning in incremental dimensions (relative positioning)

The **second G function** defines the acceleration override", that is a reducing factor, which can be set in steps of 10 %, for the acceleration/deceleration of the traversing ramps defined in MD18 and MD19. The second G function can accept the following 10 values:

- 30 → acceleration = MD18,
deceleration = MD19 (normal setting)
- 31 → acceleration = 10 % of MD18,
deceleration = 10 % of MD19
- 32 → acceleration = 20 % of MD18,
deceleration = 20 % of MD19
- 33 → acceleration = 30 % of MD18,
deceleration = 30 % of MD19
- ...
- 39 → acceleration = 90 % of MD18,
deceleration = 90 % of MD19
- **Position** in [LU] units, i.e. length unit defined by the actual value weighting factor (AVWF)
- **Traversing velocity** in [10 LU/min] units; e.g. AVWF sets
1 LU = 1 µm, desired velocity = 1000 mm/s ==> input value =
6 000 000

Further below you will find two practical examples of MDI blocks.

Select MDI block [823.3]

11 MDI blocks are available, of which one can be selected with the 4 control bits [MDI_NO] using the large selectors [823.3 and 809.4] at the top border of [823]. MDI block 0 can have any 3 connectors as a source. These can be selected with parameters U531, U532 and U533 (the G functions use a "single connector" as the source, the position and velocity use a double-connector). The remaining 10 MDI blocks, numbers 1 to 10, are stored in the non-volatile triple-index parameters U550...U559.

MDI block 0 can be transmitted to the MASTERDRIVES via a field bus (PROFIBUS-DP, USS etc.). MDI blocks 1...10 can be selected using the digital inputs of the converter terminal strip.

Numeric representation of G functions

The G functions are represented in hexadecimal format in the connector selected by U531 (MDI block "0") and in decimal format in non-volatile parameters U550.1...U559.1 (MDI blocks 1...10).

Example: Absolute positioning with 100 % acceleration override: value of connector = 5A1E (hex), value setting of fixed parameter 90 30 (decimal). 9030 is also the factory setting for the permanent G functions.

The representation of the position and velocity is identical in the double-connectors and parameters.

Example 1:
Define permanent
MDI block by
parameters

- ◆ The MDI block is to be stored as permanent MDI block no. 2 in parameter U551 [823.4].
- ◆ A length unit of 1 LU = 1 µm has been defined with reference to the actual value weighting factor (AVWF) (see the section entitled "Position sensing system for motor encoder").
- ◆ The axis is to be positioned at target position 385.123 mm using absolute dimensions.
- ◆ The traversing velocity is 65 000 mm/min.
- ◆ The movement is to be performed at 100 % of the acceleration/deceleration set in MD18/MD19.

→ The following parameters need to be entered:

U710.09 = 1 Select MDI block 1 [809.3], in this case with fixed binector "1". Any binector can be connected here, for example a digital input

U551.1 = 9030 90 = absolute dimensions, 30 = 100 % acceleration/deceleration

U551.2 = 385123 Target position = 385.123 mm = 385 123 µm = 385 123 LU

U551.3 = 6500000 Velocity = 65 000 mm/min = 65 000 000 µm/min = 65 000 000 LU/min (input in [* 10 LU/min])

Example 2:
 Define a variable MDI
 block via PROFIBUS-
 DP

- ◆ The MDI block is to be defined using receive words 6 to 10 of PROFIBUS-DP [120.6], i.e. as MDI block no. 0 [823.4].
- ◆ A rotary table is used. A length unit of 1 LU = 0.001° has been defined with reference to the AVWF.
- ◆ Relative (incremental) dimensions are to be used to position at a target position located -12.345° away from the current position.
- ◆ The traversing velocity is 190°/min.
- ◆ The traversing movement is to take place at only 30 % of the acceleration/deceleration set in MD18/MD19, since the rotary table is carrying a heavy load.

→ It is necessary to wire the MDI block from PROFIBUS to MDI mode by setting the following parameters:

U531 = 3006 Wire G functions from PROFIBUS receive word 6 [120.6] to MDI block no. 0 [823.3]

U532 = 3037 Wire PROFIBUS receive words 7 and 8 as double word connector KK3037 [120.6] to "position" of MDI block no. 0 [823.4]

U533 = 3039 Wire PROFIBUS receive words 9 and 10 as double word connector KK3039 [120.6] to "velocity" of MDI block no. 0 [823.6]

→ The contents of the PROFIBUS message frame for defining the MDI block are as follows:

Word 6 = 5B 21 (hex) ; 5B (hex) = 91 (decimal) = "relative traversing"
 ;(in incremental dimensions) 21 (hex) = 33
 ;(decimal) = "30 % acceleration/deceleration"
 Words 7 and 8 ; -12.345° = -12345 LU = FFFF CFC7 (hex)
 = FFFF CFC7 (hex)
 Words 9 and 10 ; 190°/min = 190 000 LU/min ==> input value
 = 0000 4A38 (hex) ; in [10 LU/min] = 19 000 (decimal) = 4A38 (hex)

Start of the traversing operation

A simple traversing operation is started as follows:

- ◆ Drive ON (OFF1=1; inverter enable [ENC] can remain permanently at "1"; [180])
- ◆ Select MDI mode [MODE_IN] = 3 [809.4]
- ◆ Wait for a mode checkback signal [MODE_OUT] [811]
- ◆ Set the start command [STA] to "0" [809.4]
- ◆ Wait for a start enable [ST_EN]
- ◆ Evaluate any warnings/faults (bits 3 and 7 in basic unit status word 1 [200], connector K0250 [510], parameter n540.26 [818])
- ◆ Initiate the start command (0 => 1 edge at [STA])
- ◆ The "function terminated" status bit [FUT] switches to "0" when the start command is activated and switches to "1" when the movement is finished or aborted in the event of a fault [811.4]. The [FUT] is a reliable indication that the traversing movement is finished - even in extremely short movements.
- ◆ The "destination reached, axis stationary" status bit [DRS] indicates with a "1" signal that the drive has stopped in the "exact stop window" [811.4]. The exact stop window is defined by machine data MD16 and MD17.

Wait for the checkback signal indicating the end of the traversing operation

Velocity override

You can use the velocity override [823.3] to modify the traversing velocity defined in the MDI block by a factor of 0 ... 255 %, e.g. during commissioning. The velocity override can also be varied in motion and defined, e.g. via U708 [809.1], over a field bus or from an analog input (the source connector can be selected with U709 [809.1] or U530 [809.7]).

Further information on MDI mode



You will find detailed information on "MDI mode" in the chapter of the same name in the Function Description of manual /1/. The "MDI on the fly" function is described there. With MDI on the fly, the MDI traversing record is supplied via MDI traversing record 0. The difference to the "normal" MDI lies in the control through the toggle bit, i.e. a flying change of MDI positioning is not performed until a signal change has taken place at the toggle bit.

9.4.25 Control mode [825]

Mode 4 "control" allows pure speed control of the drive without position control. In "control" mode, the drive can be moved in jog mode with the fixed velocity levels 10 % and 100 % via a ramp-function generator (in a later software version the velocity levels will be adjustable with U511). The jog velocity is multiplied by the velocity override.

Control mode is suitable for commissioning (e.g. for optimizing the speed controller) and maintenance purposes, etc.



You will find detailed information on "Control mode" in the chapter of the same name in the Function Description of manual /1/.

The following graphic shows the sequence in control mode.

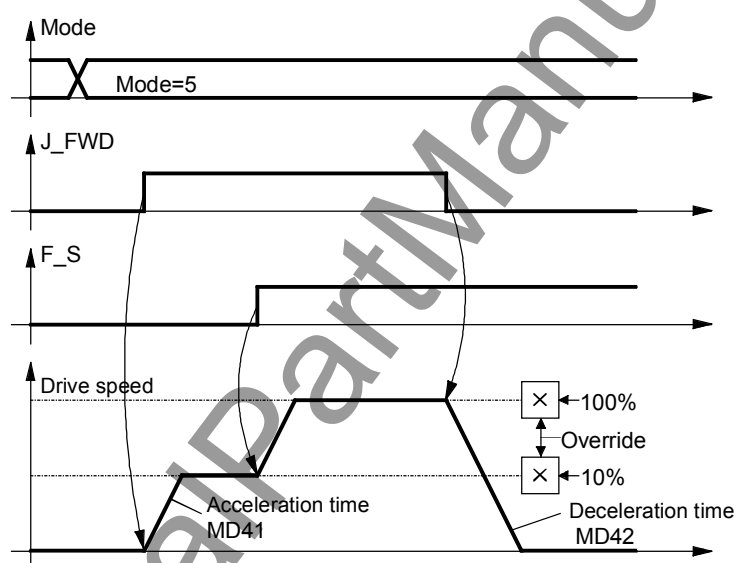


Fig. 9-28

CAUTION



Software limit switches MD12 and MD13 are not evaluated in control mode.

9.4.26 Automatic and automatic single-block mode [826, 828]



You will find detailed information on "Automatic mode" and "Automatic single-block mode" in the chapters of the same name in the Function Description of manual /1/.



The "Programming guide" chapter of manual /1/ describes how to write automatic NC programs for the automatic modes in a programming language conforming to DIN 66025.

Input of automatic programs via MASTERDRIVES parameters

Sheet [828] shows how to enter and edit automatic blocks step-by-step via MASTERDRIVES parameters U571 to U591 (see the parameter list for exact procedure).

9.4.27 Roll feed [830]

MD1 = 3 and MD11 > 0 activates the "roll feed" axis type, and the special block execution shown in [830] is valid for MDI, automatic and automatic single-block modes. The traversing curve can be adapted highly flexibly to the system conditions. In automatic mode you can start a new block on-the-fly with the "external block change" function, e.g. after detection of a printing index in order to cut printed material to length in an application where a printed image is to appear exactly in the middle of a packaging bag.

Loop counter

The loop counter enables the process of consecutive cutting to length of a selectable number of material pieces to be automated. The loop number can be set via the task interface of the S7 standard software GMC-BASIC /1/ or parameter U507. The loop count that has not yet been processed can be read at parameter n540.36.

9.4.28 Synchronization mode - overview [831]

Sheet [831] shows an overview of the synchronization functions, their interconnection, and the detailed representation of the functions on sheets [832...846] of the function diagram.



You will find detailed information on synchronization mode in the "Synchronization functions" chapter in the Function Description of manual /1/.

In the interest of the smallest possible deadtime differences, it is strongly recommended to use the virtual master axis as the master value source. An external master value generator ("real master axis", e.g. master pulse generator installed on the front section of the machine) should only be used in exceptional circumstances.

Synchronization mode includes the following functions:

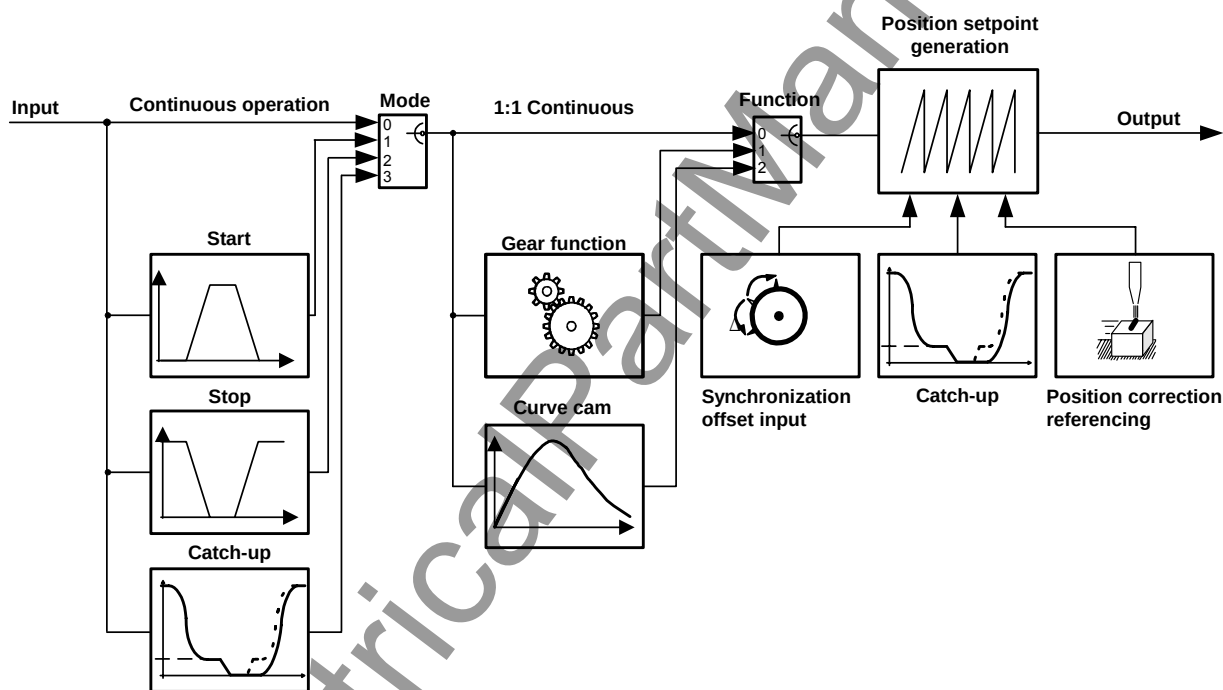


Fig. 9-29

CAUTION



By making an appropriate selection of master setpoint and synchronization parameters make sure that no inadvertent axis accelerations can occur.

Position setpoint jumps as a result of control intervention or parameter changes during running operation can occur both at the input and the output of synchronism.

Function	Use
Engaging cycle [834]	For drives that are normally stationary and are only started up for one operation (e.g. one machine cycle) in synchronization mode.
Disengaging cycle [834]	For drives that are normally used in synchronization mode and are only stopped for one operation (e.g. one machine cycle).
Gearbox function [835]	For drives requiring a transmission ratio between the master and slave axis.
Cam [839]	For drives whose sequence of movements is to be stored in a table.
Position correction [843]	A position correction can be superimposed on an angular synchronization. The position correction references the angular synchronization with external synchronization markers, e.g. pass marks.
Referencing [843]	Referencing "on the fly" to a reference index (e.g. BERO) during synchronization mode
Synchronization to master value [841]	Synchronization of the zero position of the slave axis to that of the master axis via a parameterizable compensation movement.
Displacement setting [841]	Setting of a random size of offset (displacement) angle as a fixed value or in inching mode (motorized potentiometer function)
Catch-up [837]	Coupling a drive up to and out of a synchronized drive system. The coupled-out drive can be autonomously operated at local speed and can be accurately halted at a specified position.

Definitions

The following section describes some important terms used in angular synchronization:

Master drive

The master provides the path setpoint for the synchronization block. There are two types of master: real and virtual.

With a **real master [833]**, the master position is detected by an encoder system, e. g. by a master pulse generator mounted on an upstream mechanical component. The measured position is the path setpoint for the synchronization block.

Advantage: The slave always follows the master.

Disadvantage: Load impacts and corrections affect the slave directly.

With a **virtual master**, an ideal position ramp is generated. This ramp is distributed to all drives. Even the master drive is synchronized with the virtual master.

Advantage: Synchronization has greater overall stability, since load impacts on the master drive no longer affect the slave drive.

Disadvantage: The master drive itself has to be synchronized.

Nesting the Synchronization Block

The virtual master [832] can be calculated on any MASTERDRIVES. Its output setpoints KK817 and KK816 [832.8] (path and velocity) are distributed over the SIMOLINK drive interface.

Calling up the synchronization block U953.33

The synchronization block is called up either as a free block or from the mode manager of the positioning system [802.8]. The differences are listed in the table below.

a) Calling from the mode manager of the positioning system

The mode selector [MODE_IN]=11 can be used to activate the synchronization as a "positioning mode" [809.4]. **This is the recommended method for activating synchronization.**

You can then change between positioning mode and synchronization mode. The synchronization block is called up from the mode manager of the positioning system, and the synchronization is calculated within the sampling time of the positioning modes set in U953.32. The value 20 must then be entered in parameter U953.33.

The positioning control signals are also used in this process, e.g. start command [STA] [809.4], and the corresponding checkback signals are also generated [809]. Following error monitoring is performed with reference to machine data MD15, in addition to software limit switch monitoring – for linear synchronization axes – with reference to MD12/MD13.



The "Synchronization mode" chapter in the Function Description of manual /1/ contains a detailed description of the control/checkback signals with timing charts for synchronization as a positioning mode.

b) Calling synchronization as a free block

If only synchronization [834...839], and not positioning, is required by the technology functions, it is possible to nest the synchronization like a free block in a sampling time. In this case, parameter U953.33 must be set < 20. The value $4 = 16 \cdot T_0$ (= 3.2 ms with a converter frequency of 5 kHz) is the preferred setting. The positioning modes must remain deactivated in this case with U953.32 = 20.

The use of synchronization as a free block is associated with the following advantages:

- ◆ As a result of deactivation of the mode manager, approx. 50 ... 100 µs less calculating time is required as the mode manager is not activated.
- ◆ The control sequences in the host machine controller can be simplified: it is no longer necessary to deal with the positioning control and status signals shown on sheets [809] and [810].

The disadvantages arise from the different activation methods used for synchronization and positioning and the absence of the following error and software limit monitoring (the latter can be useful with linear synchronization axes).

Differences : Synchronization as a mode ↔ as a free block		
	Synchronization as a Positioning Mode	Synchronization as a Free Block
Parameter settings for nesting in sampling time	U953.32 = 4 U953.33 = 20	U953.32 = 20 U953.33 = 4
Relevant machine data	MD11 MD49 MD12 *) Software limit switches ... MD13 *) ... for linear axis MD15 *) Following error monitoring - in motion MD23	MD11 Linear axis/rotary axis length [836.4] MD49 Precontrol - speed [836.7] MD23 (for precontrol)
Relevant positioning control signals from sheet [809]	STA] Start (0 → 1 edge must be activated after power on!) **) **) [MODE_IN] Mode selection	---
Relevant positioning status signals from sheet [811]	[ARFD] Axis referenced [FUR_M] Virtual master running [OTR] Overtravel (with linear axis) [FWD] Axis moves forwards [BWD] Axis moves backwards [MODE_OUT] Mode checkback signal [FUR] Function running [ST_EN] Start enable	---

*) The following warnings are triggered during synchronization as a positioning mode and the axis is brought to a standstill in speed control mode via the ramp configured in MD43:

A141 = Following error - in motion (MD15)

A195 = Overtravel negative (MD12)

A196 = Overtravel positive (MD11)

**) If the start command goes to "0" during travel, the axis is brought to a standstill via the ramp configured in MD42.

Nesting the synchronization in the basic unit

The nesting of the synchronization block is independent of whether the call was made as a free block or from the mode manager of the positioning system. In the following example circuit, position sensing is performed using the motor encoder.

CAUTION

Only the signals related to synchronization are shown.

NOTE

See section 9.4.41 "Continue synchronism".

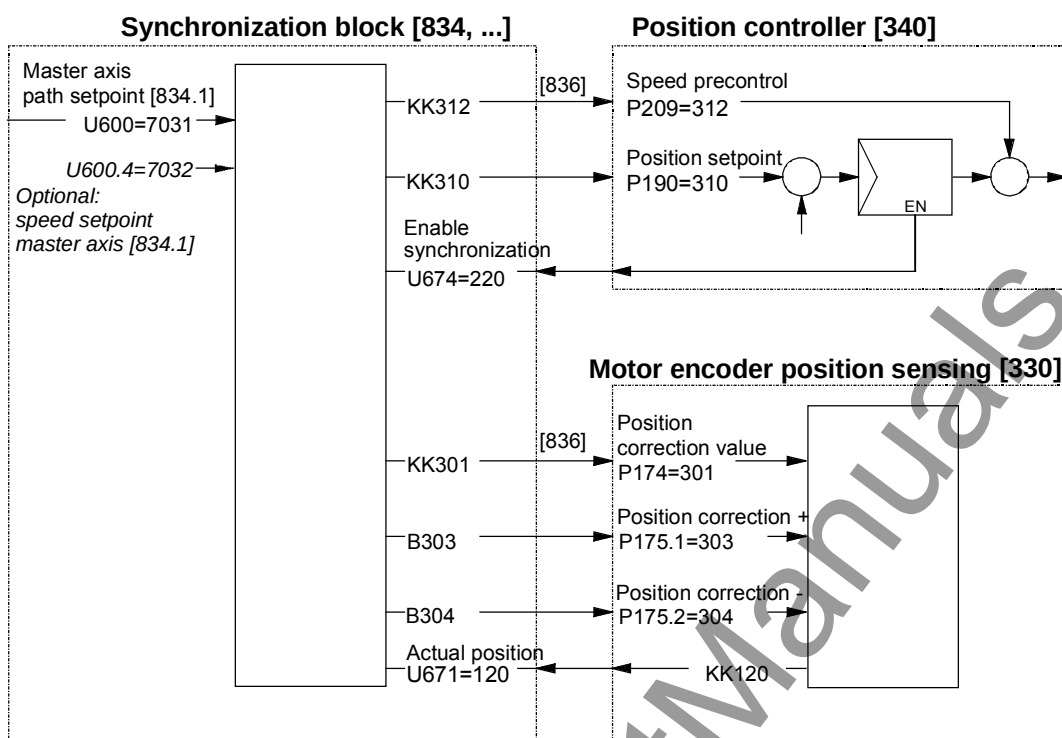


Fig. 9-30

Example

Synchronization of 3 drives with SIMOLINK.

The example shows the main application for synchronization via SIMOLINK. Drive 1 is the master drive with the virtual master axis. Drives 2 and 3 are synchronized with drive 1.

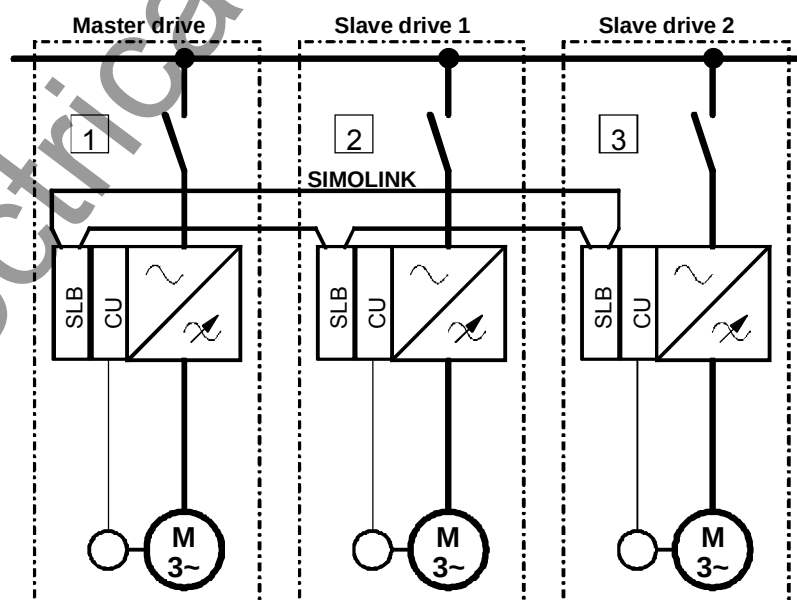


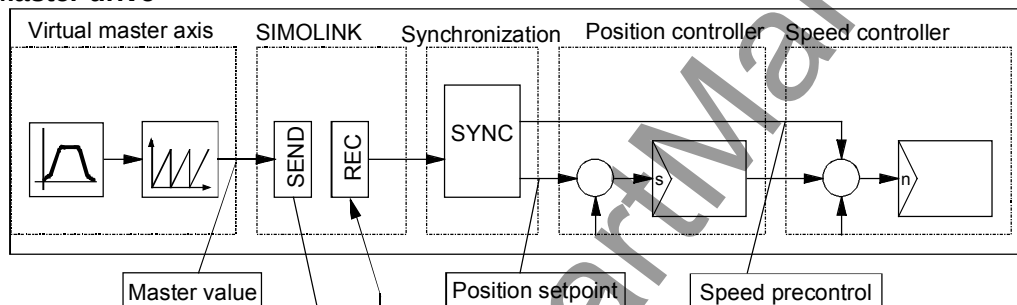
Fig. 9-31

The following rules must be followed during configuration:

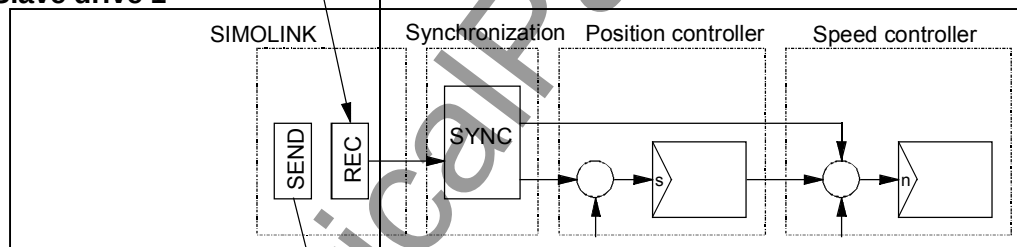
- ◆ One of the axes is defined as the master drive.
- ◆ The master drive must also be the SIMOLINK master (dispatcher). The module address is zero.
- ◆ The virtual master axis is enabled [832] on the master drive.
- ◆ All drives, including the master drive, move in synchronism with the virtual master axis [832].
- ◆ The output of the virtual master axis is wired to the SIMOLINK send block [160].
- ◆ The input of the synchronization block is connected to the receive block of the SIMOLINK, **including for the master drive**.

The graphic below illustrates the path of the master value of the virtual master axis, and the controller structure.

Master drive



Slave drive 1



Slave drive 2

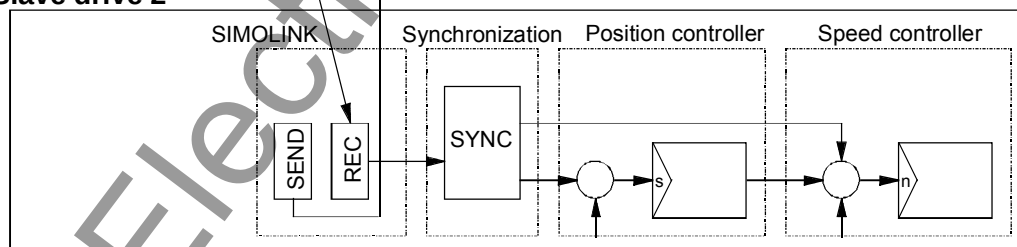


Fig. 9-32

Setting information for SIMOLINK [140...160]

The SIMOLINK cycle time in P746 should be set to the sampling time of the synchronization block, e.g. to 3.20 ms if the synchronization is nested in sampling time T4 with 5 kHz clock frequency (e.g. $U953.33 = 4$).

NOTE Start position for synchronization

If you want to start synchronization with a defined start position, you must first approach the position in a positioning mode and stop the drive at this point. You can then start the synchronization, commencing with speed "0".

The "offset setting" [841] also allows you to perform the orientation on-the-fly with reference to a synchronization marker after starting the synchronization mode.

Basic Settings of the Synchronization Block

The following section describes the settings relating to all synchronization functions.

Master path setpoint U600.01-03 / U606

Parameter U600 [834.1] can be set to predefine 3 sources as the master setpoint of the synchronization block. Parameter U606 can be used to select one of these three sources. These can be:

- ◆ The output of the virtual master axis
Output connector K817 [832] of the virtual master axis is wired to a SIMOLINK send word for the slave drives. The connection to the synchronization block should always be routed via the receive buffer of SIMOLINK, even for the master drive (e.g. KK7031 [150.7]) and not directly from the virtual master axis. You should not therefore use KK817. This ensures that the master drive receives its path setpoint from the virtual master axis at the same time as all the slave drives.
- ◆ Position sensing output as real master
For synchronization with a real master, the measured actual position is wired to the input of the synchronization block. The position can be transmitted from SIMOLINK or a position sensing system.

Master speed setpoint U600.04-06

It is possible to connect the speed setpoint as well as the position setpoint. Connecting the speed setpoint will increase the accuracy of the speed precontrol signal (KK312). If it is not connected, the speed is calculated internally from the position setpoint and the quality of the signal will depend on the set resolution. For this reason, the speed input should always be used for synchronized applications with high accuracy requirements. In such cases, it is important that the speed acting as the master setpoint of the synchronization block is produced as a percentage value [%] by the same setpoint source as the position setpoint in length units [LU].

For this purpose, it is imperative to parameterize the scaling rate master (U607.2).

Axis cycle AZL

Parameter U601 [834.2] must be set to zero for linear axes, i.e. for drives with an infinite traversing range.

In the case of rotary axes, the cycle length matches the product length (e.g. packaging machines).

When there is no fixed product length, e.g. on continuous rollers, the axis cycle can be freely defined, in which case it is normally set to match the positional difference corresponding to one motor or roller movement.

When the virtual master axis is used, its cycle length must be specified.

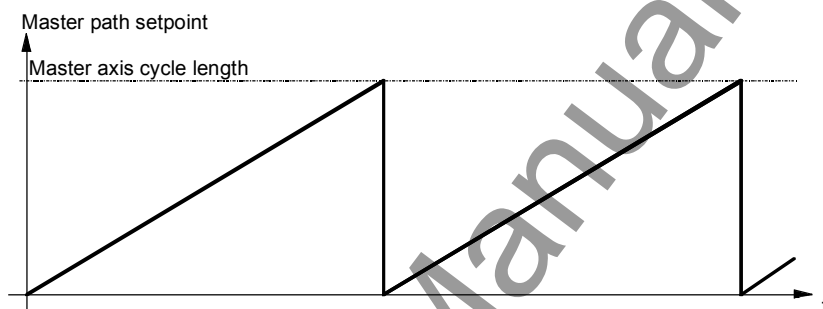


Fig. 9-33 Master axis cycle U601

Slave axis cycle U501.11 (MD11)

The above description for the master axis cycle applies analogously to the slave axis cycle [836.6]. The axis cycles for the master and slave can be set to different values.

Synchronization mode U602 U656

Synchronization mode [834.5] defines whether the synchronization block operates in

- ◆ Continuous cycle Value = 0
- ◆ Engaging cycle Value = 1
- ◆ Disengaging cycle Value = 2
- ◆ Uncoupler Value = 3

The cycle can be set by parameter or binector. Parameter U656 defines the binectors for the switchover.

If the catch-up is used here, it is imperative to parameterize the scaling rate master (U607.2). If the catch-up is selected as operating mode, the catch-up on FP 836.2 is inactive. It can only be used once (either FD 834 or FD 836).

Synchronization function U603 U657

The synchronization function [835.6] defines whether the synchronization block operates with

1:1 synchronization	Value = 0
Gear synchronization	Value = 1
Cam	Value = 2

The function can be set by parameter or binector. Parameter U657 defines the binectors for the switchover.

9.4.29 Virtual master axis [832]

You will find detailed information on the virtual master axis in "Configuring and testing the virtual master axis" in the section entitled "Commissioning the technology".

If the machine speed is to be specified as a percentage value (not in LU), it is recommended to use the comfort ramp-function generator in the free blocks [790], which produces very accurate speed and acceleration precontrol values (KK571 and KK572). From software release V1.3 and higher, a special position integrator is provided in the function diagram sheet [791] for implementing a virtual master axis using the comfort ramp-function generator. If these two function blocks are used connected in series, the virtual master axis shown on sheet [832] is no longer required.

Integrator for the virtual master axis using the comfort ramp-function generator

A special integrator is provided in the free blocks in function diagram sheet [791] for implementing a virtual master axis using the comfort ramp-function generator [790].

9.4.30 Real master with deadtime compensation [833]

Preferably, the virtual master axis should be used as the master value source for synchronization ([832] or [790]+[791]). In principle, this results in the most calm control behavior and the highest possible accuracy – also in dynamic operation – due to identical deadtimes for all axes during master value and position actual-value sensing.

However, it is often not possible to use a virtual master axis as the master value has to be read in via an external master value generator attached to an upstream machinery component, which already exists (motor encoder or built-on encoder).

In this case, the "Real master with deadtime compensation" function block is used – both on the drive in which the master value generator is evaluated and in the downstream drives to which this master value is forwarded via SIMOLINK.

First of all, the input position value from the position encoder (or from SIMOLINK) is limited to the axis cycle length, which can be set via U425. The position actual value normally reaches the position controller via the signal path of the synchronization block later than the position actual value of the dedicated axis generated directly in the fast position controller cycle. The deadtime resulting in this case is normally particularly large for such drives, which receive this master value via SIMOLINK. The deadtime compensation U424 ensures that this deadtime is compensated for by adding a corresponding "advance distance lead" to the master value. The distance lead is speed-dependent: the higher the speed, the greater the distance the material covers within the deadtime.

The speed value on which this is based can be gained by differentiating from the master value or can be directly picked up from the position sensing of the external master value generator, whereby preference is given to the latter signal. An unsmooth speed signal can be smoothed via U427, whereby the set smoothing time constant results in a higher deadtime to be compensated.

For this purpose, the drive is accelerated to two different speeds. The positional change is calculated through measurement of zero pulses or printing index.

$$s_{\text{Positionalchange}} = s_2 \text{ _____} - s_1 \text{ _____} = \text{_____} [\text{LU}]$$

$$v_{\text{Change}} = v_2 \text{ _____} - v_1 \text{ _____} = \text{_____} \text{ LU/min}$$

$$\rightarrow t_{\text{Deadtime}} = \frac{s_{\text{Positionalchange}} [\text{LU}]}{v_{\text{Change}} [\text{LU/ms}]} - 1 = \text{_____} [\text{ms}]$$

9.4.31 Engaging/disengaging cycle [834]



You will find detailed information on the engaging/disengaging cycle in the "Synchronization functions" chapter of the Function Description in manual /1/.

In the "Application areas" section you will find an example application for the engaging/disengaging cycle. The following section provides a brief overview of the function.

The engaging/disengaging cycle is similar to the engaging/disengaging of a mechanical coupling at an accurately defined position. The graphic below illustrates the sequence of an engaging or disengaging cycle [834].

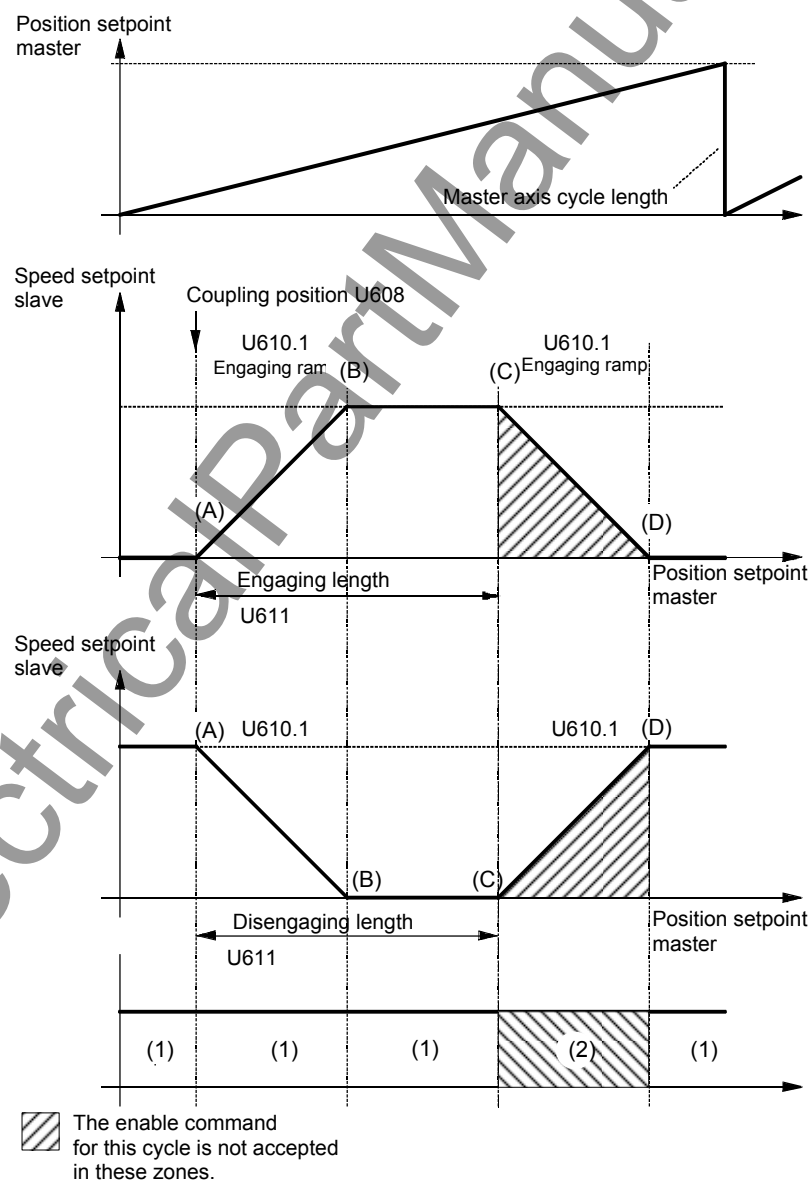


Fig. 9-34

If the engaging/disengaging cycle is enabled in the (1) zones, the engaging/disengaging cycle is started the next time the coupling position is crossed.

The operating principle described below for the engaging cycle (U475=0 default configuration) also applies to the disengaging cycle.

When coupling position (A) is crossed, the drive accelerates via the ramp. Synchronism with the master is reached at point (B), by which time the master has traveled half of the engaging/disengaging ramp length configured in U610.1 [834a.4]. At point (C), the drive starts its deceleration ramp, finishing at point (D).

In zones (A) to (D), the slave has covered the "on" length.

Engaging/ disengaging cycle enable U612

Enabling of the engaging/disengaging cycle is either edge-triggered or initiated by a static signal. The source of the enable signal can be selected with U612.01 (static signal) or U612.02 (once-off enabling by edge triggering) [834a.2].

Static engaging/ disengaging cycle enable U612.1

With the static enable (continuous enable), the engaging/disengaging cycle continues to operate as long as the signal is active.

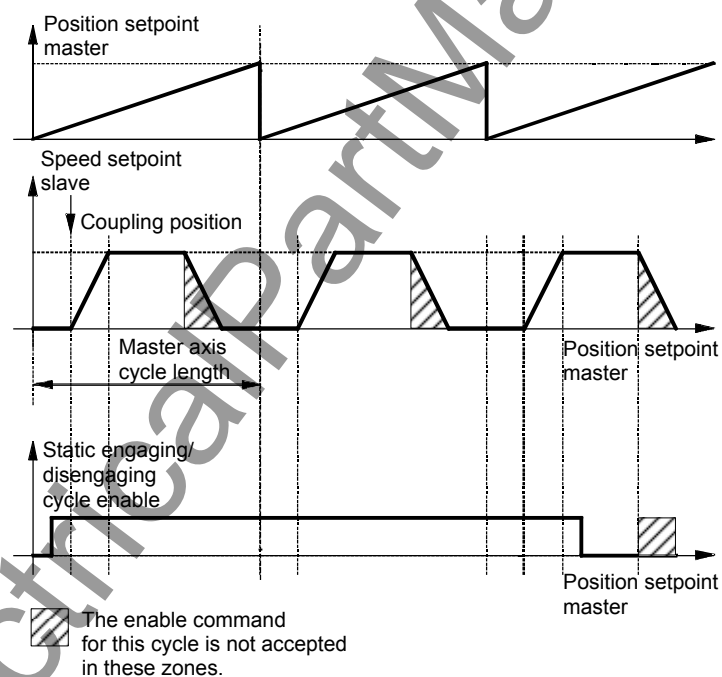


Fig. 9-35 Example: engaging cycle for rotary axis

If the enable is initiated in the shaded areas, it is no longer accepted for this cycle.

Exceptions

If the engaging length is greater or equal to the master axis cycle length, the drive switches to constant synchronization after it has crossed the coupling position with a continuous enable signal applied.

Example:

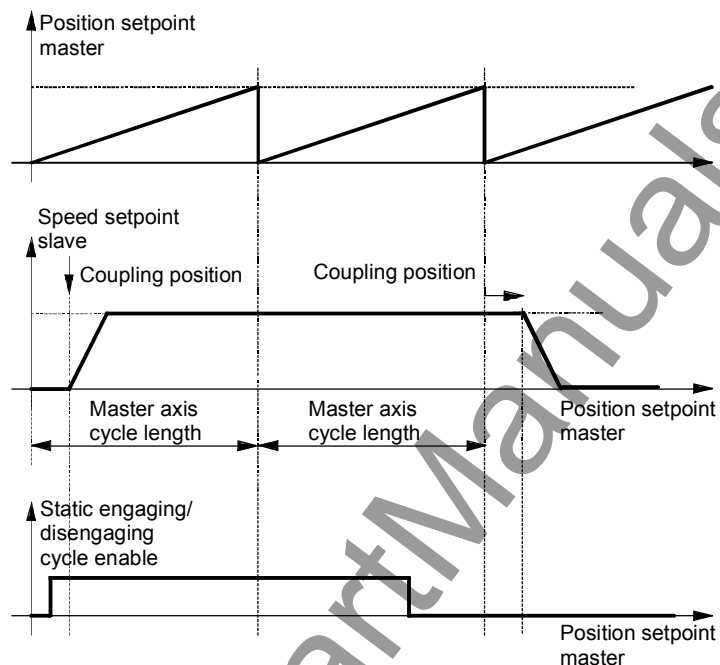


Fig. 9-36

Reversing with the engaging/disengaging cycle:
 The coupling position initiates the engaging process again.
 Exception: If the master setpoint is reversed during engagement, the engaging cycle will stop again at the coupling position.

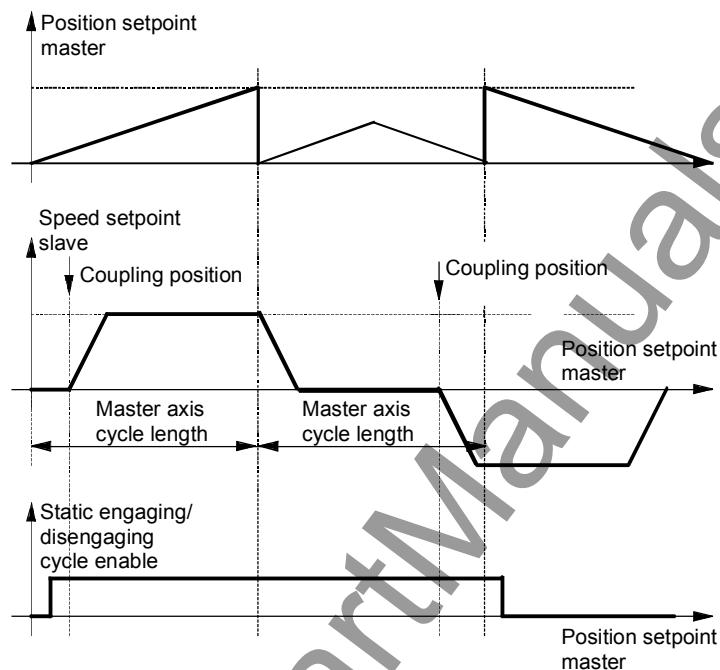


Fig. 9-37

Engaging/ disengaging cycle once-off enable U612.2

A positive edge of the once-off enable signal enables the engaging/disengaging cycle for one operation.

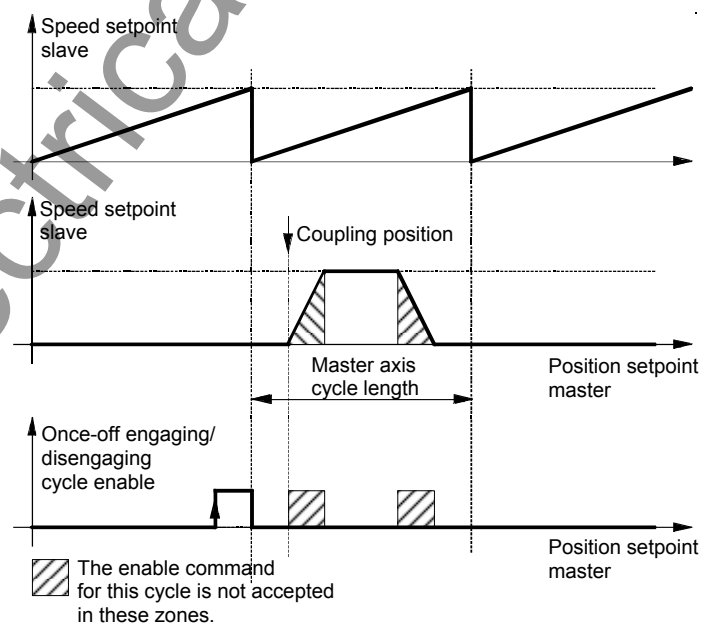


Fig. 9-38

Retriggering

If an edge is applied to the enabling input outside the shaded areas, the engaging/disengaging cycle is retriggered for another run.

If the engaging cycle is retriggered within the permissible time period, it operates as if statically enabled for the engaging length according to the number of times the retrigger edge is applied.

Example:

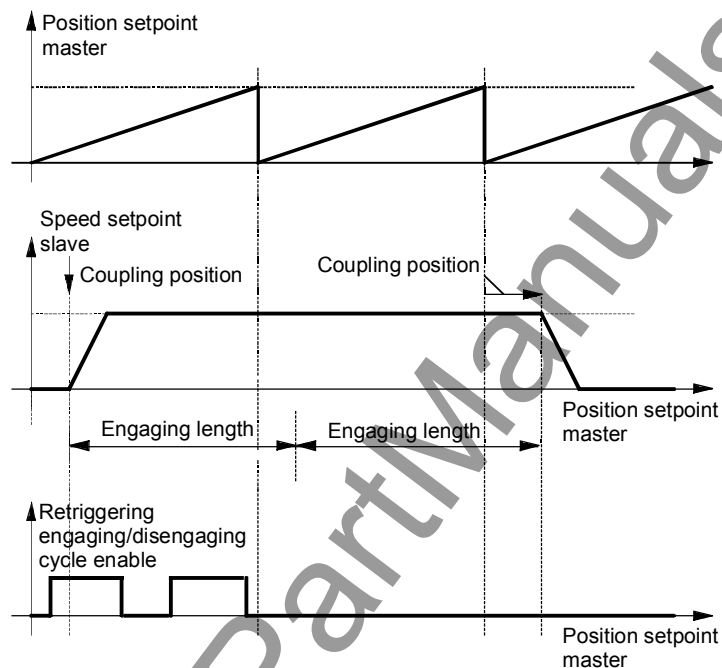


Fig. 9-39

**Continuous enable
for engage-/
disengage cycle
active permanently
U612.3**

If the continuous enable signal is permanently active, the engage/disengage cycle is executed only once when the coupling position is first reached. The drive then remains either engaged or disengaged.

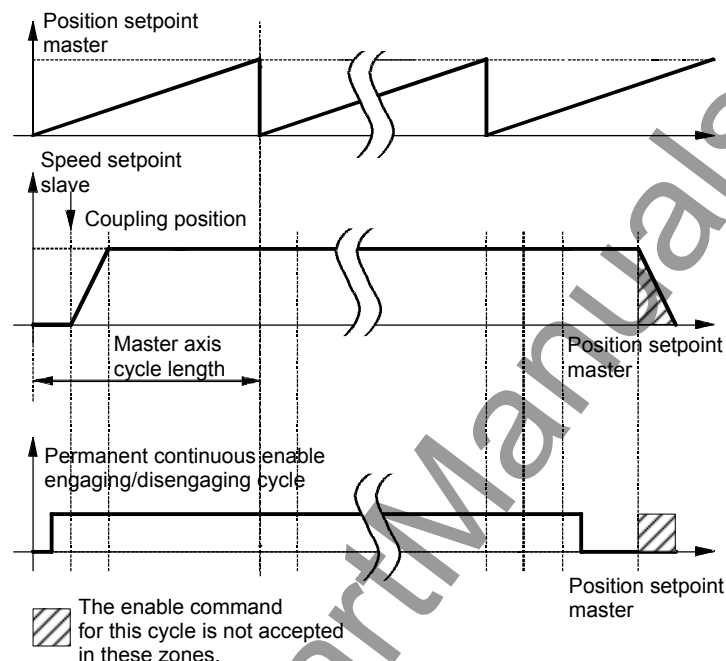


Fig. 9-40 Example: engaging cycle for rotary axis permanent continuous enable

9.4.32 Gearbox function [835]

The gearbox function [835] allows a transmission ratio to be set between the master and slave axis. The transmission ratio is a fraction.

$$\text{Transmission ratio} = i = \frac{\text{Numerator}}{\text{Denominator}}$$

Example: $i = \frac{1}{2}$: U604.1 = 1, U604.2 = 2

The slave axis traverses at half the velocity of the master axis.

The transmission ratio can also be changed in motion. If you want to avoid jumps in the transmission ratio, you can control the transmission ratio (numerator or denominator) using the simple ramp-function generator [791] in the free blocks.



You will find detailed information on the gearbox function in the "Synchronization functions" chapter of the Function Description in manual /1/.

9.4.33 Generation of the position setpoint [836]

Before the position setpoint is output, the signals from synchronization and displacement (offset) angle adjustment (V_displacement, [841]) and of catch-up [837] are switched in. As an alternative, the position setpoint can be supplied from an external source via U886 (activated via U885). The resulting speed setpoint is integrated up to the slave position setpoint in the "ACL integrator" with limitation in the case of a rotary axis to the slave axis cycle length parameterizable via MD11. The corresponding corrective actions are also initiated for the position actual value via KK301 and B303/B304.

A speed precontrol value is available at KK312. It can be switched in downstream of the position controller to reduce the dynamic tracking error.

9.4.34 Catch-up function [837]

The catch-up function enables a drive to be coupled up to and uncoupled from a synchronization system. The uncoupled drive can be operated autonomously at local speed and can be halted at an accurate position.

Uncoupling an axis

Using the "Uncouple/Stop" = 1 command it is possible to decouple an axis from a synchronization system. The axis then reduces its speed via a ramp to the "catch-up setpoint speed", which can be preset via U626.01 in the unit [10 LU/min] or via U626.02 as a percentage. The deceleration of the deceleration ramp can be set in U628.1, and the rounding of the same in U627.1. It is possible here to directly influence the internal ramp or rounding via the "Mode" command, with and without internal ramp, via any setpoint source without applying the internal rounding.

Halting an axis at a defined position

The "Enable positioning" command can be used to stop the axis at a setpoint position that can be parameterized in U626.03. However, the drive initially travels at "catch-up setpoint speed" until the halt position can be approached with the ramp, which is parameterizable in U628.3 without having to change the sense of rotation. When the factory setting value -1 is selected, the ramp of index 1 is applied.

If the "Enable positioning" command is canceled, the axis can be made to exit the halt position again and to accelerate up to "catch-up setpoint speed" by setting the acceleration in U628.4 (see the broken-line acceleration curve in [837]). In this instance as well, a factory setting value of -1 causes the ramp of index 2 to be applied.

Furthermore, the "Trigger positioning" command can be used to start the position controller and perform a new positioning operation.

The shutdown position is then approached either in "relative mode" within one revolution (axis cycle compensated) or in "absolute mode" via several axis cycles.

Coupling up an axis By canceling the "Uncouple/Stop" command, it is possible to recouple a stationary axis or axis operating with "Setpoint speed uncoupler" into a synchronization system. The axis accelerates to the machine speed set by the master via an acceleration ramp. The acceleration of this ramp can be set at U628.2, and the rounding thereof at U627.2.

Once speed synchronization has been achieved, the binector B820 "Catch-up finished" goes to "1". This binector is usually controlled with the "Synchronize to master value" input [841.2] in order to now create accurate angular synchronization with the master.

9.4.35 Cam [839]



You will find detailed information on the electronic cam (table synchronization) in the "Synchronization functions" chapter of the Function Description in manual /1/.

The cam [839] allows free assignment of master and slave positions. This allows you to define the sequence of movements on the slave axis with reference to the master.

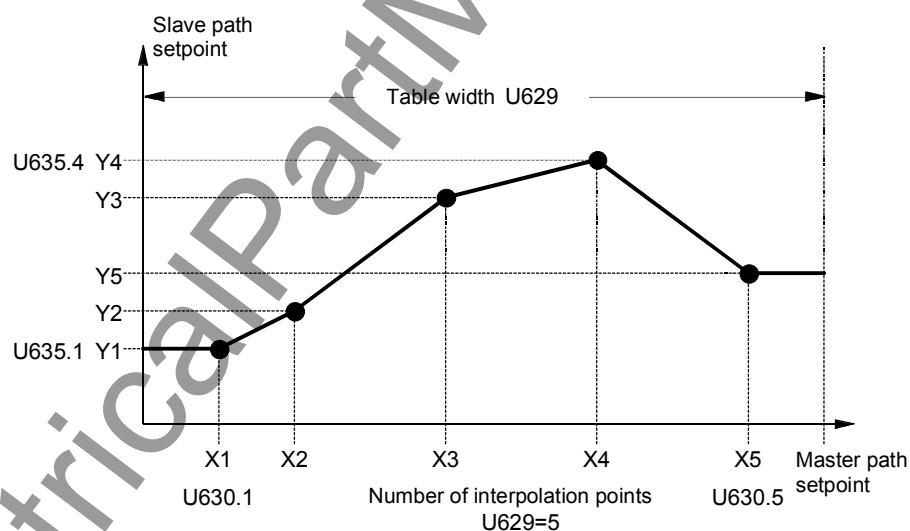
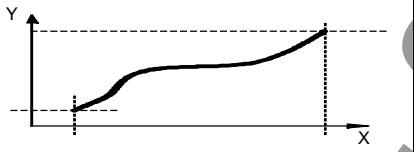
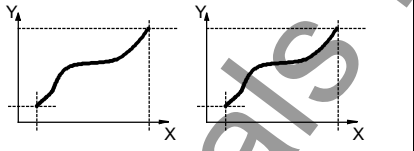
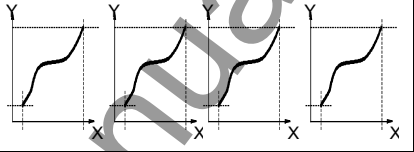
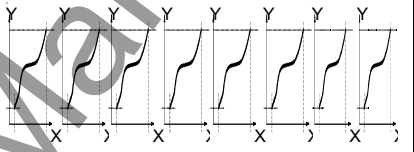
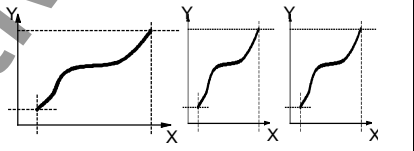


Fig. 9-41 Example of a cam with 5 interpolation points

Linear interpolation is performed between the interpolation points, i.e. a straight line is generated. Y1 is output for values less than X1 (horizontal movement to zero), Y5 is output for values greater than X5 (horizontal movement up to the table width).

**Table configuration
U615**

A total of 400 table interpolation points can be defined [839.6]. These can be used in one large table or several small ones.

U615 = 0	One table with max. 400 interpolation points	
U615 = 1	Two tables with max. 200 interpolation points each	
U615 = 2	Four tables with max. 100 interpolation points each	
U615 = 3	Eight tables with max. 50 interpolation points each	
U615 = 4	Variable involving a maximum of eight tables with a total of 400 interpolation points	

CAUTION

Note E²PROM capacity, not all interpolation values will be stored!

Owing to the different sizes of E²PROM, not all nine interpolation values will be stored in the E²PROM.

The large E²PROM is capable of storing all interpolation value parameters.

The small E²PROM can store only the parameters of interpolation values that were available in versions <1.4x, but the parameters of new values are stored only in the RAM.

Variable table configuration (U615=4)

The size and number of tables can be configured variably with this option.

A maximum of 8 tables for a total of 400 interpolation values are available.

You are not then limited to the fixed table configurations with 50, 100, 200 or 400 interpolation points with which you must use either 1,2,4,8 tables. With this option, for example, you can configure 5 tables with 80 interpolation points, or 3 tables with 200 interpolation points stored in one and 100 points in each of the other two.

You can allocate the number of interpolation points to be stored in each table

Number of interpolation points: U629.1 to U629.8 for tables 1-8

It is possible to keep track of the number of interpolation points still available in the visualization parameter.

Number of available interpolation points: n634 (1....400)

NOTE

The tables must then be stored one after the other with no gaps!

The tables are no longer stored in parameters in fixed increments of 50 points, but the Table Info viewing parameter provides a useful guide to their configuration since it is automatically calculated after the interpolation points for the tables have been entered.

Table Info displays the initial and end parameters for each table.

Meaning of Table Info (n639.x):

X	H	Z	E
No meaning	1 = U630 2 = U631 3 = U640 4 = U641 5 = U632 6 = U633 7 = U642 8 = U643	Index 1 to 50	

Example: 5 tables with 80 interpolation points:

Indices 1 to 5 in "Number of interpolation points" are set to 80.

Table Info now displays the allocation of interpolation points to individual tables:

Table begins			Table ends		
Table Info		First interpolation point in parameter	Table Info		Last interpolation point in parameter
n639.01	101	U630.01	n639.02	230	U631.30
n639.03	231	U631.31	n639.04	610	U633.10
n639.05	611	U633.11	n639.06	340	U640.40
n639.07	341	U640.41	n639.08	720	U642.20
n639.09	721	U642.21	n639.10	850	U643.50

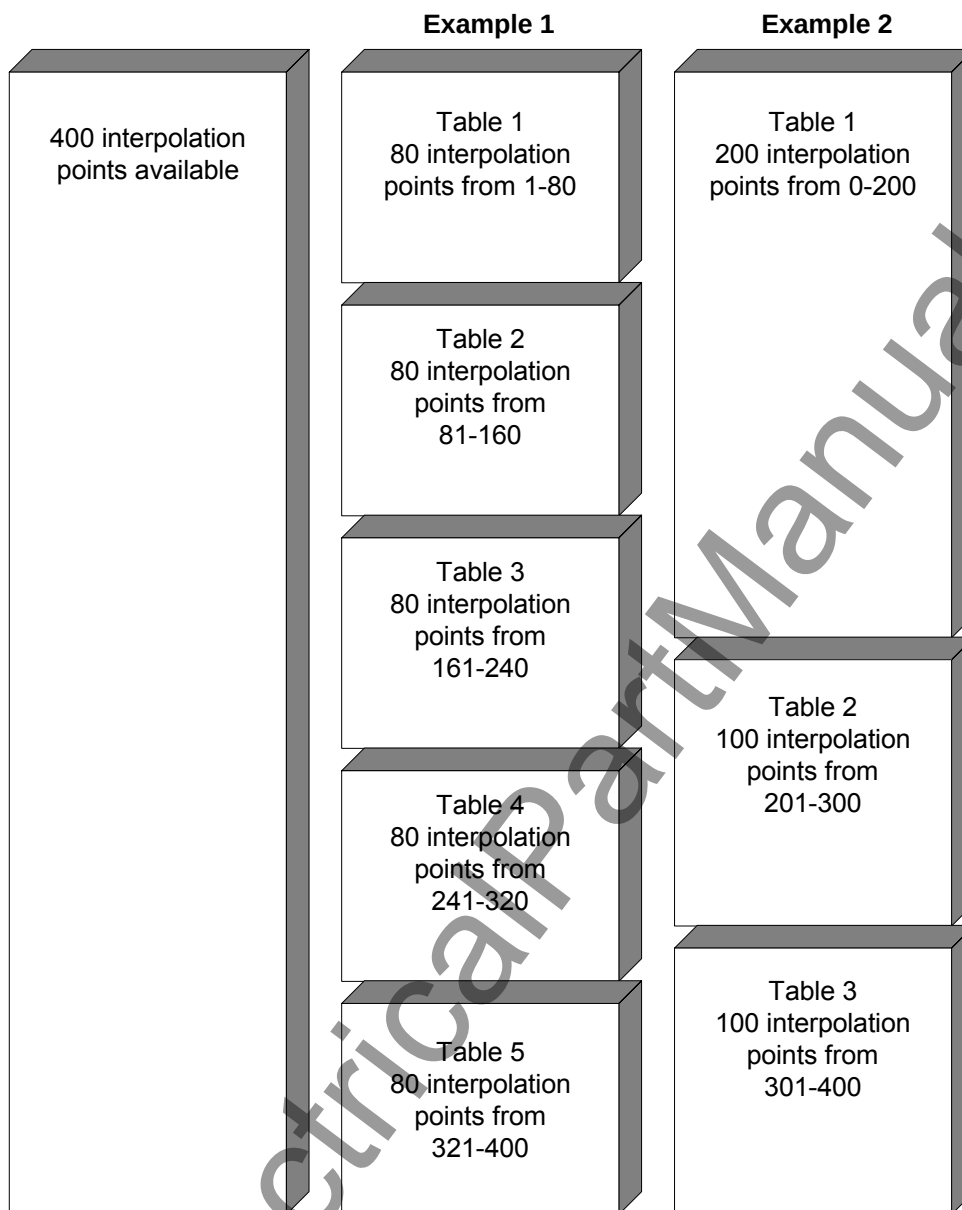
Example of variable point allocation among tables:

Table input / table check

The following sequence must be followed when entering the cam:

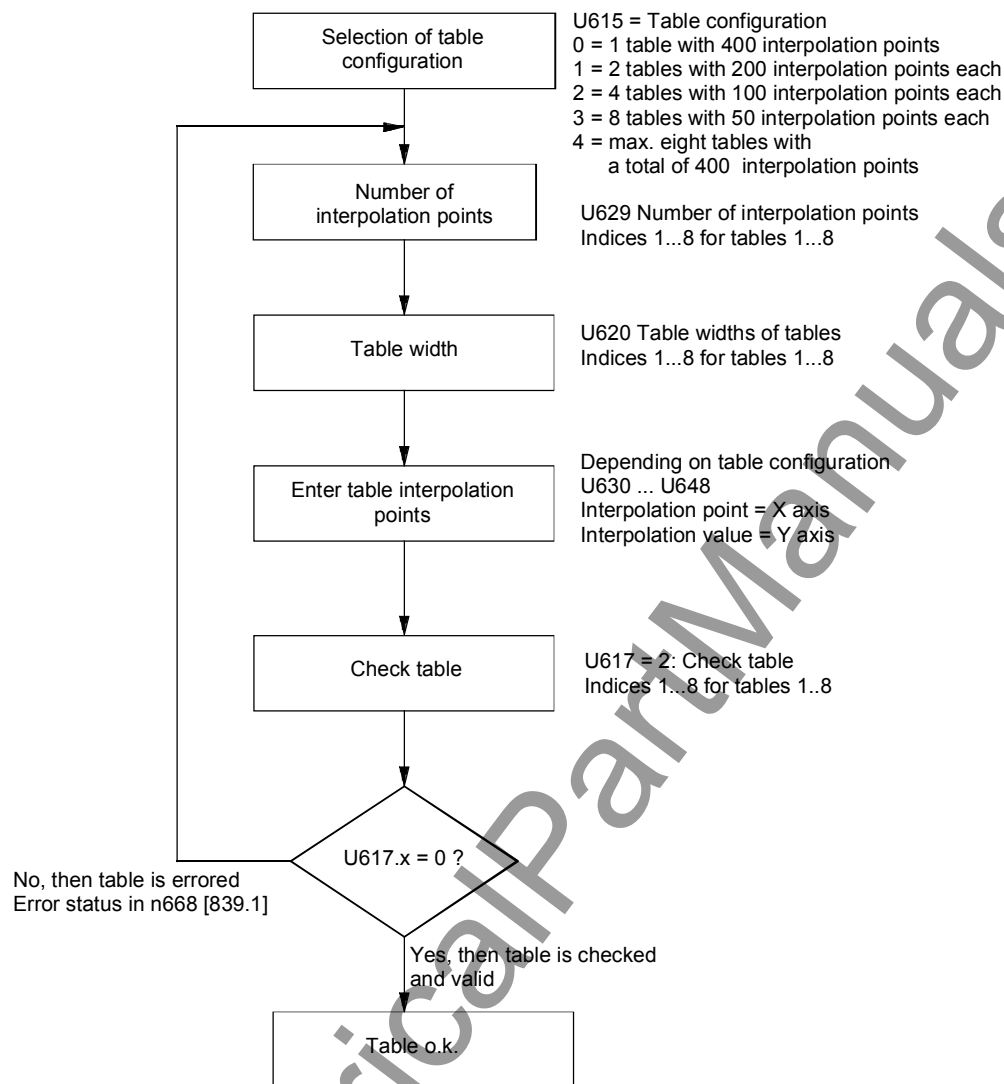


Fig. 9-42

NOTE

The interpolation points (x coordinates) must be defined in ascending order.

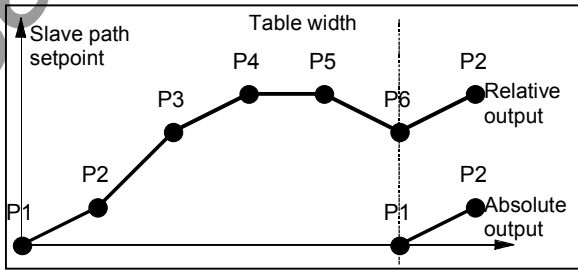
Only interpolation points within the range from 0 to the table width are allowed.

Interlocks in tables:

An active table cannot be changed at all. Except for table width and number of interpolation points, an inactive table can be changed, checked and accepted as a background function. The operating mode must otherwise be switched over to 1:1 or gearbox.

Table interpolation modes U616, U614

The following table interpolation modes can be defined [839.5]:

U616 = 0xxx	Without scaling of y axis: The y coordinates are output 1:1.
U616 = 1xxx	With scaling of y axis: The y coordinates are multiplied by the scaling factor of the y axis. This is composed of the ratio of U651.1 (numerator) and U651.2 (denominator).
U616 = x0xx	Without scaling of x axis: The direct input value is the x coordinate value of the cam.
U616 = x1xx	With scaling of x axis: The input value of the table is multiplied initially by the scaling factor of the x axis. This is composed of the ratio of U623.1 (numerator) and U623.2 (denominator). The scaling of the x axis has the same effect as a gearbox upcircuit of the cam.
U616 = xx0x	Continuous output: With continuous output, there is a return jump to the start of the table when the end of the table is crossed (rotary axis).
U616 = xx1x	Stop at end of table: In this mode, the output value is frozen at the last interpolation point when the end of the table is crossed. The return jump to the start of the table takes place after external synchronization through the binary "table synchronization" signal.
U616 = xxx0	Absolute table output: The absolute interpolation point is output on the return jump to the start of the table. If the interpolation point at the end of the table is not equal to the interpolation point at the start of the table, a jump takes place.
U616 = xxx1	Relative table output On the return jump to the start of the table, the table continues with reference to the last interpolation point. Example 
U616 = xxx2	Table change relative (without a jump), otherwise absolute table output functionality (U616 = xxx0).
U616 = xxx3	Table change relative (without a jump), otherwise relative table output functionality (U616 = xxx1).
U614 = 1	Accept Scaling 0 = Scaling is active all the time, a change to the scaling causes a jump. 1 = Scaling is activated in response to positive edge of binector U621 SYNT or a table overflow (return jump to start of table), the jump is initiated by the user or the end of the table.

9.4.36 Synchronization to master value [841]

With "Synchronization to master value", the zero position of the slave axis is synchronized once to the zero position of the master axis via a parameterizable compensation movement.

A 0 ==> 1 edge of the "Synchronize to master value" command triggers synchronization. The currently present master value is converted once over the entire synchronization path. After that, the position difference $\Delta s_{\text{Master\_Slave}}$ to be corrected between the master and the slave is calculated from the thus determined master position setpoint, the current displacement setpoint (KK812 [841.8]) and the currently applicable slave position setpoint. In order to compensate for this, the axis performs a compensation movement [841.7] with an adjustable differential speed and acceleration (U691.1 and .2). The integral of the travel curve $v = f(t)$ traversed here corresponds to the position difference that has to be corrected.

By taking the "Current displacement" into account in the calculation (841.7 ==> 841.2) it is ensured that a slave axis displacement (offset) already implemented by the displacement angle settings will remain.

The total performed position correction has the amount

$$\text{Position correction} = \text{master value} - \text{position setpoint slave} + \text{displacement}$$

i.e.

$$\Delta s_{\text{Master\_Slave}} [841.5] = s_{\text{Master}} [834.3] - s_{\text{set\_Slave}} [836.6] + \text{Current displacement} [841.8]$$

You can select via operating mode U699.1 whether the compensation movement of the axis is to take place in positive direction, in negative direction or via the shortest route (e.g. correction from 350° to 10° shall be effected by 20° forwards, not by 340° backwards).

You can also synchronize the two axes within a window.

Synchronization within window 1 is performed via the shortest route for small, quick compensatory motions. Synchronization outside window 1, but within window 2, is performed in the parameterized direction. The axes cannot be synchronized outside window 2. The applicable synchronization mode is then checked back via binector.



Detailed information about synchronization can be found in Chapter "Synchronization functions" in the Function Description of the Manual /1/.

9.4.37 Displacement angle setting [841]

Absolute displacement angle

The position of the slave axis can be corrected by a displacement value through the absolute displacement angle. This value can be preset by parameter U677 or connector (U678.01). This displacement angle acts absolutely i.e. it resets all displacement or offset movements that have been previously effected by other displacement angle settings and which have accumulated in the "Current displacement" signal [841.8].

The absolute displacement is set once in each case when the value of the "absolute displacement angle" is changed via the "Compensation movement" function [841.7] with adjustable differential speed and acceleration. In the startup process, the displacement angle is set to 0. The first change at the connector input results in a new setting of the displacement angle.

You can select via operating mode U699.2 whether the compensation movement of the axis is to take place in the specified direction (i.e. clockwise if displacement is changed to a higher value, and counter-clockwise if displacement is changed to a lower value), or via the shortest route (e.g. correction from 350° to 10°, by 20° clockwise, not by 340° counter-clockwise).

Relative displacement angle

The currently valid displacement angle can be changed via the relative displacement angle (U678.3) by the specified value. This is activated by two binectors for adjusting in the positive direction (U694.1) and in the negative direction (U694.2). The change is adopted at these control inputs with each positive edge.

The relative displacement angle $\Delta s_{\text{relative}}$ can be greater than the parameterized slave axis length.

The displacement is set via the "Compensation movement [841.7] with an adjustable ramp and differential speed.

Inching – displacement angle

The current displacement angle can be changed via the two binectors Inching+ (U696.1) and Inching– (U696.2). The variable speed and acceleration can be set via U695.2 and .3. Adjustments can be made as long as one of the two inputs is activated. No adjustment is made if both inputs are activated simultaneously.

Non-volatile storage of the displacement angle

The resulting displacement angle is output as "modulo axis cycle length" at connector KK812, i.e. the displacement angle refers to an axis cycle. The displacement angle can be assigned to a tracking/storage element [760] for data remanence and, once the MASTERDRIVES electronics power supply has been restored, can be stored again as a set value by the tracking/storage element.

9.4.38 Position correction [843]

Position correction enables synchronizing signals e.g. BERO proximity switches or printing indices of optical reading devices, to be cyclically evaluated during synchronization. The printing index is recorded by a fast, interrupt-capable MASTERDRIVES input and the actual position at the time of the interrupt is stored by the position sensing. If the setpoint position stored in the MASTERDRIVES at the time of the printing index does not correspond to the measured actual position, an automatic compensation movement takes place at a speed specified in U667 by means of which this deviation can be corrected.

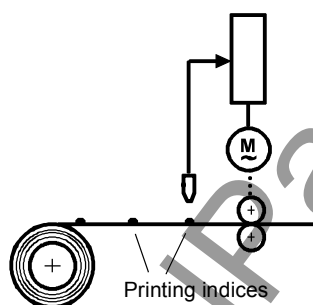
Section 9.3.3 includes an application example for position correction under the section "Printing index control".

A position correction is normally automatically started by the "Start position correction" command when a new position measured value (i.e. a measured actual position of the printing index) has been made available by position sensing.

The following two operating modes can be set via U661:

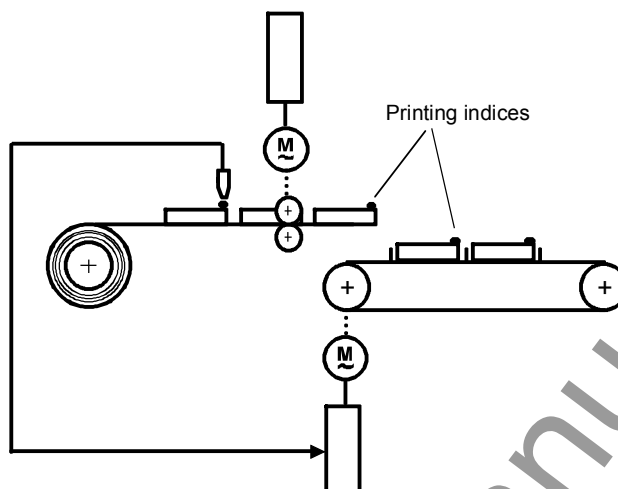
Operating mode 1

The axis transports the printing index:



If the printing index comes too late, short-time acceleration has to take place to make up the position delay again. Beforehand, the position setpoint and actual value are set in the counter-direction in order to create the correct reference to the mechanical system.

Operating mode 0 The axis does not transport the printing index:



The drive being taken into consideration is normally arranged behind the drive transporting the material with the printing index (and not itself evaluating the printing index). If the printing index comes too late, the drive must brake briefly in order to "wait" for the printing index.



You will find detailed information on position correction in the "Synchronization functions" chapter of the Function Description in manual /1/.

9.4.39 Referencing "on the fly" for synchronization [843]

The referencing "on the fly" function enables synchronization to be made "on the fly" during startup to a reference index (BERO or similar) in synchronization mode.

It is no longer necessary to approach the reference index beforehand in positioning mode and then to change over into synchronization mode from standstill.

The function is enabled via a binector (U675.2). At a positive edge of the enable signal, the output binector B808 "Referencing running" is reset. As long as the enable is active (=1), new referencing takes place with each recorded reference index. The index is recorded with the "Start position correction" input (U666).

Usually the "Start position correction" input is always made when the interrupt triggered by the reference index has resulted in a new valid position measured value.

When the reference index has been recognized, both the position actual value and the position setpoint are set to the reference position. No compensation movement takes place.

9.4.40 Docking point between basic positioner [789b] and synchronism [836]

Application example The infeed point U886 makes it possible to link the "Basic positioner" function block to technology option F01 in synchronous mode. The link is activated via U885.

Parameter	Internal implementation	Index	Meaning	
Src SYNC local ON	U885 Parameter (binector)	1	"0"	Setpoint source for synchronous branch is offset angle setting / catch-up
			"1"	Connect setpoint input U886 into synchronous branch
Src SYNC local	U886 Parameter (double word connector)	1	Setpoint input for position setpoint in [LU	
		2	Setpoint input for speed setpoint in [%]	

U885 has higher priority than U837.8 ("Catch-up active"), i.e. when U885==1 the infeed point U886 is activated regardless of the state of U837.8.

The speed setpoint at U886.2 has the same standardization as the value at U461.2 (100 % == 0x4000 0000).

The input for the position setpoint is compensated according to the axis cycle.

When the synchronism setpoint source is switched from catch-up to the external infeed at U886 (0 → 1 edge change at U885), the drive is synchronized once with the position setpoint at U886.1 for the sake of preventing a setpoint step change. On-the-fly transitions during other changeovers in the new operating mode are not implemented. The user must make external interconnections using free blocks to control the occurrence of setpoint step changes (e.g. with a 1 → 0 edge change at U885).

NOTE

Fault-free interoperation between the basic positioner and synchronous blocks is ensured only if the following conditions are met:

1. The free blocks "Basic positioner" and "Synchronism" must both be computed in the same time slot. To ensure this condition is met, the user must make the appropriate assignments between the free blocks and the time slots (parameter settings).
2. The drive must be stopped via the catch-up function before U886 is connected in.

Undesirable effects (setpoint step changes, oscillations, etc.) may otherwise occur.

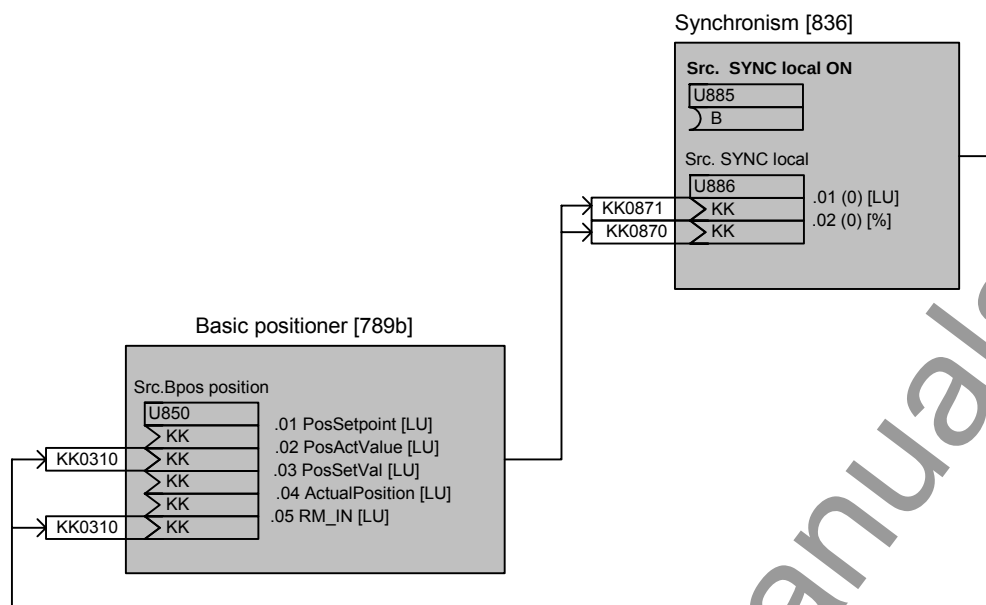


Fig. 9-43 Application example

```

REM *****
REM ***      Link Bpos to synchronous block      ***
REM *****

REM *****
REM *****
REM Analog Input NC
REM Analog Output NC
REM Binary Input1 EPOS; Pos
REM Binary Input2 NC
REM Binary Input3 Stop
REM Binary Input4 Positioning absolute/relative
REM Binary Input5 NC
REM Binary Input6 Synchronism local ON / EPOS enable
REM *****

```

...

MSG Virtual Master Axis ;

WRITE 630 00 1.00 ;AI scaling
 WRITE 634 00 100.0 ;Smoothing AI
 WRITE 640 01 11 ;AO from AI
 WRITE 2951 51 4 ;
 WRITE 2961 51 3347 ;CRG_KK571
 WRITE 2320 00 11 ;Ramp-function generator input
 WRITE 2321 00 0 ;Stop ramp-function generator
 WRITE 2322 00 0 ;Halt ramp-function generator
 WRITE 2324 00 0 ;Set CRFG
 WRITE 2327 00 1 ;Rounding mode
 WRITE 2328 00 0 ;Bridge CRFG
 WRITE 2329 00 1 ;Adaptation acceleration/deceleration times

 WRITE 2330 01 20.0 ;Acceleration time
 WRITE 2331 01 0 ;in sec
 WRITE 2332 01 20.0 ;Deceleration time
 WRITE 2333 01 0 ;in sec
 WRITE 2334 01 0.50 ;Initial rounding
 WRITE 2335 01 0.50 ;Final rounding
 WRITE 2337 00 5.0 ;Fast stop time
 WRITE 2338 00 0 ;Fast stop from BI signal
 WRITE 2342 00 100.00 ;Output limitation
 WRITE 2343 00 573 ;Output limitation positive
 WRITE 2344 00 574 ;Output limitation negative
 WRITE 2953 35 4 ;
 WRITE 2963 35 3348 ;VM_KK610
 WRITE 2429 01 571 ;Input value VM
 WRITE 2429 02 0 ;Setting value VM
 WRITE 2430 00 524288 ;Axis cycle length VM
 WRITE 2431 00 1048576.00 ;Rated master speed
 WRITE 2432 00 0 ;Set VM
 WRITE 2953 20 4 ;
 WRITE 2963 20 3349 ;LC_K255

MSG Synchronous Operation ;

WRITE 2529 00 70 ;Actual position ok
 WRITE 2535 00 120 ;Actual position to technology from position detection
 WRITE 2537 02 210 ;Reference point
 WRITE 2538 00 212 ;Acknowledgement measured position valid
 WRITE 2539 00 122 ;Measured position from position detection
 WRITE 2953 21 4 ;In time slot T4
 WRITE 2963 21 2 ;LC_B241/242
 WRITE 807 00 7005 ;Sign of life from SIMOLINK
 WRITE 808 00 0 ;Fault reset sign of life
 WRITE 809 00 4 ;Fault level at which fault is triggered
 WRITE 2953 29 4 ;In time slot T4
 WRITE 2963 29 3 ;LC_KK_846/847
 WRITE 2800 01 7031 ;Position setpoint from SIMOLINK
 WRITE 2800 02 7033 ;Speed setpoint from SIMOLINK
 WRITE 2801 00 241 ;Communication error LC_B241/242
 WRITE 2802 00 524288 ;Axis cycle master in LU

WRITE 2953 33 4 ;In time slot T4
 WRITE 2963 33 5 ;Processing sequence
 WRITE 2600 01 846 ;Position setpoint master axis from SIMOLINK
 WRITE 2600 04 847 ;Speed setpoint master axis from VM via SIMOLINK
 WRITE 2601 00 524288 ;Master axis cycle length
 WRITE 2602 00 0 ;Synchronous mode continuous operation
 WRITE 2606 00 0 ;Master value source from SIMOLINK
 WRITE 2607 02 1048576.00 ;Standardization speed
 WRITE 2603 00 0 ;Synchronous function electronic gearbox
 WRITE 2604 01 30000 ;Gear ratio numerator
 WRITE 2604 02 30000 ;Gear ratio denominator
 WRITE 2605 01 804 ;Gear factor numerator
 WRITE 2605 02 805 ;Gear factor denominator

MSG Offset Setting ;
 WRITE 2676 00 0 ;Synchronize signal
 WRITE 2677 01 100000 ;Absolute offset
 WRITE 2697 01 1000.00 ;Acceleration in 1000 LU/s2 with offset setting
 WRITE 2697 02 100000.00 ;Positioning speed in 1000 LU/min with offset setting
 WRITE 2699 00 1 ;Direction of synchronization always positive
 WRITE 2501 11 524288 ;Slave axis cycle
 WRITE 2607 01 1048576.00 ;Maximum traversing speed in 1000 LU/min (P171*P353)
 WRITE 2501 49 100 ;Weighting factor
 WRITE 2671 00 120 ;Actual position, from position detection
 WRITE 2674 00 220 ;Position controller enabled, from position controller

MSG Position Detection Motor;
 WRITE 171 00 19 ;Position resolution 19 bits
 WRITE 172 00 0 ;Position setting value
 WRITE 173 00 0 ;Set position
 WRITE 174 00 301 ;Position correction value from synchronous block
 WRITE 175 01 303 ;Correction + from synchronous block
 WRITE 175 02 304 ;Correction - from synchronous block
 WRITE 177 00 0 ;Enable reference point detection
 WRITE 179 00 0 ;Enable measured-value memory
 WRITE 180 01 1 ;AVWF numerator
 WRITE 180 02 1 ;AVWF denominator
 WRITE 183 00 1001 ;Encoder detection (1=single-turn encoder)
 WRITE 184 00 0 ;Position offset from technology

MSG Position Controller ;
WRITE 190 00 310 ;Position setpoint from synchronous block
WRITE 202 01 134 ;Position controller limit by BB_DW
WRITE 204 01 1.000 ;Servo gain factor
WRITE 205 00 1048576.00 ;Vrated=P353*P171
WRITE 206 01 0 ;Reset time position controller
WRITE 209 01 312 ;Position controller precontrol from synchronous block
WRITE 210 01 205 ;Position controller enable from ramp generator bypass
WRITE 211 01 104 ;Position controller enable if drive ON
WRITE 212 01 0 ;Setpoint for control mode =0
WRITE 213 01 0 ;Only closed-loop position control permitted
WRITE 770 00 1 ;Ratio synchronous block to position controller
WRITE 771 00 111848 ;Maximum step change for interpolation

MSG Ramp-Function Generator;
WRITE 462 01 5.00 ;Ramp generator acceleration time 5.00 sec
WRITE 464 01 5.00 ;Ramp generator deceleration time 5.00 sec
WRITE 469 01 0.010 ;Ramp generator smoothing
WRITE 772 00 1 ;Ramp generator bypass always active

MSG Speed Controller ;
WRITE 220 01 75 ;n-set from bypass
WRITE 221 01 0.8 ;Setpoint smoothing
WRITE 222 00 91 ;n-act from position detection
WRITE 223 00 0.8 ;Actual value smoothing
WRITE 228 01 152 ;Input speed controller
WRITE 232 01 0 ;Input signal kp adaptation
WRITE 233 01 0.0 ;Characteristic point 1 for kp adaptation speed controller
WRITE 234 01 100.0 ;Characteristic point 2 for kp adaptation speed controller
WRITE 235 01 25.0 ;Cp1 for kp adaptation speed controller
WRITE 236 01 25.0 ;Cp2 for kp adaptation speed controller
WRITE 240 01 50 ;Speed controller reset time

MSG Torque Limitation ;
WRITE 262 01 0 ;Supplementary torque setpoint as acceleration precontrol
WRITE 263 01 200.0 ;FSW positive limit torque controller
WRITE 264 01 -200.0 ;FSW negative limit torque controller

MSG Control Word ;
WRITE 554 01 10 ;On command ON/OFF1 from BI

MSG Catch-Up
WRITE 2625 01 0x14 ; Input3 Stop
WRITE 2628 01 20000 ; Acceleration
WRITE 2628 02 20000 ; Deceleration

MSG Basic Positioner

WRITE 2885 00 0x20 ;**Input6** Synchronism local function ON
 WRITE 2886 01 0x871 ;Link betw. synchronism and position setpoint

 WRITE 2886 02 0x870 ;Link betw. synchronism and speed setpoint

 WRITE 2953 61 4 ; Enable basic positioner FB361 -789b- in T4

 WRITE 2856 00 1048576.00 ; BPOS rated speed
 WRITE 2857 00 787.5 ; BPOS rated acceleration
 WRITE 2858 01 524288 ; BPOS axis cycle
 WRITE 2850 01 0x418 ; BPOS position setpoint (constant value B2018)

 WRITE 2850 02 0x310 ; BPOS feedback position setpoint
 WRITE 2850 03 0x0 ; BPOS position setting value
 WRITE 2850 04 0x120 ; BPOS Actual position for POS_Ok checkback

 WRITE 2850 05 0x310 ; BPOS RM_IN FP789b
 WRITE 2851 00 0x41 ; BPOS positioning speed [%] P401
 WRITE 2852 01 0x1 ; BPOS adaptation speed UP
 WRITE 2852 02 0x1 ; BPOS adaptation speed DOWN
 WRITE 2853 1 0x0 ; BPOS preselect traversing direction forwards

 WRITE 2853 2 0x0 ; BPOS preselect traversing direction backwards

 WRITE 2853 3 0x0 ; BPOS setup mode
 WRITE 2854 1 0x10 ; **Input1** BPOS positioning<Start TCW B5

 WRITE 2854 2 0x16 ; **Input4** BPOS positioning Absolute(0)/Relative(1)

 WRITE 2855 1 0x0 ; BPOS set output
 WRITE 2855 2 0x20 ; **Input6** BPOS enable basic positioner

 WRITE 2656 1 0x3203 ; Switchover to catch-up SYNC input
 WRITE 2656 2 0x3203

...

9.4.41 Continue synchronism

The "Synchronism" function can be temporarily deactivated at U674.1 via disable/enable or call-up in the mode manager. Previously internal values/statuses of synchronism were reset in the process.

If the new function "Continue synchronism" is activated via binector input U674.2 the internal values/statuses are frozen. No internal reset of values/statuses takes place. Thus when synchronism is temporarily deactivated it behaves as if it had never been shut down.

The new bit 23 in the synchronism status word n450./ KK0800 and the binector 826 in function diagram 846 shows the activation of the "Continue synchronism" function.

During deactivation the synchronism output KK0310 is tracked to the position setpoint value output at U671, i.e. in the factory setting to actual position value KK0120.

During deactivation neither the master axes nor the slave axes may move in order to ensure that the transmission characteristics of the gearbox implemented via synchronism remain.

If, however, the master or slave axes do move, synchronism to the reference value can take place again (via the existing synchronizing functions with window evaluation).

Thus when the "Continue synchronism" function is active the following statuses remain:

- ◆ Table does not return to X0.0
- ◆ Synchronous status
- ◆ Referenced status
- ◆ Engage/disengage coupled
- ◆ Synchronization, offset angle setting [FD 841] are continued
- ◆ Position correction, referencing [FD 841] are continued

9.5 Communication with the technology

Communication with the technology functions across serial interfaces such as

- ◆ PROFIBUS-DP [120...135]
- ◆ CAN bus [120...135]
- ◆ USS [100...111]
- ◆ SIMOLINK [150...160]

takes place using the same features as used for access to the basic unit. This applies both to the high-speed cyclic process data interface ("PZD") and the acyclic parameter interface ("PKW"). Only process data, not parameters, can be accessed via SIMOLINK.

9.5.1 Process data communication (PZD)

The process data interface can be used to transfer all the signals of the MASTERDRIVES MC (actual values and status bits) defined as connectors or binectors (see [125] for example: here, any connectors can be "wired" to the send message of the PROFIBUS-DP field bus using the selection parameter P734).

All send data of the host system are already defined implicitly as connectors and binectors (e.g. K3001...K3060 and B3100...B3915 from the send message of PROFIBUS-DP [120]). They can therefore be "wired through" on the MASTERDRIVES converter as setpoints and control commands.

You can set up any configuration of send and receive messages using BICO technology simply by setting the appropriate parameters, however we recommend that you normally use a permanent message assignment for the positioning and synchronization functions with 10 words each in the send and receive direction (PPO type 5 for PROFIBUS-DP). This permanent message assignment can be established quickly and conveniently with the DriveMonitor download file POS_1_1.DNL [806].

The process data interface defined using this procedure is referred to below as the "GMC interface", since it is used in the "GMC-BASIC" software of the configuring package /1/ (GMC = General Motion Control).



The signals exchanged with the technology over the GMC process data interface are described in detail in the "Control and checkback signals" chapter of manual /1/. In the following two tables, you will find an illustration of the message format for send and receive messages:

Control signals from host system → MASTERDRIVES with the GMC interface

	7	6	5	4	3	2	1	0		Axis_n.
DBBx	RES	RES	RES	RES	RES	LB	RES	RES	BIN	IN_1
DBBx+1	ACK_F	RES	RES	RES	ENC	OFF3	OFF2	OFF1	BIN	IN_2
DBBx+2	MODE_IN				J_FW_D	F_S	J_BW_D	BLSK	BIN	IN_3
DBBx+3	OVERRIDE								DEZ	IN_4
DBBx+4	PROG_NO OR MDI_NO								DEZ	IN_5
DBBx+5	SIST	RST	FUM	ACK_M	CRD	STA	RIE	TGL_I	BIN	IN_6
DBBx+6	R_VM	S_VM	EN_RF	SSC	OPERATION		FUNCTION		BIN	IN_7
DBBx+7	ST_VM	TABLE_NO			SYN_T	SST	ST_S	SET_T	BIN	IN_8
DBBx+8	CU_DR	CU_EN	CU_SP	SYNC	DI_RN	DI_RP	DI_JN	DI_JP	BIN	IN_9_0
DBBx+9	RESERVED									IN_9_1
DBWx+10	OPTIONAL VALUE 1 INPUT									IN_9_2
DBDx+12	OPTIONAL VALUE 2 INPUT									IN_10
DBDx+16	OPTIONAL VALUE 3 INPUT									IN_11

The first data word (Dbx, Dbx+1) is reserved for control word 1 of the MASTERDRIVES basic unit [180]. The other words are technology-specific.

Checkback signals from MASTERDRIVES → host system with the GMC interface

	7	6	5	4	3	2	1	0	Axis_n.
DBBy	RES	RES	RES	RES	OTM	OTC	OLC	S MAX	BIN OUT_1
DBBy+1	RES	WA RN	OFF3	OFF2	FAU LT	IOP	RDY	RTS	BIN OUT_2
DBBy+2	FAULT_NO								DEZ OUT_3
DBBy+3	WARN_NO								DEZ OUT_4
DBBy+4	STR M	ARFD	FUR VM	OTR	FUT	BWD	FWD	DRS	BIN OUT_5
DBBy+5	M_NO_1								DEZ OUT_6
DBBy+6	MODE_OUT				FUR	ST EN	T_R	TGL O	BIN OUT_7
DBBy+7	M_NO_2								DEZ OUT_8
DBBy+8	CU TE	CU VR	CU PR	SYNC	DI_A	POS A	CL_A	VM RA	BIN OUT_9_0
DBBy+9	RESERVED								OUT_9_1
DBWy+10	OPTIONAL VALUE 1 OUTPUT								OUT_9_2
DBDy+12	OPTIONAL VALUE 2 OUTPUT								OUT_10
DBDy+16	OPTIONAL VALUE 3 OUTPUT								OUT_11

The first word is reserved for status word 1 of the MASTERDRIVES basic unit [200]. The second word is assigned to the fault/warning number KK250 [510]. The remaining words are technology-specific.

The tables above show the data block addresses as implemented with the SIMATIC-S7 "Motion Control Configuring Package" software /1/. The message assignment shown is equally suitable if you do not use the configuring package, but only use the DVA_S5 and DRIVE ES SIMATIC block packages or a bus other than PROFIBUS-DP (USS, CAN bus, etc.).

Configuration and starting of traversing operations in positioning and synchronization mode via a serial interface is simple and convenient, thanks to the clearly defined message format. It is possible, for example, to define a positioning offset with a single message ("MDI block") (in the "Optional Values") and to start the movement simultaneously with the start command [STA]. The complete traversing operation is then performed automatically without further intervention from the host controller. At the end of the traversing operation, the axis returns a checkback message indicating that the target position has been reached ("destination reached, axis stationary" status bit [DRS]). This naturally applies not only to PROFIBUS-DP, but also to other field buses (CAN bus, USS, etc.).

9.5.2 Parameter transfer (PKW)

Every setting and display parameter of the MASTERDRIVES MC can be read and modified over a serial interface – with the exception of SIMOLINK – including all the technology parameters.



The methods used for PKW access are described in detail in the "Communication" section of this Compendium.

Cyclical services

Only one parameter can be accessed in a message frame. It is not possible to access a new parameter until the old parameter access is complete ("handshake method").

Acyclical services

The new PROFIBUS-DPV1 services and USS protocol also allow access to all the indices of a parameter in a "long message" (see below).

When transferring parameters, it should be noted that the U and n parameters used for the technology are addressed with a "1" in the most significant bit (bit 15) of the index word.

Example:

Access to U551 => Parameter number in parameter identifier word
= 551

Bit 15	in index word	} =1 (PARA PAGE SELECT bit)
	with DPV1 and USS	
	or	
Bit 7	with cyclical	
	PROFIBUS services	

9.5.3 Standard function blocks for PROFIBUS-DP and USS

As you can see in tables /3/ and /4/, a solution for connecting to MASTERDRIVES via PROFIBUS-DP and USS is available for almost every SIMATIC S5 and S7 system.

Function block packages DVA_S5 /3/ and Drive ES SIMATIC /4/, which are available for this purpose, enable convenient access to the process data and parameters of the MASTERDRIVES from the perspective of the SIMATIC application programmer.

The control and checkback signals (e.g. in the standard message assignment specified above) are stored in convenient packages in data blocks for each drive axis.

PROFIBUS-DPV1 services

The SIMATIC-S7 CPUs with integrated PROFIBUS interfaces (see table /4/) also support communication with MASTERDRIVES using the new PROFIBUS-DPV1 services. The DPV1 services allow parameter transfer to the drive in long messages: all indices of a parameter are transferred in a single PROFIBUS message frame. This allows, for example, the transfer of a cam table with 100 interpolation points (= 200 double words) in seconds, in 4 instead of 200 messages.

9.5.4 Additionally available SIMATIC S7 software

Standard function blocks DVA_S5 and Drive ES SIMATIC mentioned above allow you to access all the positioning and synchronization functions of the MASTERDRIVES MC - with one exception: a solution has not yet been released for the definition of automatic programs.

If you want to

- ◆ Download new cam tables onto the MASTERDRIVES MC on a frequent basis, e.g. on a product change
- ◆ Define extensive automatic programs
- ◆ Access preprogrammed OP screens and
- ◆ Are prepared to invest extra time for familiarization

the following 2 components are available for the total integration of MASTERDRIVES MC with distributed technology in a SIMATIC S7-300/400 programmable controller system (further information is provided in manual /1/ and Catalog LS01):

- ◆ **SIMATIC S7 "Motion Control Configuring Package" software on CD-ROM (included in /1/):**

This software package for the SIMATIC S7-300 and S7-400 includes software for PROFIBUS-DP communication between the S7 user program and the technology across a clear, easy-to-handle data interface. The structure of the communication interface to the technology is illustrated in the following figure:

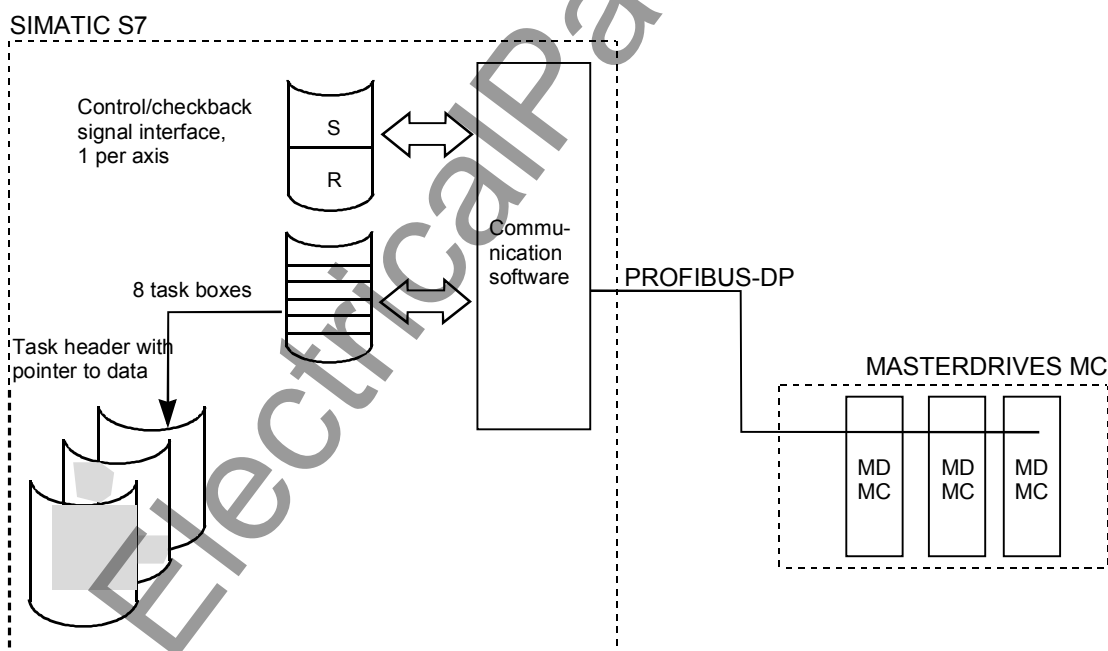


Fig. 9-44 Communication interfaces from GMC-BASIC to technology

- ◆ The configuring package offers the following advantages:
 - Transfer of control/checkback signals to technology (one data area per axis)
 - Task interface for definition of MDI and automatic NC blocks and programs, in addition to transmission ratios, cam tables, etc.
 - The task concept is largely identical to the system used on the SIMATIC WF721/723 positioning modules.
 - The communication software provides the user with 8 task boxes in which he can enter the tasks. These are executed automatically and enable superior structuring of the user program.
 - A task comprises a task header, with the necessary control information, and a pointer to the actual user data.
 - Resources such as an OP25 operator panel, a STEP 7 program or a SIMATIC PG programming device can use the task interface.
- ◆ **Motion Control HMI Package for SIMATIC S7 (see I2/):**
Software for the application interface to operator/touch panels OP25, OP27, OP37, TP37 etc. with standard screens for operating the positioning axes, including the following functions:
 - Block and automatic NC program input
 - Input of machine data and cam tables
 - Diagnostic screens with definition/display of control and checkback signals



In manual /1/ you will find detailed descriptions of the configuring package and HMI package.

9.5.5 USS interface

The MASTERDRIVES MC compact PLUS units have one USS interface; the compact and rack-mount units have two. The USS interface is the preferred interface for connecting the OP1S plain text servicing unit or a servicing and commissioning PC with the DriveMonitor service program. In applications where the transmission rate is not critical, the USS interface can also be used as a low-performance field bus.

USS has the following features:

- ◆ The logical message contents are basically identical to the PROFIBUS-DP message contents. Point-to-point connection (max. 15 m) via RS232 or
- ◆ Network connection via RS485 with up to 32 stations (max. 1000 m)
- ◆ Baud rate adjustable from 300 . . . 38400 baud (with additional modules up to 187.5 Kbaud)
- ◆ Simple, high-performance protocol with only 4 bytes overhead. You will find a specification of the USS protocol in the "Communication" chapter of this Compendium.
- ◆ **Every** MASTERDRIVES, SIMOREG and MICROMASTER converter has at least one USS interface as standard.
- ◆ User data:
 - All setting and diagnostic parameters are accessible; up to 200 bytes of parameter data can be transmitted in one message (one parameter or all indices of a parameter).
 - Up to 16 words of process data (setpoints/actual values, control/status bits)
- ◆ Where the network polling time is not critical, the USS can be used as a low-cost field bus.
- ◆ A USS interface and drivers are available for almost every SIMATIC S5/S7 CPU and PC interface (see /3/ and /4/).
- ◆ The USS is suitable for the connection of Siemens power converters to third-party PLCs, PCs or customized automation systems.
- ◆ The network polling time for a USS bus operated at 19.2 Kbaud with 10 drives connected to an S7 with CP340: approx. 650 ms (with 6-word message frames, 4 words of parameters and 2 words of process data).

9.5.6 SIMOLINK

The SIMOLINK drive interface is the "backbone" of the synchronization function. SIMOLINK distributes path/angle setpoints and velocity setpoints from the master axis to the slave axes quickly and time-synchronously. The synchronization of the sampling times of all stations is assured by special SYNC messages.



The "Communication" chapter in this Compendium provides detailed information on the configuration and commissioning of SIMOLINK.

The list below provides a brief overview:

SIMOLINK has the following features:

- ◆ Fiber-optic ring with plastic or glass fiber
- ◆ Baud rate 11 Mbaud
- ◆ Max. 200 nodes per fiber-optic ring
- ◆ Polling time with 100 data messages each with 32 bits: 630 µsec
- ◆ Jitter-free synchronization of the sampling times of all stations by special SYNC messages
- ◆ Peer-to-peer configuration (drive-to-drive interface without master) or master/slave configuration possible
- ◆ Max. cable length:
 - 40 m with plastic
 - 300 m with glass
 - 1000 m for complete ring
- ◆ Up to 1000 double word message frames can circulate on the SIMOLINK ring.
- ◆ Binectors and connectors allow flexible wiring of signals from and to the SIMOLINK per software in MASTERDRIVES MC [150...160].

Applications of SIMOLINK:

- ◆ Replacement of mechanically coupled moving axes with individual drives
- ◆ Transmission of angle setpoints between master and slave axes for angular synchronization and cam function
- ◆ Replacement of conventional RS485 peer-to-peer connection for data exchange between SIMOVERT power converters

Special SIMOLINK properties in a master/slave configuration

Master interfaces are available for:

- SIMADYN D
- SIMATIC FM458
- SICOMP SMP
- ♦ The master can write data into a maximum of 1000 doubleword messages. The slaves can read doubleword information from a maximum of any 8 message locations.
- ♦ Cross-traffic, i.e. message exchange between slaves, is possible. The traffic is always routed via the master, however.

Special SIMOLINK properties in a peer-to-peer configuration with no host system

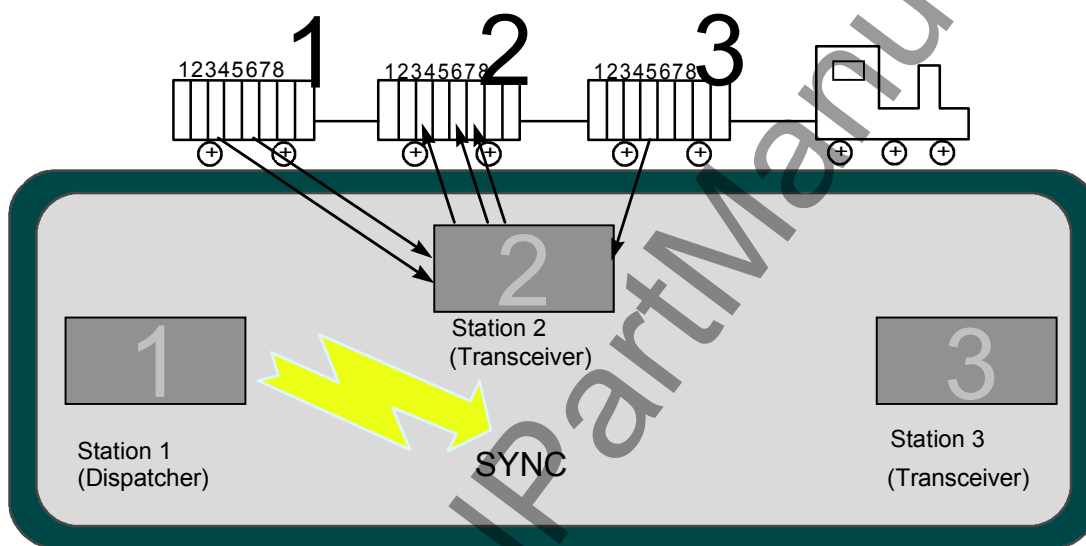


Fig. 9-45

The SIMOLINK drive interface can be compared to a goods train that passes through various stations on a circular route. The stations in this example are represented by 3 MASTERDRIVES converters:

- ♦ A "data train" sent by a dispatcher travels along the SIMOLINK ring. Otherwise, the dispatcher has the same function as the two transceivers.
- ♦ A goods carriage with capacity for 8 double word messages (called "channels") is permanently assigned to each station. The station can only store send packets in this carriage.
- ♦ Each station can read up to 8 data packets from any goods carriage, however.
- ♦ After one lap of the circuit, the dispatcher sends a SYNC message "to all". All stations now start their sampling time at exactly the same instant with related setpoints.
- ♦ The drives coordinate themselves without the need for a centralized host system.

9.6 Configuration

9.6.1 Encoders for the position sensing system

The sensor boards for the various encoders that can be connected to MASTERDRIVES MC are described in the first section of "Brief description of the technology functions". The table is intended as a configuring aid, and shows an overview of the properties of the various encoders:

Encoder Type	Sensor Board in MASTERDRIVES MC	Resolution [Increments/ Revolution]	Achievable Positioning Accuracy ¹⁾ [Increments/ Revolution]	Can be Used as	
				Motor Encoder	External Encoder
Resolver ²⁾	SBR1/SBR2 (without/with pulse encoder simulation)	4096 inc./rev. with 2-pole resolver	1024 inc./rev. with 2-pole resolver	Yes	No
Sin/cos encoder ERN 1387 ⁵⁾	SBM2	16,8 x 10 ⁶ Inc/rev.	10 ⁵ ... 10 ⁶ Inc./rev.	Yes	Yes
Absolute encoder EQN 1325 ⁵⁾	SBM2	16.8 x 10 ⁶ inc./rev. 4096 rev. reproducible ⁷⁾	10 ⁵ ... 10 ⁶ Inc./rev.	Yes	Yes
Pulse encoder ³⁾	SBP	No. of lines x 4, i.e. 4096 inc./rev. with standard motor encoder	Number of lines x 1, i.e. 1024 inc./rev. with standard motor encoder	Yes (with asynchronous motor)	Yes
SSI absolute encoder ⁴⁾	SBM2	4096 inc./rev. typical 4096 revolutions typically reproducible	1024 inc./rev. typical	No	Yes
Absolute encoder EQI1325 ⁶⁾	SBM2	4096 Inc/rev.	1024 Inc/rev.	Yes	No

Notes:

- 1) In practice, the resolution of the encoder must be between 1 and 10 times higher than the required positioning accuracy. The accuracies specified in the table are only approximate guidelines.
- 2) Notes on resolver:
 - Correspondingly higher resolution and accuracy with multi-pole resolvers
 - In the following cases, you should use an ERN1387 sin/cos encoder instead of a resolver:
 - If extremely high positioning accuracy is required
 - If an extremely high control response is required
 - If printing indices are to be detected exactly
 - If a good rotary response is required at extremely low speeds below approx. 5 rpm.
 - With the SBR2, the pulse encoder simulation is connected to terminals with 2 tracks each with 512 or 1024 pulses per revolution (adjustable) and zero pulse, RS422 level (TTL differential signal). Valid for 2-pole resolvers; correspondingly larger number of pulses per revolution with multi-pole resolvers.

- 3) Notes on pulse encoder:
 - Pulse quadrupling takes place internally on the SBP (edge evaluation)
 - Number of lines configurable between 4 and 32768 lines per revolution
 - Max. pulse frequency that can be evaluated: 410 kHz
 - HTL and RS422 level that can be evaluated
- 4) Notes on SSI encoders:
 - Large number of SSI encoder designs on the market with various resolutions (singleturn and multiturn, linear scales, etc.)
 - All encoders with standard SSI protocols can be evaluated (e.g. SIEMENS, Stegmann, TR, Fraba, Heidenhain, infrared distance measuring systems, etc.)
- 5) Notes on SBM2: pulse encoder simulation connected to terminals with 2 tracks each with 2048 pulses per revolution and zero pulse; RS422 level.
- 6) Notes regarding SBM2: Pulse encoder simulation brought out to terminals with 2 tracks with 32 pulses each per revolution and zero pulse, RS422 level.
- 7) The maximum possible number range for position setpoints and actual values is limited to 32 bits. If, for example, a resolution of 24 bits per revolution is selected, only 7 bits are then possible for the representation of the countable revolutions. One bit is required for the positive/negative differentiation. If the 4096 revolutions of the absolute encoder (EQN 1325, EQI 1325) are to be represented, the resolution for one revolution must be reduced to at least 19 bits.

9.6.2 Requirements of position encoders for rotary axes

Condition for rotary axis positioning with **absolute encoder** (i.e. without homing):

1 rotary table revolution must correspond to 2^n encoder revolutions ($n = 0, 1, 2, 3, 4, \dots$).

Example:

For an SSI encoder, which can detect 4096 revolutions with 4096 steps each, $\Rightarrow$ 1 rotary table revolution must correspond exactly to 1, 2, 4, 8, 16, 32 etc. revolutions of the encoder.

Remedy: see section 9.4.9

Condition for rotary axis positioning with **incremental encoder** (resolver, ERN encoder, pulse encoder):

When determining the actual value weighting factor (AVWF, number of LUs per encoder increment; e.g. P169, P170 when using the motor encoder), the result must be a number with a maximum of 8 decimal places; the 9th and all subsequent decimal places **must** be "0".

Examples:

- ◆ 1 encoder increment corresponds exactly to 23.123456780000 LU
 $\Rightarrow$ O.K.
- ◆ 1 encoder increment corresponds to 23.123456789123...LU
 $\Rightarrow$ not O.K.

Remedy: specify AVWF with numerator and denominator (P180 or P181)

9.6.3 Brake controller

The automatic brake controller [470], which is already integrated in the standard MASTERDRIVES software, can be activated with P605 = 1 or P605 = 2 for brakes with checkback contact(s) [470.7].

The integrated brake controller [470] prevents inconvenient pauses on application and release of the brake. Even lifting systems are positioned rapidly and reliably - with minimum effort required on the external machine controller and during commissioning.

The output signals of the brake controller are the binectors "open brake" B275 and "close brake" B276 [470.8]. Relays for actuation of the brake are not installed in the MASTERDRIVES unit. The following options are available for brake actuation:

- ◆ Use of a relay output on terminal expansion board EB2
- ◆ Use of an external relay actuated by a digital output of the MASTERDRIVES
- ◆ The relay installed on compact and rack-mount units for main contactor actuation can be used for the brake controller if no main contactor is provided (P601 = 275).

The opening and closing of the brake can also be initiated by external commands (using the connectors selected with P608, P609 and P614 [470.1]), however the brake controller normally operates fully automatically without intervention from the external machine controller. The BICO wiring required for this purpose is shown in the comment boxes in [470]. The fully automatic brake controller normally operates as follows:

Open brake

When the drive switches to "operation" mode after power on, the inverter enable is activated and the brake is opened. After the brake opening time set in P606 (factory setting 200 ms [470.5]), if the "brake open" checkback signal is active, the setpoint enable is activated.

The limit monitor, set in P611 [470.3], can be used in special situations, in order to open the brake depending on a specific condition (e.g. if a certain torque level is exceeded; in this case "brake open" is activated by binector B281 and binector B277 "setpoint enable" must not be wired directly).

Close brake

When the drive is brought to a standstill, i.e. when its speed has dropped below the threshold set in P616 [470.3] and it is switched off with OFF1, OFF2 or OFF3, the brake closes. The inverter enable is canceled when the brake closing time set in P607 has expired (factory setting 100 ms [470.5]) and the "brake closed" signal is activated (by a checkback contact if one is installed). You should avoid using OFF2 where possible, since the pulses are disabled immediately on an OFF2 command, and the motor is already without power during the brake closing time.

9.7 Application examples

9.7.1 Positioning of a linear axis via PROFIBUS

Application examples can be requested from the regional office of SIEMENS AG or from the application center for production machines.

9.7.2 Positioning and synchronization with virtual master axis (suitable for self-study)

9.7.2.1 Task description

This example is intended to

- ◆ Help the user configure and commission the drive
- ◆ Provide a rapid means of familiarization with the positioning and synchronization functions with reference to a trial configuration.

You can run this example application using the 2-axis demonstration pack available from Siemens (Order No. 6SX7000-0AF10; see /1/).

You need the following components in order to use the example configuration:

Component	Quantity Required for Positioning	Quantity Required for Synchronization
1FT6 or 1FK6 motor with resolver or encoder *)	1	2
MASTERDRIVES MC with option F01 and the matching sensor board	1	2
Switch box with 6 switches	1	2
Potentiometer, approx. 10 K **)	---	1

OR

1-axis demo. pack /5/	1	2

OR

2-axis demo. pack /6/	1	1

*) An asynchronous motor can also be used, however some parameter settings require slight modification.

**) You can connect the +10 V terminal on the potentiometer to the analog output on terminal X101.11. You must then set P640 = 1 [80.1], so that +10 V is output at the analog output (corresponds to 100 %).

The application example contains the following configuration:

- ◆ 2 Siemens synchronous servo motors: 1FK6 with resolver and 1FT6 with optical sine/cosine encoder (only one motor required for positioning)
- ◆ 2 MASTERDRIVES MC converters with technology option F01 (only one converter required for positioning)
- ◆ Both drives should be operated in the following modes:
 - Homing (this is required for positioning, since resolvers and optical encoders are incremental and not absolute encoders)
 - Point-to-point positioning (MDI; axis type "rotary axis", i.e. without fixed stops)
 - Synchronization with 1:1 transmission ratio using the virtual master axis and the SIMOLINK drive interface
- ◆ When the two-axis pack is used, the synchronization can be checked with reference to an LED light beam, which is visible through drilled holes in the flywheel mounted on the motor shafts when the synchronization is operating correctly.

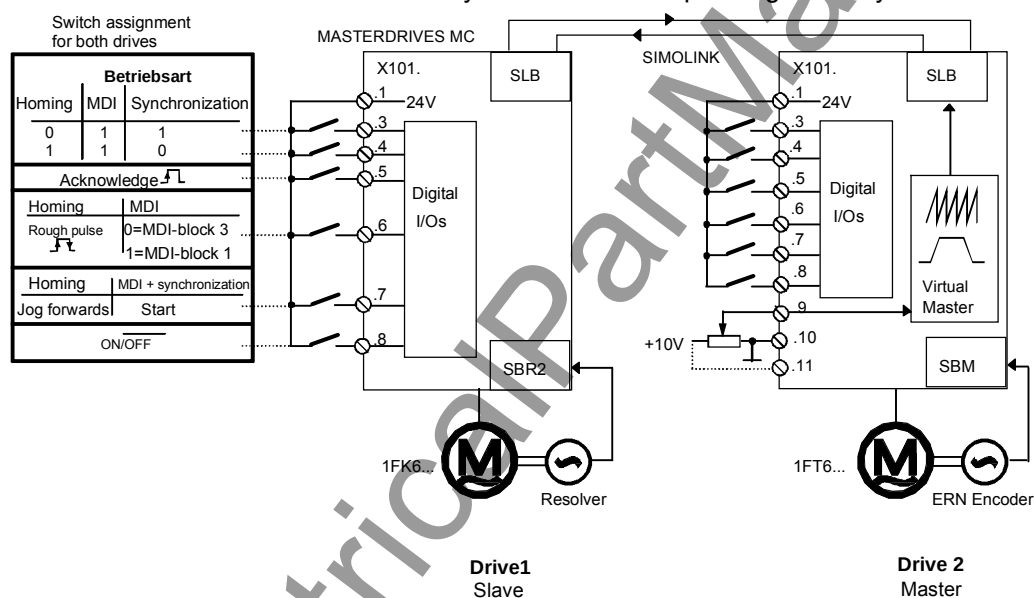


Fig. 9-46 Application example 2: hardware configuration and wiring

The application example guides you through the relevant pages of the function diagram and the parameter settings. It is assumed that the basic units have been started up in speed control mode, as described in Chapter 6. If you only want to use the positioning functions, you need only one drive instead of two for the self-study and you can skip the sections starting from 9.7.10.

9.7.2.2 Overview diagram

The overview diagram in Fig. 9-47 shows how the technology functions are interconnected.

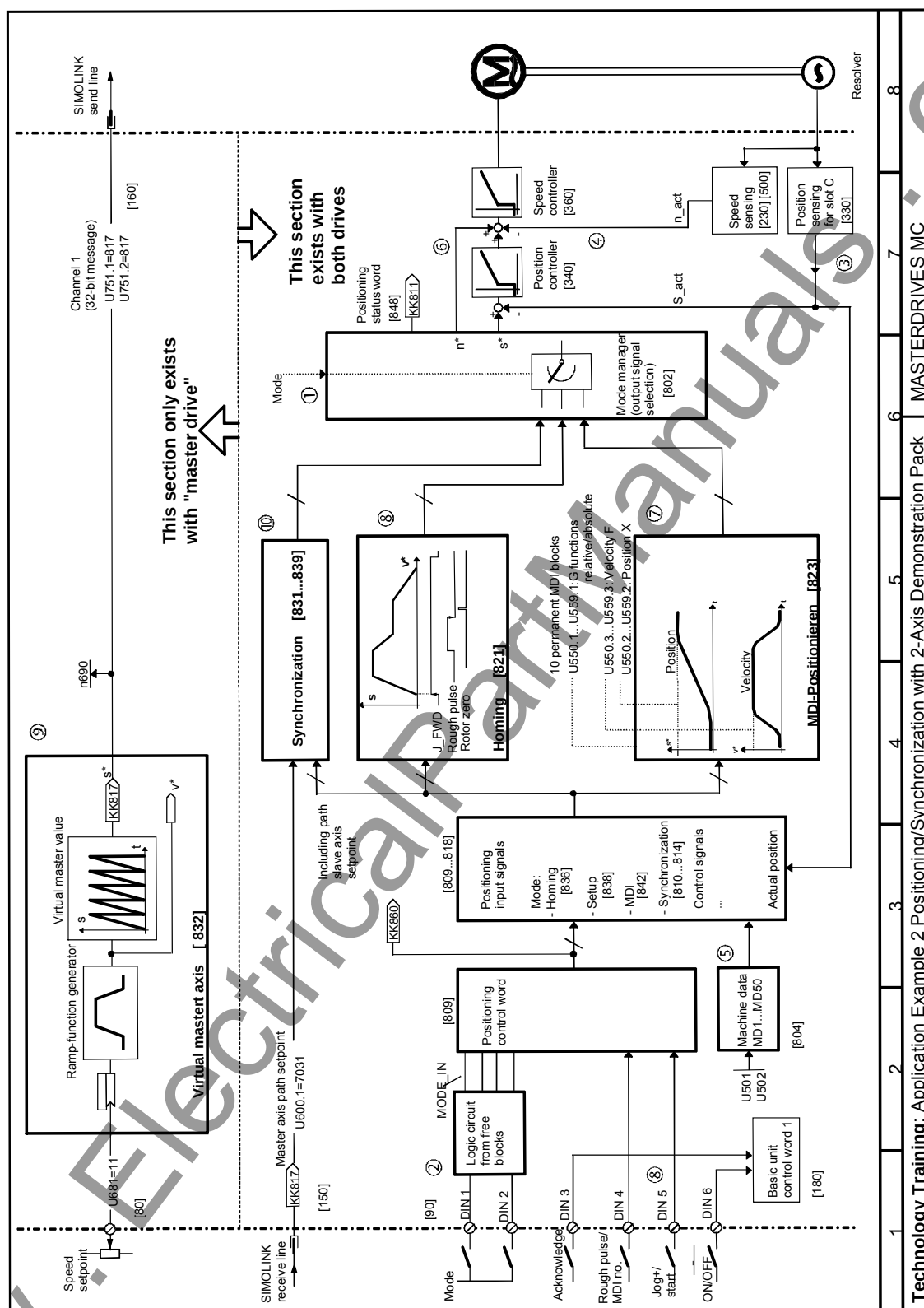


Fig. 9-47

Application example 2: positioning/synchronization with 2-axis demonstration pack

The numbers in circles indicate the corresponding areas on the overview diagram.

The "mode", "acknowledge" and "MDI no." switches, which are connected to the 4 digital inputs (②, function diagram, sheet [90]), are routed to positioning control word [809]; the mode is generated from free blocks via a small logic circuit (described below). The "acknowledge" and "ON/OFF1" switches act directly on basic unit control word 1 [180].

The mode selector ① activates the synchronization ⑩, homing ① and MDI (point-to-point positioning ⑦) modes. The mode manager ① ensures that the output signals of the active mode are switched through to the setpoint command on the position and speed controller ⑥.

The virtual master axis ⑨ contains the velocity ramp-function generator for both drives and the "saw-tooth pulse generator" for the position setpoint generation (master value, period corresponds to 10 motor revolutions). The virtual master axis is only calculated on drive 2. It is not activated on drive 1. This, and the velocity setpoint potentiometer, which is only connected to drive 2, are the only differences in the parameter settings for both drives. We will initially limit our description to drive 2. The further procedure is as follows:

- ◆ Commission the positioning functions on drive 2 (sections 9.7.2.3...8).
- ◆ Test the positioning functions on drive 2 (section 9.7.2.8). Users whose application only involves positioning can skip the following steps.
- ◆ Commission the virtual master axis on drive 2.
- ◆ Test the virtual master axis on drive 2.
- ◆ Commission the synchronization on drive 2.
- ◆ Commission the positioning and synchronization functions on drive 1.
- ◆ Test the positioning and synchronization functions on drives 1 and 2.

The parameters marked with (WE) do not have to be entered, since the factory settings are suitable.

9.7.2.3 Connection of digital inputs

Fig. 9-47 shows the digital input assignments used in this example ②.

The assignment of functions to individual terminals used here is arbitrary. The BICO technology (connector/binector technology) allows virtually any terminal wiring to be implemented.

Terminal 8 is connected by means of the following parameter setting to the OFF1 command in basic unit control word 1, which also actuates the inverter enable in this example (the square brackets indicate the relevant pages in the function diagram):

```
P554.1=20 ; FF1 command from terminal X101.8 [90] ==> [180]
```

Terminal 5 is assigned to the "acknowledge fault" function (basic unit control word 1)

```
P565.1=14 ; Acknowledge fault from terminal X101.5 [90] ==> [180]
```

Terminal 7 has a dual assignment:

- ◆ In homing mode you use it to define the "jog forwards" signal [J_FWD], which starts the homing procedure:

```
U710.28=18 ; Jog forwards [J_FWD] from term. X101.7 [90] ==> [809]
```

- ◆ In MDI and synchronization modes you use it to specify the start command [STA] used to start a movement (see manual /1/ "Motion Control for MASTERDRIVES MC and SIMATIC M7", "Control and checkback signals" chapter).

```
U710.3=18 ; Start command [STA] from terminal X101.7 [90] ==> [809]
```

Terminal 6 has a dual assignment:

- ◆ In homing mode, the rough pulse is expected from the home position cam or proximity switch (BERO) acting on the position sensing system (see also MD45 in the "Machine data input" section):

```
P178=16 ; Reference BERO from terminal X101.6 [90] ==> [330]
```

- ◆ In MDI mode, terminal 6 is used to switch between MDI block 1 (low signal) and 3 (high signal). This selection is made via bit 9 of the positioning control word [809], which is connected to the MDI mode [823] and switches there between the permanent NC blocks configured in U550 and U552.
Bit 8 of the positioning control word is initialized permanently to "1":

```
U710.10=16 ; MDI block selection [MDI_NO] from terminal X101.6
U710.09=1 ; [90] ==> [809]
```

Terminals 3 and 4 select the mode according to the following truth table:

Signal at Terminal 3	Signal at Terminal 4	Mode	Bitmap at [MODE_IN] [809.4]			
			2 ³	2 ²	2 ¹	2 ⁰
0	0	-	-	-	-	-
1	0	11 = Synchronization	1	0	1	1
0	1	2 = Homing	0	0	1	0
1	1	3 = MDI	0	0	1	1

The small logic circuit below generates the required mode selection bits 28...31 [MODE_IN] for the positioning control word [809] from the signals at terminals 3 and 4:

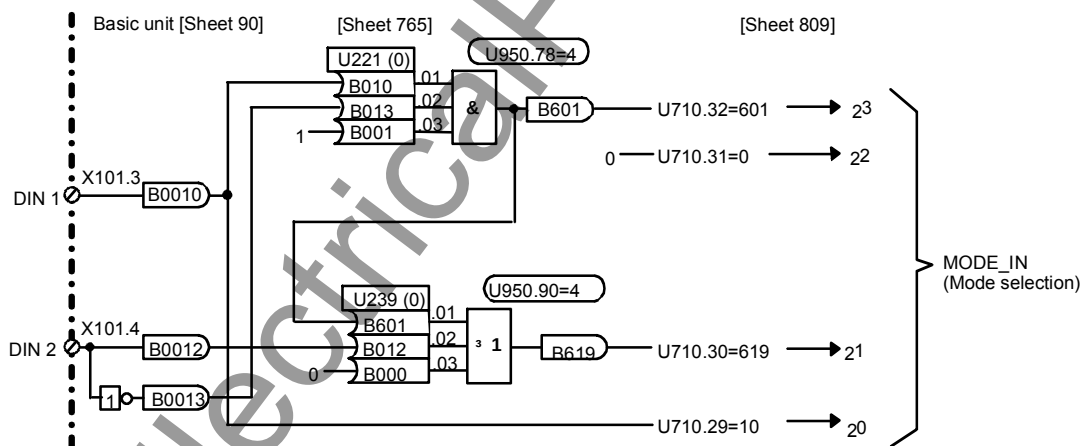


Fig. 9-48 Application example 2: circuit for generating the modes

This circuit is established by the following parameters using a free AND and OR element on sheet [765] of the function diagram:

```

U950.78=4    ; Nest AND element in sampling time 24 x T0    [765]
U950.90=4    ; Nest OR element in sampling time 24 x T0     [765]
U221.1=10
U221.2=13
U221.3=1      ; (WE) factory setting; can be retained
U239.1=601
U239.2=12
U239.3=0      ; (WE) factory setting; can be retained
U710.32=601
U710.31=0     ; (WE) factory setting; can be retained
U710.30=619
U710.29=10

```

You can check that the mode input is generated correctly at the [MODE_IN] switches with display parameter n540.14 [809.8] after you have nested the binector/double-connector converter for generation of the positioning control word [809] in a sampling time as follows (see also [702]):

```

U953.30=4    ; Nest pos. control word generation in sampling time T4
              (=24*T0=16*200µs=3,2 ms with 5 kHz converter freq.)

```

If you use the 2-axis demonstration pack, please note that all 4 jumpers must be plugged in crosswise so that all 4 bidirectional digital I/Os are configured as inputs.

9.7.2.4 Connection and parameters of the position sensing system

Connection of the position sensing system

The technology [815] is connected to the position sensing system ③ for the motor encoder in slot C [330] using the following circuit. Most of the parameters can remain in the factory setting (WE):

Signals from position sensing [330] ==> technology [815]:

```
U535=120          ; Actual position
U529= 70          (WE) ; "Actual position O.K." binector from resolver
                  ; sensor board in slot C [230]
U539=122          ; Measured position from measured value memory
U538=212          (WE) ; "Measured position valid" binector
U537.02=210      (WE) ; "Home position detected" binector
```

Signals from technology [815] ==> position sensing [330]:

```
P172=302          ; Position set value
P173=302          (WE) ; "Set position" binector
P174=301          ; Position correction value
P175.01=303       (WE) ; "Correct position +" binector
P175.02=304       (WE) ; "Correct position -" binector
P184=303          ; Position offset
P179=308          (WE) ; "Enable measured value memory" binector
P177=307          (WE) ; Enable homing
```

Defining the length unit (LU) and setting the actual value weighting factor (AVWF)

- ◆ Defining the length unit (LU):
In this application example, the actual position measurement is to be weighted such that the user can specify his position setpoints in the length unit [1 Length Unit = 1 LU = 0.1°], i.e. in tenths of one degree. For example, a setpoint of 3600 is to correspond to a distance of 360.0°, i.e. one revolution of the motor. It is assumed that no gearbox is used.
- ◆ Determining the actual value weighting factor (AVWF):
With the factory setting P171 = 12, the "shift division block" [330.4] outputs an actual position signal with 4096 increments per motor revolution. The actual value weighting factor (AVWF) specifies the number of length units (LU) per increment. Hence,
 $AVWF = 3600/4096 \text{ [LU/increment]} = 0.87890625$.

The actual value weighting factor is entered as follows in parameters P169 and P170 [330]:

```
P169=0            ; Decimal places of actual value weighting factor
P170=87890625     ; Decimal places of actual value weighting factor
```

Configuration of the position sensing and homing systems

The following parameters enable position sensing and homing [330.2] for the motor encoder in slot C and select the homing direction in the direction of increasing positional values (the same direction must be entered in machine data MD5; see step ④):

```
P1830011         ; Enable sensing and homing,
                  ; positive approach direction for home position
                  ; to right of BERO
```


9.7.2.5 Velocity normalization P353 [20.5] and P205 [340.2]

Parameters P353 and P205 are used to define the maximum traversing velocity ④, which must never be exceeded during operation (mechanical limit velocity).

In our application example, we want to set a limit of 1 000 000°/min, i.e. 10 000 000 LU/min (1 length unit = 1 LU = 0.1°; see above). P205 must therefore be set to the following value:

P205=10 000	; Rated velocity 10 000 000 LU/min,	
	; Input in [1000 LU/min]	[340.2]

This parameter mainly affects only the normalization of the loop gain for the position controller. The value of P205 must also be entered in machine data MD23 (see below).

Since no gearbox is installed, the reference speed of the motor P353 (in min⁻¹) can be calculated directly, i.e. the motor speed at which the rated velocity P205 is reached:

$$P205 = 10\,000\,000 \text{ LU/min} = 1\,000\,000^\circ/\text{min} = (1\,000\,000/360) \text{ min}^{-1} = 2777.777 \text{ min}^{-1}$$

P60=5	; Change to "drive setting" parameter menu
P353=2778	; Reference speed in motor revs/min [20.5]
P60=0	; Exit drive setting

The speed setpoint definition KK0150 for the speed controller [360.4] refers to this reference speed: if KK0150 is equal to 100 %, the reference speed of 2778 min⁻¹ defined in P353 is the motor speed. The following motor speed setpoints output by the technology refer to this reference speed:

- ◆ The speed precontrol value KK312 [817.7 and 836.8] for the position control modes
- ◆ The speed setpoint K311 [817.7] for the speed control modes, e.g. homing

The velocity override is normally set to U708 = 100 %. You can use this parameter to reduce the velocity of all movements, e.g. in the initial commissioning phase.

U708=78	(WE) ; Fixed velocity override 100 % [809.1]

P770	; Setting as described in Section 9.4.10
P771	

9.7.2.6 Machine data input U501 and U502 [804]

Machine data MD1 to MD50 (parameters U501.01 to U501.50) are used to define centralized settings required by the working machine and the mechanical transmission elements for positioning and synchronization ⑤. In our application example, the following settings are required:

Position encoder and axis type

U501.01=1	; MD1: encoder type "incremental encoder" (resolver)
U501.11=36000	; MD11: axis type "rotary axis with axis length 36000 LU (equals 10 motor revolutions, see actual value weighting factor (AVWF))"

Defining the traversing velocity and the acceleration/ deceleration ramps:

The same value as P205 should be entered as the maximum traversing velocity MD23 (see above). All output motor speed setpoints and the acceleration/deceleration times of the velocity ramps in the homing [821] and control [825.3] modes refer to MD23.

U501.23=10 000	; MD23: Max. traversing velocity is 10 000 000 LU/min. Input in [1000 LU/min]. Enter same value as P205; see above!

For the acceleration MD18 and deceleration MD19 for the position control modes we will assume that the customer requires an acceleration time of 0.5 s from 0 to MD23. The acceleration is then as follows:

$$\text{Acceleration} = \text{MD23} / \text{acceleration time} = (10\,000\,000 \text{ LU/min}) / 0.5 \text{ s} = 333\,333.333 \text{ LU/s}^2$$

We will also assume that the deceleration MD19 is to take place with the same ramp gradient as the acceleration:

U501.18=333	; MD18: acceleration for the position control modes [$\times 1000 \text{ LU/s}^2$]
U501.19=333	; MD19: deceleration for the position control modes [$\times 1000 \text{ LU/s}^2$]

For the acceleration time MD41 for the speed control modes homing [821] and control [825.5], we will assume that the customer requires 0.7 s for an acceleration from 0 to the velocity defined in MD23. 0.7 s are also used as the deceleration time in MD42 (refers to a deceleration from the velocity in MD23 to 0). The following machine data must be entered:

U501.41=700	; MD41: acceleration for the speed control modes [ms]
U501.42=700	; MD42: deceleration for the speed control modes [ms]

Defining the machine data for homing [821]:

The home position is to be approached at 1/5 of the maximum velocity (MD23/5). When the axis moves away from the proximity switch area (falling edge of rough pulse), the velocity is to be reduced to 1/40 of the maximum velocity. Machine data MD7 and MD6 should be set as follows:

```

U501.07=2000    ; MD7: Home position approach velocity = 1/5
                  ; of maximum velocity = MD23/5 =
                  ; 2000 [x 1000 LU/min], equals 556 min-1 on the
                  ; motor shaft
U501.06=250      ; MD6: Home position reducing velocity = 1/40
                  ; of maximum velocity = MD23/40 =
                  ; 250 [x 1000 LU/min], equals 69 min-1 on the
                  ; motor shaft

```

Note <3> on [821.1] must be observed for the alignment of the home position rough pulse for a unique assignment of the resolver zero crossing to the rough pulse. In our application example, we will assume that the home position is located to the right of the current position, i.e. in the direction of increasing positional values, at the start of the homing procedure. We will also assume that the machine zero, to which all position setpoints refer, is located at an offset distance of +3440 LU (equals 344°), as defined in MD4, from the zero crossing. This results in the following machine data:

```

U501.03=0    (WE); MD3: Home position coordinate = 0, i.e. MD3 can
                  ; remain at the factory setting.
U501.04=3440    ; MD4: Home position offset = 3440 LU
U501.05=1    (WE); MD5: Home position approach direction "to right
                  ; of BERO" (N.B.: this value must also
                  ; be entered in P183 [330]; see Chapter 4

```

The rough pulse proximity switch is connected to digital input terminal X101.6. This signal is already connected to the position sensing system via P178 (see Chapter 3). Machine data MD45 must also be used to connect the rough pulse to the technology [90] ==> [813.4] ==> [821.2]:

```

U536.4=16    (WE); Connect BERO signal from digital input
                  ; terminal 6 to "digital input I4 for positioning"
U501.45=7000    ; MD45: I4 acts as
                  ; "BERO for home position"

```

Transferring machine data [804]:

The transfer and activation of the machine data is performed by switching the drive on/off or by setting the following parameters (only possible when the drive is stationary).

```

U502=2        ; Transfer and activate machine data. If
                  ; machine data transferred without error, U502
                  ; is automatically reset to "0".           [804.2]

```

9.7.2.7 Connecting the technology to the speed and position controllers

The position setpoint KK310 output by the technology acts as a setpoint for the position controller ⑥:

P190.1=310 (WE) ; Wire position setpoint [817.7] ==> [340.1]

The actual position from the motor encoder in slot C is wired to the position controller as an actual value:

P194.1=120 (WE) ; Wire actual position [330.8] ==> [340.1]

The enabling of the position controller [340.3] and the speed setpoint for control and homing modes [340.7] is performed exclusively via binector B305, which is output by the technology [817.7]. The two "enable position controller" commands [340.3] must be permanently set to "1" for this purpose:

P210.1=1	; Set enable 1 for position controller
	; permanently to "1" [340.1]
P211.1=1	; Set enable 2 for position controller
	; permanently to "1" [340.1]
P213.1=305 (WE)	; Enable speed setpoint for control mode
	; [817.7] ==> [340.7] (0/1 = position control/speed
	; control mode)

The speed setpoint output by the technology for the speed control modes "control" and "homing" [817.7] is connected to the speed setpoint input [340.7] downcircuit of the position controller:

P212.1=311 (WE)	; Wire speed setpoint for control/homing
	; mode [330.8] ==> [340.1]

Output signal KK131 of the position controller is connected to the speed controller input:

P220.1=131	; Wire position controller output to speed
	; controller [340.8] ==> [360.1]

9.7.2.8 Setting the parameters for the positioning modes

Nesting the positioning modes in the sampling time:

U953.32 is used to nest the positioning modes [802.8] in a sampling time. In the factory setting of this parameter (20), the positioning software does not run (see [702]).

U953.32=4	; Nest positioning modes in sampling time T4 (=24*T0=16*200µs=3.2 ms with converter frequency 5 kHz)

MDI block numbers 1 and 3 [823], which are selected by the switches on terminals 3 and 4 (Chapter 3), should be set as follows in our application example:

MDI block 1:

- ◆ First G function = 90 (absolute, not relative positioning)
- ◆ Second G function = 30 (100 % of the acceleration/deceleration set in MD18/MD19)
- ◆ Position setpoint (X) = 0 LU
- ◆ Velocity (F) = 5 000 000 LU/min (equals 500 000°/min = half the maximum velocity in MD23/2; equals 1389 motor revolutions/min)

NOTE

The velocity in the MDI block is defined in [10 LU/min] instead of [1000 LU/min] as in the machine data

U550.01=9030 (WE)	; Absolute positioning, 100 % acceleration override	[823.4]
U550.02=0 (WE)	; Position setpoint X=0	[823.5]
U550.03=500 000	; Velocity F=5 000 000 LU/min, input in [10 LU/min]	[823.6]

MDI block 3:

- ◆ First G function = 90 (absolute not relative positioning)
- ◆ Second G function = 30 (100 % of the acceleration/deceleration set in MD18/MD19)
- ◆ Position setpoint (X) = 16 200 LU (1620° in clockwise direction of rotation, equals 4.5 revolutions)
- ◆ Velocity (F) = 1 000 000 LU/min (equals 100 000°/min = 1/10 maximum velocity MD23; equals 277 motor revolutions/min)

U552.01=9030 (WE)	; Absolute positioning, 100 % acceleration override	[823.4]
U552.02=16 200	; Position setpoint X=16200 LU	[823.5]
U552.03=100 000	; Velocity F=1 000 000 LU/min, input in [10 LU/min]	[823.6]

9.7.2.9 Testing the positioning functions of the application example

Performing the homing procedure

- a) Note: The sequence of the homing procedure is described in function diagram sheet [821] and the Function Description in the "Motion Control for MASTERDRIVES MC and SIMATIC M7" manual /1/.
- b) Select "homing" mode at the switches on terminals 3 and 4 (see Figure 9-42).
- c) Acknowledge any active positioning warnings "Axxx" at the switch on terminal 5. The most important warnings are generated by the following error monitoring and "in-position monitoring" systems [818.5]. If necessary, select a more tolerant setting for the monitoring system temporarily by increasing MD14...MD17.
- d) Switch on the drive at terminal 8.
- e) Start homing with "jog forwards" (1 signal at terminal 7)
- f) Simulate a rough pulse at DIN 4 (0-1 edge reduces the velocity, 1-0 edge terminates referencing)
- g) Optimize the position controller loop gain. For example, the optimum setting for a two-axis demonstration pack is as follows:

P204.1=8,000 ; Loop gain factor for position controller [340.3]

Positioning with MDI on drive 2 (function diagram, sheet [823])

- a) Select MDI mode at the switches on terminals 3 and 4
- b) Select MDI block 3 at terminal 6
- c) Start the positioning movement using the START command at terminal 7
- d) The wheel now moves 4.5 revolutions to the right.
- e) Switch from MDI block 3 to MDI block 1 at terminal 6. The MDI block is permanently assigned with position setpoint X = 0 and five times the velocity F.
- f) Start another positioning operation. The drive now moves back to position 0 at 5 times the velocity (in the clockwise direction, because jog forwards = 1), i.e. through 5.5 revolutions.

9.7.2.10 Setting the parameters for the virtual master axis

Nesting the virtual master axis in the sampling time

The virtual master axis ⑨ [832] is a separate free block (which can be used independent of positioning and synchronization). It is activated with the following parameter setting and is nested in the same sampling time as the positioning system:

```
U953.34=4      ; Nest virtual master axis in sampling time T4
                (=24*T0=16*200µs=3.2 ms for converter frequency 5 kHz)
```

Input signal and enable signal of virtual master axis

The virtual master axis is enabled in this application example together with the start command for positioning (terminal 7; see Fig. 9-46). The input velocity setpoint is transmitted from the potentiometer at the analog input (terminal 9/10):

```
U689=18        ; Enable for virtual master axis [832.2] together with
                ; start command from terminal 7 [90]
U681=11        ; Velocity setpoint for virtual master axis
                ; [832.1] from potentiometer at analog input [80]
```

Rated velocity and acceleration ramp for virtual master axis:

The rated master velocity (maximum machine velocity) in this example is set to the same value as the maximum traversing velocity MD23 for positioning:

```
U682=1 000 000 ; Rated velocity for virtual master = MD23
                ; =10 000 000 LU/min (input in [10 LU/min]), this
                ; equals 2778 min-1 on the motor shaft (see
                ; P353) [832.2]
```

We will assume an acceleration time for the virtual master of 1 s for acceleration from 0 to the rated velocity of 10 000 000 LU/min configured in U682. This corresponds to the following acceleration:

```
U685=167       ; Acceleration for velocity ramp-function generator
                ; in virtual master axis [832.5] =
                ; (10 000 000 LU/min) / 1s = 166 667 LU/s2
                ; (input in [1000 LU/s2])
```

Setting the axis cycle length of the virtual master axis

The axis cycle length for the virtual master ACL_V in this application example is the same as the rotary slave axis length of 36000 LU defined in MD11 for the positioning functions; this equals 10 motor revolutions with 3600 LU each (1 LU = 0.1°):

```
U687=36 000    ; Axis cycle length for the virtual master [832.6]
                ; = 36000 LU equal to 10 motor revolutions each
                ; with 360.0° (1LU = 0.1°; see AVWF)
```

9.7.2.11 Testing the virtual master axis

- a) Start command = 1 (switch on terminal 7) on drive 2
- b) Set potentiometer to 10 V (corresponds to 100 %)
- c) Check the velocity setpoint of the virtual master axis at KK820 [832.8] (e.g. at r33.1 [30.2] if P32.1 = 820 is set)
- d) Start = 0
- e) Set potentiometer to 0 V
- f) Start = 1 ==> observe the reduction in velocity setpoint from 100 % to 0 % at r33.1: it takes 1 s (this is easier to visualize if the acceleration time is increased temporarily from 1 s to 10 s with U685 = 17).

9.7.2.12 Configuring the synchronization function

Nesting the synchronization function

The synchronization function ⑩ is nested as a "positioning mode" in our application example (see the section entitled "Synchronization mode - overview" in "Brief Description of the Technology Functions" and [802.8]), i.e. U953.33 can retain the factory setting of 20.

Wiring the master value for synchronization

The input path setpoint [834.1] is transferred from SIMOLINK receive double word 1 KK7031 [150.6] with the factory setting (U600.01 = 7031 and U606 = 0). The master value is therefore already correctly connected to the output of the virtual master axis - via the SIMOLINK (see steps 10 and 13 in application example 2).

Setting the master axis cycle length

The master axis cycle length [834.2] must be set to the same value as the axis cycle length of the virtual master (U687; see [832.6] and step 10 in application example 2):

```
U601=36 000 ; Master axis cycle length [834.2] = Axis cycle
; length of the virtual master [832.6]
; = 36000 LU = 10 motor revolutions each
; with 360.0° (1LU = 0.1°; see actual value weighting factor AVWF)
```

Setting the synchronization mode

Angular synchronization 1:1 is to be used in our example (no engaging/disengaging cycle, no gearbox, no cam). This mode is already initialized with the factory settings: [Operation] = 0 [834.5] and [FUNCTION] = 0 [836.4].

Setting the slave axis cycle length

The slave axis cycle length [836.4 and 836.6] has already been set correctly to 36000 LU during configuration of the positioning system with machine data MD11 (step 6 in application example 2).

Setting the parameters for position correction

The control inputs of the position correction system [836.4] are already wired appropriately with the factory settings (for the use of a motor encoder). The normalization of the output velocity setpoint MD23 [836.7] was also already set to the correct value when the machine data were configured (step 6 in application example 2).

9.7.2.13 Configuring the SIMOLINK master

We are currently concerned with drive 2 of the application example, which calculates the virtual master axis ⑨ and handles the SIMOLINK dispatcher function.

The SIMOLINK drive interface is described in detail in the "Communication" chapter of the Compendium and in [140...160]; the hardware commissioning is detailed in the instruction manual of the SLB board. The SIMOLINK master is configured from the "board configuration" parameter menu (see the "Parameterizing Steps" of the Compendium). In our example, drive 2 is only required to send two double words, i.e. the path setpoint of the virtual master axis and a reserve word (unused). Both drives receive the path setpoint of the virtual master axis from SIMOLINK (including the master itself - this safely eliminates a deadtime difference between the path setpoints for drives 1 and 2).

```
;      Configuration of the SIMOLINK master (dispatcher)
P60=4      ; Select the parameter menu "board configuration"
P740=0      ; The dispatcher always has SIMOLINK address "0"
P741=100ms  ; Message timeout
P742=1      ; "Low output power" adequate with
              ; short cable
P743=2      ; No. of stations = 2 drives
P745=2      ; 2 channels (i.e. 2 send messages, 32 bits each)
              ; per station; all stations must have same setting
              ; based on the station that sends most messages
              ; in this case the master: 1 word for path setpoint
              ; of virtual master axis, 1 word reserved
              ;
P746=3,20   ; Set cycle time 3.2 ms for SIMOLINK ==>
              ; every 3.2 ms the master automatically sends a SYNC
              ; message, which synchronizes the sampling times of
              ; all stations. P746 should be set to the same
              ; sampling time in which the synchronization is
              ; nested (U953.32=4)
P749.01=0,0 ; 1st SIMOLINK receive double word KK7031 [150.7] =
              ; channel 0 of station 0 (i.e. of master)
P749.02=0,1 ; 2nd SIMOLINK receive double word KK7033 =
              ; channel 1 of master                                [150.7]
P60=0      ; Exit board configuration
P751.1=817 ; Connect output path setpoint KK817
P751.2=817 ; of virtual master axis [832.8] to send channel 0
              ; of SIMOLINK (assign the same double-connector to
              ; send words 1 and 2)                                [160.1]
```

9.7.2.14 Setting the parameters for drive 1 (SIMOLINK slave)

In Sections 9.7.2.3 to 13, we commissioned drive 2 completely, step-by-step with its positioning functions and the virtual master axis. We can now focus on drive 1 and commission it for position control mode before we tackle the testing of the SIMOLINK and the synchronization functions.

The procedure for setting the parameters and commissioning the positioning functions for drive 1 is identical to the description in steps 3 to 12 for drive 2. You can omit steps 10 and 11, since the virtual master axis is not required in drive 1.

You can then configure the SIMOLINK drive interface for drive 1 as slave ("transceiver") as follows:

```
; Configuration of the SIMOLINK slave (transceiver) [140+150]
P60=4      ; Select the "board configuration" parameter menu
P740=1      ; SIMOLINK address of drive 1 (>0 = "transceiver")
P741=100ms  ; Message timeout
P742=1      ; "Low output power" adequate with
            ; short cable
P749.01=0,0 ; 1st SIMOLINK receive double word KK7031 [150.7] =
            ; Channel 0 of station 0 (i.e. of master) = path
            ; setpoint of virtual master axis in drive 2
P749.02=0,1 ; 2nd SIMOLINK receive double word KK7032 =
            ; channel 1 of master (reserve channel) [150.7]
P60=1      ; Exit board configuration
            ; SLB send words do not have to be wired, since
            ; drive 1 is only to receive, not send data
```

9.7.2.15 Testing the synchronization in the application example

Checking the SIMOLINK connection

Check whether the SIMOLINK fiber-optic cables are correctly connected "crosswise" to the SLB boards (each transmitter connected to the receiver of the other drive). If the wiring and configuration is correct, all 3 LEDs should flash on all SLB boards.

Start the virtual master axis on drive 2 with the start switch (terminal 7), and check at r750.01 and .02 [150.5] on drive 1 whether the virtual master value transmitted from drive 2 is received correctly.

The further procedure for testing the synchronization function is described below with reference to the example of the 2-axis demonstration pack. An LED light beam visible through both flywheels indicates that the synchronization function is operating correctly.

Establishing the correct start position through homing

Start several homing procedures for both drives, as described in step 9. Set the home position offsets with machine data MD4 = U501.04 by trial and error starting with value "0" until the desired starting position for the synchronization is reached (a reminder: a value of 3600 LU in MD4 means one motor revolution). With the 2-axis demonstration pack, the start position is as follows on the two flywheels:

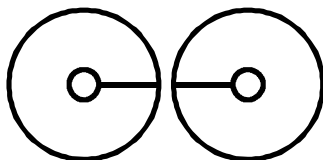


Fig. 9-49 Start position for synchronization with 2-axis demonstration pack

Testing the synchronization function

In this start position, you will see the LED through the drilled holes on both wheels.

When you have configured a suitable MD4 setting on both drives, please proceed as follows in order to test the synchronization via SIMOLINK:

- Set 0 V on the setpoint potentiometer of drive 2; this corresponds to a velocity setpoint of 0 %.
- Perform the homing procedure for both drives, in order to establish the start position in the sketch above (→ with the 2-axis demonstration pack, the LED beam must be visible through both flywheels).
- Cancel the start command for both drives (START = 0). Both drives are now in the start position, position-controlled.
- Start the synchronization on the slave drive (drive 1) with START = 1. It does not yet start to move, because the virtual master on drive 2 has not yet been enabled, and defines a path setpoint of "0".
- Start the master drive with START = 1 (drive 2). This activates the enable signal for the virtual master axis.
- You can now start both drives with the potentiometer and modify the speed (0...10 V equals 0...MD23 equals 2778 min^{-1} in r230 [360]).
- With all speeds, the LED beam should be visible through both flywheels if the synchronization is operating correctly.

End of application example 2

Application example 2 is now complete. When you have worked through all the steps in application example 2, you will have a sound overview of the "positioning" and "synchronization" functions and their connection and commissioning. You will have also received an introduction to the documentation available in an easy-to-understand example. The further steps required to commission your own customized application should now be much easier.

9.7.3 Synchronism with the virtual master axis by means of clock-synchronized Profibus (suitable for private study)

The objective is to use the equidistance (clock synchronism) of the PROFIBUS to achieve synchronism via the PROFIBUS only and to dispense with SIMOLINK, which has been necessary to replace the data for synchronous operation.

NOTE

PROFIBUS operation is only possible with an external bus master e.g. SIMATIC S7 (see MC Compendium Section 8.2.2. ff).

The "Drive ES Basic" tool is needed for configuring. A CBP2 is required for the "clock synchronous PROFIBUS" mode. The number of (clock synchronous) nodes is limited to max. 10..

Make sure that the baud rate of the PROFIBUS is set to 12 Mbit/s so that the data of the technology are transmitted fast enough. Also, select PROFIBUS as the synchronization source in parameter P744 (SIMOLINK board, function diagram 140).

Finally, activate the equidistance (clock synchronism) when configuring the hardware under the S7 project.

To achieve this, make the following connections on the drive control units:

- 1) Select PROFIBUS as the synchronization source (in function diagram SIMOLINK Board FP140)

WRITE	744	1	0	; Synchronization source PROFIBUS
WRITE	744	2	1	; Synchronization source PROFIBUS

- 2) Provision of transmit data (function diagram 125) on the "master axis" unit

The position and velocity setpoint of the virtual master axis (KK817 and KK820) and the generated sign-of-life signal (K255) are entered in the transmit data (parameter 734).

WRITE	734	15	820	; Velocity of the virtual master axis on PROFIBUS
WRITE	734	16	820	; Velocity of the virtual master axis on PROFIBUS
WRITE	734	11	817	; Position of the virtual master axis on PROFIBUS
WRITE	734	12	817	; Position of the virtual master axis on PROFIBUS
WRITE	734	13	255	; Sign-of-life signal on PROFIBUS

- 3) Connect the position setpoint extrapolator (U800.1, U800.2) and sign-of-life signal monitoring (U807) (FP 170) to the receive data of the PROFIBUS (FP 120).

WRITE 807 0 3013 ; Sign-of-life signal from PROFIBUS to sign-of-life monitoring

WRITE 2800 1 3041 ; Position of the virtual master axis from PROFIBUS to extrapolator

WRITE 2800 2 3045 ; Velocity of the virtual master axis from PROFIBUS to extrapolator

- 4) Sign-of-life signal valid (B0241) on communication fault (U801) of the position setpoint extrapolator connect, set axis cycle length (U802) of the extrapolator.

WRITE 2801 0 241 ; Sign-of-life signal to position extrapolator

WRITE 2802 0 4096 ; Axis cycle length extrapolator (based on U687!)

- 5) Connect position/velocity setpoint input of the synchronism (U600/U606) to the position setpoint output.

WRITE 2600 3 846 ; Position VM from extrapolator to SY input position setpoint

WRITE 2600 6 847 ; Speed setpoint VM from extrapolator to SY input V setpoint [%]

WRITE 2606 0 2 ; Switchover master value source

- 6) Set the nominal master velocity of the virtual master axis (U682 in function diagram 832) and the normalization velocity master in synchronism (U607.2 in function diagram 834) to the same value.

WRITE 2682 0 x ; Nominal master velocity of the virtual master axis

WRITE 2607 2 x ; Normalization velocity master in synchronism

7) Insert blocks into time slots

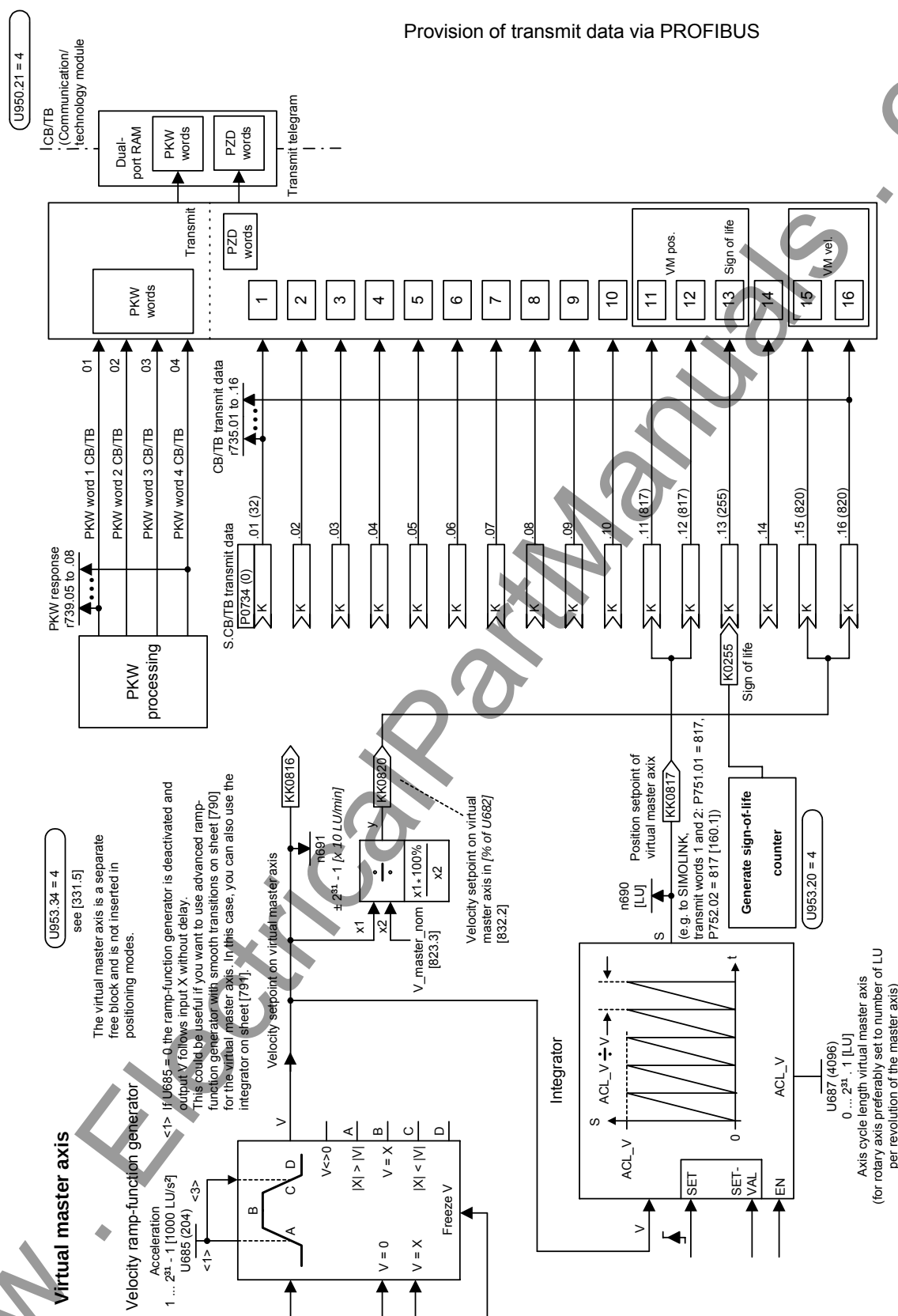
WRITE	2953	20	4	; Generate time slot sign-of-life (only on "master axis")
WRITE	2953	21	4	; Evaluate time slot sign-of-life (on "master" and "slave axis")
WRITE	2953	29	4	; Time slot position extrapolation (on "master" and "slave axis")
WRITE	2953	33	(4)	; Only insert in time slot with autonomous synchronism, with synchronism via the operating mode manager the factory setting 20 is retained
WRITE	2953	34	4	; Time slot "virtual master axis"
WRITE	2953	40	4	; Insert technology setpoints at the input of the position controller into a slower time slot (instead of factory setting =3)

8) Define the execution sequence

The consecutive events are inserted in the time slots in such a way that they are also executed consecutively and with high priority (at the beginning of the time slot).

WRITE	2960	11	0	; PROFIBUS receive (factory setting: 110)
WRITE	2963	21	1	; Sign-of-life signal receive (factory setting: 3210)
WRITE	2963	29	2	; Position setpoint extrapolator (factory setting: 3290)
WRITE	2963	32	3	; Operating mode manager (factory setting: 3320)

Provision of transmit data via PROFIBUS





Setting of hardware configuration

As stated above, the master value (KK817, KK820, FD 832) and the sign-of-life signal (K255, FP 170) are routed to the PROFIBUS where they are read again and connected on.

To ensure that these values are correctly routed to the PROFIBUS and can be read again by it, first adapt the hardware configuration of Step 7. GMC control must continue to be operable as usual.

To be able to use GMC control at all, make sure that GMC Basic has been installed and that the project has been adapted using the example P7MC1_EX.

Good help with this is available in file "Getting_started_mc_10.pdf" on the DriveMonitor CD under path: Gmc\Getting_Started\English.

Please note that in the DB 100, you must enter the PKW and PZD addresses for the axis concerned in accordance with the hardware configuration.

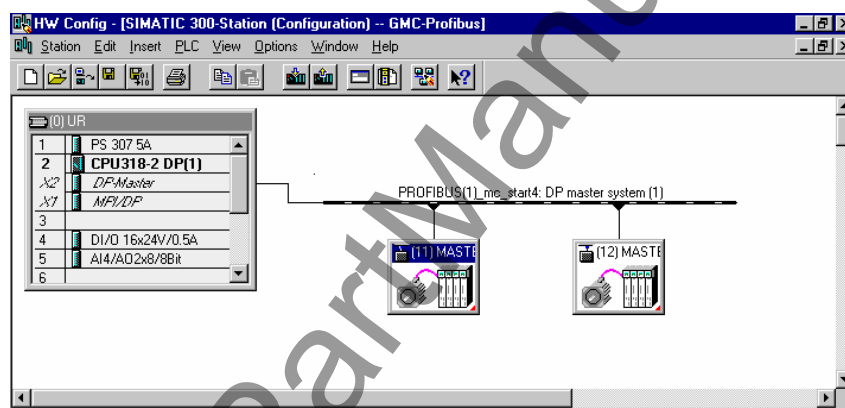


Fig. 9-50 Hardware configuration

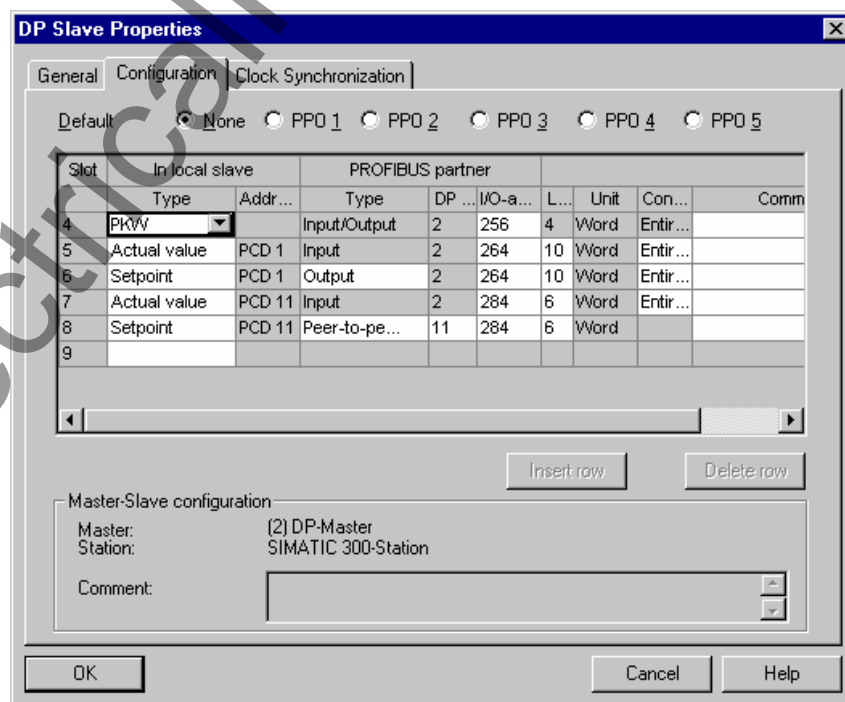


Fig. 9-51 Master axis CBP address 11

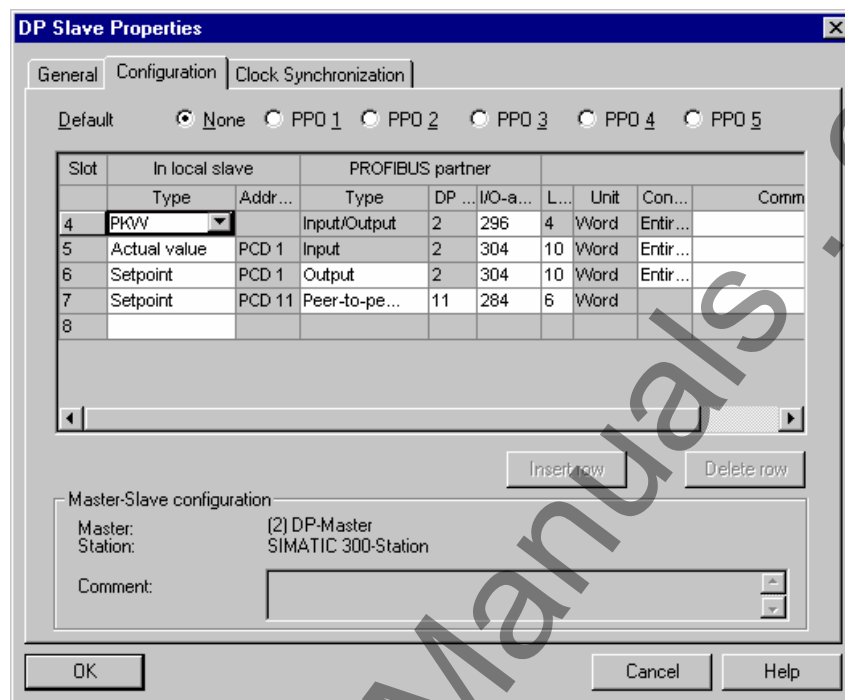


Fig. 9-52 Slave axis CBP address 12

Fig. 9-50 shows the hardware configuration. The left converter (drive unit), the master axis "dispatcher" (on it the virtual master axis is calculated) has PROFIBUS address 11. The right-hand converter, "transceiver" (it reads the position setpoint, i.e. the value of the virtual master axis from the "dispatcher") has address 12.

The additional six words to be sent (KK817, KK820 and K255) are routed to the bus from the "dispatcher" as the actual value. This is shown in Fig. 9-51 under slot 7.

They are then read by the "dispatcher" itself via slave-to-slave traffic (Fig. 9-51 Slot 8) and by the "transceiver" by slave-to-slave traffic (Fig. 9-52 Slot 7).

Via slots 4 to 6 (Fig. 9-51 and Fig. 9-52) PPO type 5 is emulated in the two converters.

370.0	X_axes.i_number_axes	INT	1	2	Number of axes in GMC_DB_ORG
372.0	X_axes.X_axis1.i_axis_type	INT	0	2	<1> = M7, <2> = MCT, <3> = M7/MCT
374.0	X_axes.X_axis1.i_dbw_no_cmd	INT	0	0	Pointer of the commands
376.0	X_axes.X_axis1.i_m7_no	INT	0	0	Number of the M7 (1..4)
378.0	X_axes.X_axis1.i_log_axis_no	INT	0	0	Logical axis number 1..n
380.0	X_axes.X_axis1.i_profibus_addr	INT	0	11	MCT PROFIBUS address
382.0	X_axes.X_axis1.i_ppkw	INT	0	256	I/O area, PKW address of the MCT
384.0	X_axes.X_axis1.i_ppzd	INT	0	264	I/O area, PZD address of the MCT
386.0	X_axes.X_axis1.res7	INT	0	0	
388.0	X_axes.X_axis2.i_axis_type	INT	0	2	<1> = M7, <2> = MCT, <3> = M7/MCT
390.0	X_axes.X_axis2.i_dbw_no_cmd	INT	0	0	Pointer of the commands
392.0	X_axes.X_axis2.i_m7_no	INT	0	0	Number of the M7 (1..4)
394.0	X_axes.X_axis2.i_log_axis_no	INT	0	0	Logical axis number 1..n
396.0	X_axes.X_axis2.i_profibus_addr	INT	0	12	MCT PROFIBUS address
398.0	X_axes.X_axis2.i_ppkw	INT	0	296	I/O area, PKW address of the MCT
400.0	X_axes.X_axis2.i_ppzd	INT	0	304	I/O area, PZD address of the MCT
402.0	X_axes.X_axis2.res7	INT	0	0	

Fig. 9-53 DB 100 in data view

Activation of the equidistance

For GMC to run correctly, as stated above, you must make a few changes in DB 100.

In DB 100, go to the end of the block in declaration view and copy the last line, that is axis 1 and rename the copy axis 2. After that, you can switch to data view and continue to the end of the block. Fig. 9-53 shows the end of block DB 100 in data view. Here you must enter the number of axes, the PROFIBUS address of the axes, and the base address of the PKW and of the PZD range for the axis in question. This information must match that of the hardware configuration.

In the hardware configuration, a window is opened on the CPU under X2 DP master with the right mouse button. In it you can select the object properties.

- ◆ A new window opens, in which you can click on PROFIBUS properties.
- ◆ Click on Properties again.
- ◆ Click on the tab card System settings.
- ◆ Select the transmission rate 12 Mbit/s.
- ◆ Click on Options.
- ◆ Activate "Equidistant Profibus" in the window "Constant Bus Cycle Time" (cf.: Fig. 9-54).

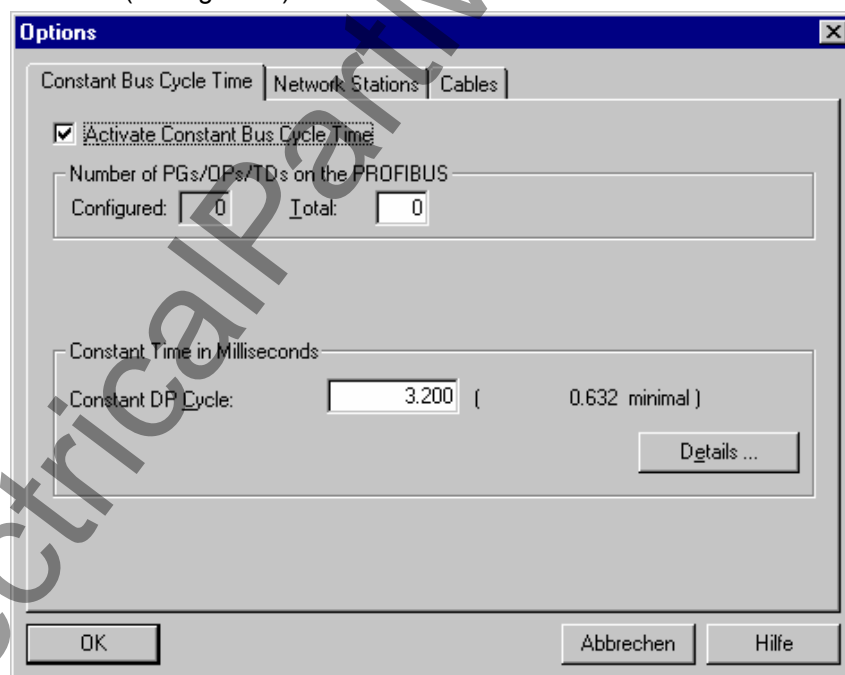


Fig. 9-54 Activation of the equidistance on the CPU

On the converter, you must also activate the equidistance. To do that, click on the converter with the right mouse button and select the tab card "Clock Synchronization" (Fig. 9-55).

Now synchronize the drive with the equidistant DP cycle. Then click on "Alignment".

Proceed in the same way for the second converter.

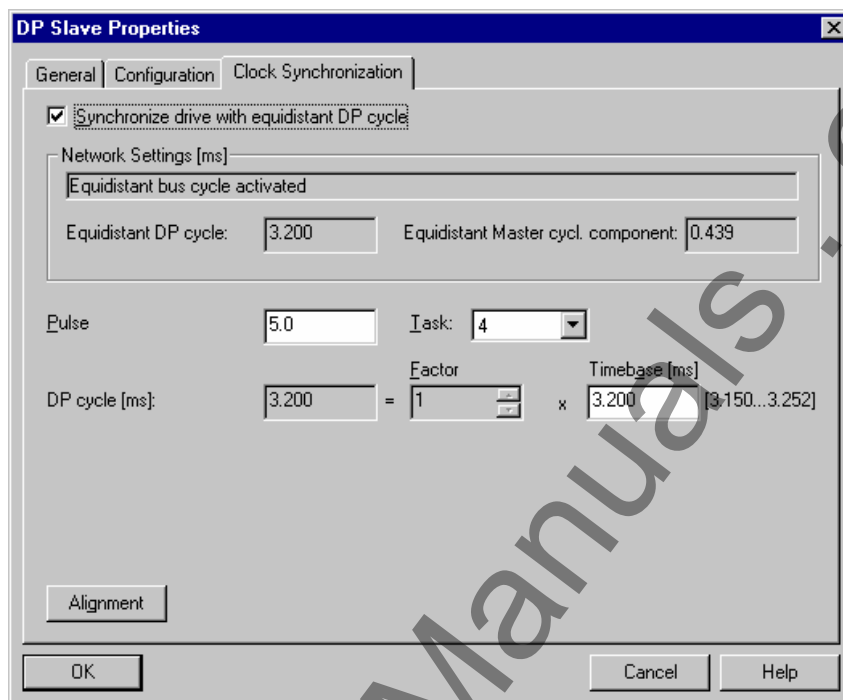


Fig. 9-55 Activation of the equidistance on the converter

On tab card "General" you can check the baud rate. If it is not 12 Mbit/s, you can set it under Properties, Network settings.

Please note:

If the CPU is reset, communication via your PROFIBUS interface is not possible.

You must load the hardware configuration into the CPU via the MPI interface. After that, communication via the PROFIBUS interface will be possible again. If you reset the CPU via the PROFIBUS interface, communication is interrupted.

If you address the PLC via the MPI or PROFIBUS interface, you must select the PG/PC interface in the SIMATIC Manager under "Extras".

9.7.4 Roll feed

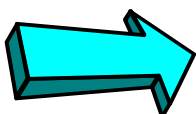
(under development)

9.7.5 Application using the SIMATIC S7 GMC software

(under development)

9.8 Commissioning the technology

9.8.1 Measurement and diagnostics resources



Commissioning step:

Familiarize yourself with the measurement and diagnostics resources:

The following measurement and diagnostics resources are available for MASTERDRIVES MC:

- ◆ **Faults, warnings, diagnostics:**
Read the section of the same name at the end of this chapter of the Compendium to find out which warnings and faults are generated by the technology and which technology signals you can track in the monitoring parameters.
- ◆ **Connector status in display parameters:**
You can connect any connector or binector to a display parameter, in order to track signals during commissioning and troubleshooting. These freely connectable display parameters are listed on sheets [30] and [705] of the function diagram.
Example:
U045=803 [705.7] => You can observe the status of binector B803 "engaging/disengaging cycle running" [834.5] in display parameter n046
- ◆ **Recording signals with the integrated trace function:**
A high-speed real-time trace function is available on the MASTERDRIVES MC for recording any connectors and binectors. You can operate this function conveniently with DriveMonitor. The trace function has the following features (see also the online help in DriveMonitor):
 - Flexible setting of time resolution from 500 µs
 - 8 measuring channels
 - Memory capacity: over 10 000 samples per channel
 - Flexible setting of zero offset and gain for the signal display across large range
 - Settable
 - Flexible setting of trigger condition (trigger signal, trigger threshold, post/pre-trigger)

NOTE

Binectors that are to be traced must first be entered in a connector using a binector/connector converter [720].

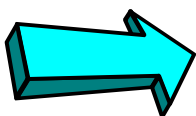
◆ **Recording signals with an oscilloscope or continuous-line recorder:**

The analog outputs on the converter terminal strip [80] and terminal expansion boards EB1 and EB2 [Y01...Y08] are available for this purpose.

◆ **Checking high-speed switching operations on the PMU:**

Thanks to its extremely high refresh time, the 7-segment PMU parameter monitoring unit is ideal for checking high-speed switching operations. You can monitor short-term level changes in all positioning control and status signals on the PMU in n541.01... .04. This is not possible on the OP1S or in DriveMonitor because of the serial transfer times.

9.8.2 Commissioning the technology



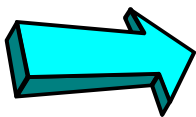
Commissioning step:

Use an application example to get started with the technology:

If you are not yet familiar with the positioning and synchronization functions in MASTERDRIVES MC, you should study the example nearest to your application in the "Application examples" section using one or two MASTERDRIVES MC units and one or two idling motors (you only need 2 drives for synchronization). In application example 2, for example, you are shown how to operate 2 MASTERDRIVES MC converters in position control and synchronization mode. The complete actuation system is implemented via the converter terminal strip (using switches and a potentiometer). The application example also guides you through the available documentation and the relevant function diagrams.

The section below provides you with a general commissioning guide which takes you step-by-step through the commissioning procedure. It is impossible, of course, to cater for all special applications in this guide.

9.8.3 Checking the speed/position controller



Commissioning step:

Check the speed/position controller:

If you are in any doubt as to whether the correct speed/position controller, the correct encoder cable or the correct sensor board are mounted, you should carry out the following checks:

Checking the encoder:

- ◆ For 1FK6 and 1FT6 motors, optical sin/cos encoder ERN1387/1381 is identified by the appearance of "Optical Encoder" on the nameplate.
- ◆ For 1FK6 and 1FT6 motors, multiturn absolute encoder EQN1325 is identified by the appearance of "Absolute Encoder" on the nameplate.
- ◆ For 1FK6 and 1FT6 motors, the resolver is identified by the appearance of "Resolver" on the nameplate or by the absence of an encoder type designation on the nameplate.

Checking the encoder cable for 1FK6, 1FT6 and 1PA6 motors:

- ◆ The correct encoder cable for the **resolver** can be identified from the order number "6FX□002-2CF01-□□□0", which is printed in red on the cable (□ = option and length specification).
- ◆ The correct encoder cable for the **ERN1387/1381** can be identified from the order number "6FX□002-2CA31-□□□0", which is printed in red on the cable (□ = option and length specification).
- ◆ The correct encoder cable for multiturn absolute encoder **EQN1325** can be identified from the order number "6FX□002-2EQ00-□□□0", which is printed in red on the cable (□ = option and length specification).
- ◆ The correct encoder cable for the **pulse encoder** (with unipolar HTL signals) can be identified from the order number "6SX7002-0□H00-□□□0", which is printed in red on the cable (□ = option and length specification).

The connector and terminal assignments on the motor and sensor boards are described in Catalog DA65.11.

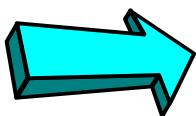
Checking the sensor board:

MASTERDRIVES MC automatically detects an installed sensor board. You can check in display parameter r826 whether the correct sensor board is installed (see parameter list). The sensor boards have the following codes:

- ◆ 111 = SBP (suitable for ROD431 etc.)
- ◆ 112 = SBM (suitable for ERN1397, ECN1313, EQN1325, SSI encoders from Siemens, Fraba, TWK, TR, Stegmann, linear scale LC181 etc. The ASIC chip on the SBM should have firmware version V1.3 or higher.)
- ◆ 113 = SBM2 (same as SBM with additional analog fine resolution on the board)
- ◆ 114 = SBR1 (for resolver, without pulse encoder simulation)
- ◆ 115 = SBR2 (for resolver, with pulse encoder simulation)

Check the switches and hardware parameters on sensor boards SBx if special encoders or third-party motors are used, as described in the "Encoder evaluation" sections in "Brief description of the technology functions" and in the hardware instruction manual for the SBx board.

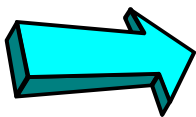
9.8.4 Defining the actual speed value normalization

**Commissioning step:****Set the actual speed value normalization with P353:**

First, set the actual speed normalization in parameter P353 [20] as follows:

- ◆ Define the maximum velocity (limit velocity), tolerated by the mechanical system of your machine, which must not be exceeded under any circumstances.
- ◆ In P353 enter the speed in [min⁻¹] at which the motor encoder rotates at this maximum velocity (allowing for transmission ratios, spindle pitches, etc.). Before entering the value in P353 you must change to the "drive settings" parameter menu with P60 = 5. You must exit this menu with P60 = 1 when the input is complete.
- ◆ If you use an external machine encoder for position sensing, enter in P355 the speed in [min⁻¹] at which the external encoder rotates at the maximum velocity.

9.8.5 Commissioning the MASTERDRIVES basic functions



Commissioning step:

Commission the MASTERDRIVES basic unit:

NOTE

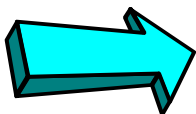
The following commissioning guide assumes that you have commissioned the MASTERDRIVES basic functions completely as described in Chapter 6 of this Compendium.

Commission the basic drive functions by following the steps below:

- ◆ Parameter reset (factory setting if necessary)
- ◆ Board configuration
- ◆ Drive settings (enter hardware and motor data)
- ◆ If possible, decouple the drive from the working machine and test it in speed control mode; optimize the speed controller

Set, commission and test the parameters for the communication functions (in the "board configuration") parameter menu (if field bus CBx and/or SIMOLINK SLB interface installed).

9.8.6 Defining the length unit LU



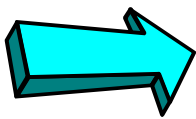
Commissioning step:

Define the length unit LU:

Define the length unit (LU) in which you want to define the positional values. Please note that the position is defined via field bus and parameter without decimal places in [LU]. For example, if you want to define the target positions to a resolution of 0.001 mm, the length unit 1 LU = 1 µm. If you want to define a position setpoint of 12.345 mm, MASTERDRIVES MC expects the setpoint 12345.

With pure synchronization axes (electronic shaft/gearbox), you will normally choose one increment of the position encoder as the LU (e.g. 1/4096 of an encoder revolution with P171 = 12 [330.3]).

9.8.7 Defining the actual value weighting factor (AVWF)



Commissioning step:

Check the resolution and value range of the actual position (P171):

Resolution of the actual position

NOTE

In the factory setting, the resolution of the actual position value of the motor encoder after shift division is 4096 steps per encoder revolution [330.4]. This is adequate for most applications.

In some documents the german abbreviation IBF may be used instead of AVWF.

The following section describes exceptional circumstances where the resolution has to be increased or reduced with P171 [330.3].

The resolution of the actual position on the encoder after shift division and before multiplication by the AVWF P169/170 is 4096 increments per encoder revolution in the factory setting. The rotor position KK090 [500 and 330.1] is resolved with 2^{32} steps per encoder revolution. This generates a positional value with 4096 increments per revolution as a result of the shift division by 2^{20} [330.4], due to the factory setting $P171 = 12$, i.e. the resolution is 12 bits. You will find detailed information on position detection in the section entitled "Position sensing system for motor encoder".

Please note that although the actual position of the position sensing system has a range of 2^{32} LU, it is limited by the technology to a value range of -999 999 999 ... +999 999 999 LU [815.4].

Resolution with resolver

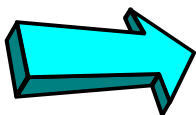
This resolution of the actual position value after shift division [330.3] of 4096 increments per motor revolution is exactly the same as the resolution implemented by the measuring system with a 2-pole resolver, and can virtually always remain unchanged when a resolver is used. Only if extremely long traversing distances are involved, can it become necessary to reduce the resolution (see following example):

Example where the position resolution must be reduced with $P171 < 12$:

- ◆ LU = 1/4096th of an encoder revolution selected
- ◆ The traversing distance is over 244 000 encoder revolutions
- ◆ => The traversing range would no longer fit into the range of 999 999 999 LU in this case (999 999 999 LU traversing range / 4096 LU per encoder revolution = 244 140 encoder revolutions)

Resolution with optical sin/cos encoder

With optical sin/cos encoder ERN1387, the resolution of the measuring system is $2^{24} = 16\,777\,216$ steps per motor revolution: after "pulse quadrupling" (evaluation of zero crossings), each 2048 sine and cosine periods per revolution produce a "digital rough resolution" of 8196 steps per motor revolution. The analog fine amplitude evaluation of the sine/cosine signals resolves each quarter period with a further 2048 steps. If you want to use the full resolution of the ERN encoder for the positioning and synchronization functions, you must set $P171 = 24$, which reduces the resolution of the rotor position signal KK090 from 2^{32} to the actually implemented resolution of 2^{24} . However, if you use this setting and set the actual value weighting factor (AVWF) $LU = 1$ encoder increment, you can only achieve traversing ranges (with linear axes) or axis cycle lengths (with rotary axes) of 59.6 encoder revolutions, as the position setpoints and actual values can otherwise no longer be represented by the numeric range from -999 999 999 ... +999 999 999 LU (numeric range $999\,999\,999\text{ LU} / 16\,777\,216\text{ LU per revolution} \rightarrow 59.6\text{ revolutions}$). For larger traversing distances, you must therefore reduce the resolution "artificially" by setting $P171$ to values < 24 or by selecting a smaller AVWFactor (i.e. a larger length unit LU).



Commissioning step:

Enter the actual value weighting factor (AVWF) (P169/P170 or P152/P153):

Entering the actual value weighting factor (AVWF)

For the actual value weighting factor (AVWF), you must enter the number of length units per actual position increment (LU/increment) in parameter $P169/P170$ if you use the position sensing system [330] for the motor encoder in slot C, or in $P152/P153$, if you use an external machine encoder [335].

NOTE

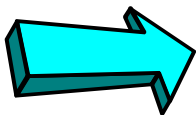
- ◆ You must always enter 8 decimal places in the AVWF in $P169$ or $P153$, padding any non-significant decimal places with "0" (e.g.: $AVWF = 1.5\text{ LU/increments} \Rightarrow P169 = 1. P170 = 5000000$; $P170 = 5$ would result in an incorrect AVWF of 1.00000005 !!)
- ◆ Please check carefully whether the AVWF has been entered correctly. Many of the parameter and machine data settings described below refer to the length unit LU and must be entered again if the actual value weighting factor (AVWF) is changed.

It is normally recommended to use a length with three decimal places for the LU, e.g. $1\text{ LU} = 0.001\text{ mm} = 1\text{ }\mu\text{m}$ or $1\text{ LU} = 0.001^\circ$. This is particularly important if you use the SIMATIC S7 "Motion Control Configuring Package" software /1/, in which the OP screens always use 3 decimal places for length parameters.

With pure synchronization axes (electronic shaft/gearbox), the actual value weighting factor (AVWF) can frequently remain in the factory setting AVWF = 1.0, i.e. 1 LU = 1 position encoder increment.

The "Position sensing system for motor encoder" section includes an example calculation for determining the AVWFactor.

9.8.8 Defining the maximum traversing velocity

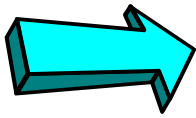


Commissioning step:

Define the maximum traversing velocity (MD23, P205):

Enter the maximum traversing velocity, which you defined in Section 1, in P205 [340.2] and machine data MD23 (P550.23 [804]) in the unit [1000 LU/min]. MD23 should not be modified again unless absolutely necessary, since this machine data is a normalization value for the velocity setpoint output [817] and the acceleration ramps (MD41, MD42 and MD43) for the speed control modes "control" and "homing", and must correspond to the reference speed P353.

9.8.9 Procedure for using the "GMC-BASIC" S7 software



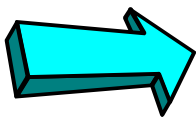
Commissioning step:

Configure the technology for the use of the SIMATIC S7 "Configuring Package" software:

If you use the GMC-BASIC software in the "Motion Control Configuring Package" /1/ for the SIMATIC S7, you should proceed as follows:

- ◆ Use DriveMonitor to download the parameter set shown in [806].
- ◆ The process data interface to the MASTERDRIVES converter (described in the "Control and checkback signals" chapter of the Function Description in manual /1/) is then set up via PPO type 5 (10 process data words with 16 bits each in send and in receive direction)
- ◆ This message assignment allows all the technology functions to be actuated from the SIMATIC S7 system. The message assignment can also be useful if you do not use the GMC-BASIC software (if a SIMATIC S5 or third-party controller is installed or if you use a field bus other than PROFIBUS-DP, e.g. CAN bus or USS).
- ◆ If you use an external machine encoder ("direct position sensing system") instead of the motor encoder, set the external machine encoder parameters in [815] as described in Section 10.
- ◆ The further commissioning procedure is performed from the SIMATIC S7 via PROFIBUS-DP. Direct parameter modification on the MASTERDRIVES unit is only required in exceptional circumstances.

9.8.10 Defining the positioning input signals



Commissioning step:

Define the positioning input signals:

- ◆ **Control commands**
- ◆ **Mode selection**
- ◆ **Velocity override**

All input signals of the technology can be "wired" flexibly using the BICO technology, e.g. from PROFIBUS-DP or the converter terminal strip. A hybrid configuration is also possible, with some signals being transmitted from the field bus and others from the MASTERDRIVES terminals.

Define which positioning input signals you need, and specify where they are transmitted from [809].



All control and checkback signals for positioning are described in detail in the "Control and checkback signals" chapter of manual /1/; in the subsequent chapters of the Function Description you will also find timing charts detailing the control sequences for each of the modes. Special and exceptional situations are also described.

Control signals for positioning

If you want to define the control signals using the individual binectors, which can be selected with U710, (i.e. if U530 = 860 [809.7]), you must nest the "generation of positioning control signals" block in a sampling time with U953.32 (recommended value = 4).

Defining the modes

The first step is to define which modes you want to activate with [MODE_IN]. You will find detailed information on the individual modes in the "Function description" Chapter of the "Motion Control for MASTERDRIVES MC and SIMATIC M7" manual /1/.

For example, if you only want to perform point-to-point positioning with incremental encoders, you require at least modes 2 and 3 ("homing" and "MDI"). In step 2 of application example 2, you will find a description of a simple method of mode activation via the converter terminals.

Check that the modes are defined correctly with display parameter n540.14 [809.8].

Defining the control commands and the override

Define which positioning control commands you want to use [809] and where they are to be transmitted from. For example, if you only want to use the point-to-point positioning (MDI) and homing modes, you only need to wire the following control commands in the simplest case (see also application example 2, step 2):

- ◆ Jog forwards [J_FWD] or
- ◆ Jog backwards [J_BWD], depending on the homing direction
- ◆ START [STA]

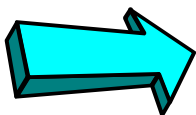
Control signals for synchronization

Decide whether and how you want to define a velocity override, or whether this can remain in the permanent factory setting of 100 % [809.1].

Check that the control commands are defined correctly in display parameters n541.01 and n541.02, and check the override input in n540.11.

If you only use the synchronization function, please refer to the "Synchronization mode - overview" section in "Brief description of the technology functions" to find out which signals you need on sheet [809]. The special control commands for synchronization are shown in [832...839].

9.8.11 Defining the positioning status signals



Commissioning step:
Define the positioning status signals you need:

All output signals of the technology can be "wired" flexibly using the BICO technology, e.g. to PROFIBUS-DP or the converter terminal strip.

Define which positioning input signals you need, and specify where they are transmitted to [811].



All control and checkback signals for positioning are described in detail in the "Control and checkback signals" chapter of manual /1/.

Status signals for positioning

For example, a simple linear axis with MDI positioning and resolver might use the following checkback signals:

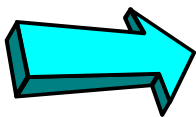
- ◆ Axis referenced [ARFD]
- ◆ Overtravel [OTR]
- ◆ Function terminated [FUT]
- ◆ Destination reached, axis stationary [DRS]
- ◆ Mode checkback [MODE_OUT]
- ◆ Start enable [ST_EN]

In this case, a movement should only be started from the external machine controller with the start command [STA] if the axis is referenced [ARFD], the required mode has been confirmed with [MODE_OUT] and the start enable [ST_EN] is active. Proper termination of the traversing movement is indicated by the status signals [DRS] and [FUT] using a handshake method.

Status signals for synchronization

If you only use the synchronization function, please refer to the "Synchronization mode - overview" section in "Brief description of the technology functions" to find out which signals you need on sheet [811]. The special status signals for synchronization are shown in [832...839] (binectors B800...B820).

9.8.12 Connection and parameters of the position sensing system



Commissioning step:

Set the parameters of the position sensing system:

Follow function diagram [815] in order to connect the technology with the position sensing system for the motor encoder [330] or an external machine encoder [335]. On this sheet, you will find two columns for each type of position sensing with the parameter settings (one for the control signals and one for the evaluation signals). In the factory setting, the motor encoder is prewired to a large extent. If you use the motor encoder – this is normally the case – you only need to change the following parameter settings:

```
; Position sensing for motor encoder in slot C [330]:
-----
; Connect to technology [330] <==> [815] [836]:
U535=120      P172=302
U539=122      P174=301
               P184=303

; Additional connections usually required for synchronization
; [330] ==> [836.4] (not required for positioning!)
U665=122      ; Meas. pos. for pos. correction [330] ==> [836.4]
U671=120      ; Use actual position as initial set value
               ; for slave path setpoint

; Configuration for incremental encoder (resolver, ERN encoder...):
P183=xx01     ; No homing (e.g. for
               ; roll feed or synchronization)
P183=xx11     ; Home position to right of rough pulse/ BERO
               ; (see also MD5 and [821])
P183=xx21     ; Home position to left of rough pulse/ BERO
; Configuration for absolute encoder:
P183=xxx2     ; [330]
U950.19=3     ; Nest encoder detection in sampling time [260.8]
               ; If not standard encoder: set parameters as
               ; described in "Multiturn encoder evaluation"
```


If you want to use an external machine encoder for position sensing, the following parameter settings are required:

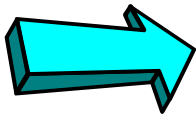
```
; Position sensing for external machine encoder [335]:
-----
; Connect to technology [335] <==> [815]:
U535=125      P155=302
U529=71       P156=302 (or =0 with absolute encoder)
U539=127      P157=301
U538=217      P158.01=303
U537.02=215   P158.02=304
               P167=303
               P162=308
               P160=307

; Additional connections usually required for synchronization
; [335] ==> [836.4] (not required for positioning!)
U665=127      ; Meas. pos. for pos. correction
U671=125      ; Use actual position as initial set value
               ; for slave path setpoint

; Configuration for incremental and absolute encoder
P166=xx01     ; No homing (e.g. for
               ; roll feed or synchronization)
P166=xx11     ; Home position to right of rough pulse/ BERO
               ; (see also MD5 and [821])
P166=xx21     ; Home position to left of rough pulse/ BERO

; Configuration for absolute encoder:
U950.18=3     ; Nest encoder detection in sampling time [270.8]
               ; If not standard encoder: set parameters as
               ; described in "Multiturn encoder evaluation"
```

9.8.13 Machine data input MD1...MD50

**Commissioning step:****Enter the machine data:**

Machine data MD1 to MD50 (parameters U501.01 to U501.50) are centralized settings required for positioning and synchronization from the perspective of the working machine and the mechanical transmission elements. The machine data are only active if they have been transferred with U502 = 2 when the drive is stationary (see [804]).

The machine data are listed in abbreviated format in [804].



In the "Machine data of the technology" chapter of the Function Description in manual /1/ you will find a detailed description of all machine data. Please note that the decimal points specified in some of the machine data there are not entered in parameters U501.01...50, e.g. max. traversing velocity MD23 = 10000 LU/min → Input on MASTERDRIVES MC: U501.23 = 10, in OP screen 10.000.

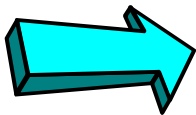
If you use the positioning functions, the following machine data are relevant:

Machine Data Overview for Positioning	
MD1, MD2, MD11	Encoder and axis type
MD3...MD7	Configuration of homing (only relevant for incremental encoders); see [821] and "Positioning block" section
MD12...MD17	Software limit switches (for linear axis) in addition to following error and "in position" monitoring
MD18, MD19, MD23	Velocity, acceleration and deceleration ramps for position control modes (see above for MD23)
MD41...MD43	Acceleration and deceleration ramps for speed control modes (control and homing)
MD21, MD29...37, MD46, MD48	Special machine data for roll feed only
MD20, MD24, MD25, MD44	Special machine data for automatic mode only
MD38...MD40	Backlash compensation
MD45, MD47	Configuration of special digital I/Os for positioning
MD49, MD50	Weighting of velocity and acceleration precontrol
MD10	Offset value for absolute encoder

If you only use the synchronization function, you only need to modify the following machine data (see also the "Synchronization mode - overview" section in "Brief description of the technology functions"):

Machine Data Overview for Synchronization		
	Synchronization as Positioning Mode	Synchronization as Free Block
Relevant machine data	MD11	MD11 Linear axis/rotary axis length
	MD49	[836.4]
	MD12 *) Software limit switches ...	MD49 Weighting of velocity precontrol [836.7]
	MD13 *) ... for linear axis	
	MD15 *) Following error monitoring - in motion	
	MD23	MD23

9.8.14 Connecting the technology to the speed and position controller



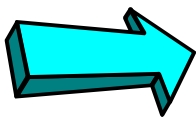
Commissioning step:

Connect the technology to the speed and position controllers:

[817] shows the setpoint outputs for the position controller and speed controller in addition to the enabling of the position controller by the technology. These connections are established to a large extent in the factory settings. Only the parameters below need to be modified (see the section entitled "Connecting the technology to the speed and position controllers" in application example 2):

```
; Connecting the technology to the position and speed controllers
P210.1=1, P211=1 ; Enable 1 and 2 for position controller
; permanently set to "1" [340.1]
P220.1=131 ; Wire position controller output to
; speed controller [340.8] ==> [360.1]
P194.1=120 (WE) ; Connect actual pos. from motor
; encoder [330] to actual value input [340.1] of
or ; position controller ...
P194.1=125 ; ... or actual position of external machine
; encoder, if one is used.
```

9.8.15 Setting the parameters for the positioning modes



Commissioning step:

Set the parameters for the positioning modes:

(you can skip this step if you only want to use the synchronization function)

First, nest the positioning block in a sampling time (otherwise it is not calculated). A suitable sampling time, for example, is T4 (= 3.2 ms with a converter frequency of 5 kHz):

```
U953.32=4      ; Nest positioning modes in sampling time T4
                (=24*T0=16*200µs=3.2 ms with converter frequency 5 kHz)
```

Configure the MDI block definition and selection for the "MDI positioning" mode [823]. For the first commissioning steps, it is recommended to use MDI block 1:

```
; Define MDI block 1 for the first commissioning steps
U710.09=1      ; Select MDI block number 1 [809.3] ==> [823.3]
U550.1         ; Define 1st and 2nd G function e.g. value=9030
                ; ==> Positioning in absolute dimensions,
                ; acceleration override = 100 %
U550.2         ; Define target position, e.g. value = 1000 ==>
                ; target position 1000 LU
U550.3         ; Define traversing velocity in [10 LU/min],
                ; e.g. velocity 100 000 LU/min ==>
                ; input value = 10 000 LU (must be less than
                ; MD23)
```

9.8.16 Safety information, hardware limit switches

Before you start the positioning operation, please read the following safety information:

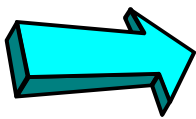


DANGER

- ◆ Please implement external control measures to ensure that the drive is isolated from the power supply immediately, and, if necessary, that the mechanical brake is engaged, in the event of hazardous conditions (e.g. safety guard open, hardware limit switch crossed, falling load hazard, etc.).
- ◆ Reduce the traversing velocity for the first commissioning steps by setting the velocity override [819] to a lower value, e.g. 1...4 %. You can monitor the current velocity override setting in display parameter n540.11 [809.8]. Virtually all traversing operations of the technology (except for home position reducing velocity and synchronization) are affected by the override.

You can use a contactor on the input or output side to isolate the drive from the power supply, or you can use option K80 "Safe STOP". Option K80 contains a special positive-action relay with a checkback contact which deactivates the actuation power for the power transistors (IGBTs). This prevents the rotation of the motor, however the motor is not electrically isolated from the power source. Option K80 is available for all MASTERDRIVES MC units except for the Compact PLUS AC units up to and including 4 kW and the Compact AC units. The "Safe STOP" function has been certified by the Institute for Work Safety.

9.8.17 Commissioning the positioning modes

**Commissioning step:****Commission the positioning modes:**

(you can skip this step if you only want to use the synchronization functions) ◆

Position encoder

First check the configuration of the encoder for position sensing with reference to the section entitled "Encoder evaluation and position sensing". If you use the EQN absolute encoder, for example, enter the correct baud rate and the suitable zero offset in P149. Check the function of the position encoder by operating the drive by hand, if possible, and monitor the actual position value at n540.03 [815.4]. For a linear axis, check the setting of the software limit switches again in MD12 and MD13.

Control

First operate the drive under pure speed control in "control" mode 4. You can move the axis in jog mode with no evaluation of the software limit switches (in software version 1.2 only fixed levels of 10 % and 100 % are implemented as jog setpoints).

Setup

You can then move the drive with "jog forwards" [J_FWD] and "jog backwards" [J_BWD] under position control in mode 1 "setup" [819]. The software limit switches are evaluated with linear axes, however with incremental position encoders they are only evaluated if a homing procedure was performed after the system was switched on.

Optimizing the position controller

You can now optimize the position controller in setup mode: set the loop gain factor P204.1 of the position controller [340.3] for the optimum dynamic motion characteristic. In special situations, it can also be practical to smooth the actual position and position setpoint, in order to achieve an optimum control response - in P195.1 and P191.1 [340.2]. The integral component of the position controller is not normally used, i.e. P206.1 can remain in the factory setting "0" [340.4].

Homing

If you use "homing" [821], first align the rough pulse switch (BERO) as described in the "Position sensing system for motor encoder" section in "Brief description of the technology functions". When you have selected the mode [MODE_IN] = 2 you can start the homing procedure with the control command "jog forwards" [J_FWD] or "jog backwards" [J_BWD]. MD4 allows the exact assignment of the position coordinate to the machine zero (if you change MD4, please adjust the software limit switches MD12/13).



You will find further information on "homing" mode in the "Brief description of the technology functions" section and in the "Reference point approach" chapter of the Function Description of manual /1/.

Setting the monitoring systems

In the section below entitled "General commissioning information", you can read how to temporarily define more tolerant settings for following error and "in position" monitoring (warnings A140...142) as long as the position controller has not yet been optimized.

Please consult the section below entitled "Help, my axis won't start!" if you have problems starting a traversing movement.

MDI positioning

By selecting mode 3 (= MDI) and defining the start command (0 => 1 edge at [STA]) you can start a traversing operation with the MDI block configured in "Setting the parameters for the positioning modes". If you enter other target positions in U550.2 [823.5], you can approach different destinations with a start command.

Speed precontrol

In many cases, an improved dynamic response can be achieved with even smaller overshoot in movements by activating the speed precontrol. To do this, set the following parameters:

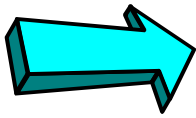
```
; Wiring the speed precontrol KK312
P209.1=312          ; Apply speed precontrol setpoint [817.7] [836.8]
                   ; to output of position controller [340.7]
```

The technology normally weights the precontrol value correctly, provided that MD49 is 100 % and P353 contains the exact motor speed that is reached at the maximum velocity in MD23. In special situations, scaling is also possible with MD49 [817.5]. If the speed is correctly precontrolled, output KK0132 and input r198 of the position controller will now only perform slight correction movements around "0" [340.5]. The trace function in the MASTERDRIVES is highly suitable for recording these signals. This function can be operated with DriveMonitor.

Jerk limiting

When using software versions < 1.30, if you set jerk limiting [817.4] via U505>0, you should not use the speed precontrol (KK312 = 0, see [817.6]), because otherwise the position control and speed control will "work against one another".

9.8.18 Configuring and testing the virtual master axis

**Commissioning step:****Commission the virtual master axis:**

(you can skip this step if you only want to use the positioning functions)

Setting the parameters for the virtual master axis

The virtual master axis [832] generates a path setpoint KK817 and a velocity setpoint KK816 [832.8] for drives that are to be operated in synchronism. These setpoints are normally distributed to the drives over the SIMOLINK drive interface. The common velocity ramp-function generator for all drives should be calculated on the drive on which the virtual master axis is activated. The velocity setpoint that the ramp-function generator is to reach can be transmitted via a field bus (PROFIBUS-DP etc.) or as an analog signal.

The velocity ramp-function generator [832.5] integrated in the virtual master axis can be used as a ramp-function generator for simple applications and extremely short cycle times (of several 100 ms). For more demanding technical conditions, you should use the "comfort ramp-function generator" [790], which operates with rounded ramps and provides a large variety of control features and selection of acceleration/deceleration times. You can connect the output of the comfort ramp-function generator to the virtual master axis with U681 = 571 [832.1] and U683 = 0 [U832.3].

The virtual master axis is a separate free block (it can be used independent of positioning and synchronization). It is activated with the following parameter settings and should be nested in the same sampling time as the synchronization function, e.g.:

```
U953.34=4      ; Nest virtual master axis in sampling time T4
                ; (=24*T0=16*200µs=3.2 ms with converter frequency 5 kHz)
```

Connect the desired enable commands to the virtual master axis with U684 and U689 [832.2].

If you use the percentage-weighted input setpoint (U683 = 0), you must enter the maximum machine velocity in U682 [832.2]. In many cases, this will be the value already entered in MD23 (see section 16 and [836.7] [804]; N.B.: MD23 is entered in [1000 LU/min.] but U682 is entered in [10 LU/min]!)

Select the master axis cycle length with U687. In many cases this will be the number of LUs per encoder revolution or per revolution of the gearbox output shaft, e.g.:

```
U687=4096      ; Example: axis cycle length[832.6] for the virtual
                ; master = 4096 LU equals 1 encoder revolution with
                ; P171=12 [330.3]
```

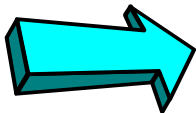

Testing the virtual master axis

Proceed as follows to test the output setpoints of the virtual master axis approximately before connecting them to the SIMOLINK drive interface:

- a) Apply a variable velocity setpoint to the setpoint input, e.g. from a potentiometer, or a fixed setpoint.
- b) Connect the output setpoints to display parameter [30], e.g.
 - ♦ P32.01 = 820 => velocity setpoint can be monitored in r33.01 in [%]
 - ♦ P44.01 = 817 => path setpoint can be monitored in r44.01 (value range 0...axis cycle length)
- c) Temporarily enter a long acceleration/deceleration time, e.g. from 20 s (U685 = 102 with an axis cycle length of 4096 LU and use of the integrated velocity ramp-function generator).
- d) Enable the ramp-function generator, change the velocity setpoint and check the output signals in the display parameters.

9.8.19

Setting the parameters for the synchronization block

**Commissioning step:****Set the parameters for the synchronization block:****Nesting the synchronization block in a sampling time**

(you can skip this step if you only want to use the positioning functions)

There are two ways to activate the synchronization block:

The synchronization block [834...839] is normally nested as a mode in the positioning block with parameter U953.32 [802.8].

In special situations, you can also activate the synchronization block as a completely independent free block with U953.33. In this case, the positioning block must be deactivated (U953.32 = 20). In this case, the synchronization block requires less computing performance, since the mode manager [802] is not active. However, the input and output signal processing by the mode manager is non-operational, e.g. the start command [STA] is not required in order to start the synchronization and the following error monitoring and indication is omitted.

You will find more detailed information in the "Synchronization mode - overview" section in "Brief description of the technology functions".

Selecting the input path setpoint for synchronization

U600 [834.1] is used to select the source for the path setpoint. If possible, the path setpoint from the virtual master axis on SIMOLINK receive channel 1 should be used. This selection is already configured with the factory settings U600.01 = 7031 and U606 = 0.

Wiring the master value for synchronization

The input path setpoint [834.1] is already received from SIMOLINK receive double word 1 KK7031 [150.6] with the factory setting (U600.01 = 7031 and U606 = 0). The master value is therefore correctly connected to the output of the virtual master axis - via the SIMOLINK interface (see the section entitled "Communication with the technology").

Setting the master axis cycle length

The master axis cycle length U601 [834.2] must be set to the same value as the axis cycle length of the master axis, e.g. to U687 (see [832.6] and section 16). The master axis cycle length is required by the DVAL block in order to reproduce the "position saw-tooth signal" of the master correctly.

Setting the synchronization operation

Set the desired synchronization operation [OPERATION] in U602 [834.5]:

U602=0	; Continuous operation with engaging/disengaging cycle
U602=1	; Engaging cycle (see "Synchronization block" section)
U602=2	; Disengaging cycle (see "Synchronization block")

Use U603 [836.4] to select the desired synchronization function [FUNCTION]:

U603=0	; Angular synchronization 1:1
U603=1	; Gear synchronization, gearbox: see [834.4]
U603=2	; Electronic cam/table synchronization
	; (see [839])

Setting the slave axis cycle length

For rotary axes, you normally set the number of LUs per position encoder revolution of the slave axis as the slave axis cycle length in U501.11 (MD11 [836.5 + 7]). MD11 is required in order to generate the correct "position setpoint saw-tooth signal" for the slave with the IVAL block [836.7] and the corresponding "actual position saw-tooth signal" with the actual position correction block KOR [836.8], in order to prevent overflows in the numeric range for a rotary axis.

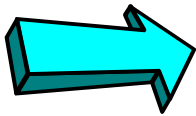
Setting the parameters for position correction

You only need to modify further parameters for the position correction [836] if you use the printing index correction or want to evaluate a synchronization signal.



You will find detailed information on the parameters for position correction in the "Synchronization functions" chapter of the Function Description in manual /1/.

9.8.20 Configuring and testing the SIMOLINK drive connection



Commissioning step:

Commission the SIMOLINK drive connection:

Setting the parameters for the SIMOLINK connection

(you can skip this step if you only want to use the positioning functions)

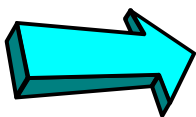
To configure the SIMOLINK master (dispatcher) and the SIMOLINK slaves (transceivers), follow the procedure described in sections 13 and 14 of application example 2.

Testing the SIMOLINK connection

First check that the SIMOLINK fiber-optic cables are connected correctly from the transmit connector of one SLB board to the receive connector of the next SLB board, and that the SIMOLINK ring to the dispatcher is closed. If the configuration and wiring is correct, all 3 LEDs should flash on all SLB boards.

To test the SIMOLINK connection, you should repeat the procedure described above for "Testing the virtual master axis" and check in r750 [150.5] that the output setpoints of the virtual master are received correctly by all drives.

9.8.21 Testing the synchronization functions



Commissioning step:

Commission the synchronization functions:

(you can skip this step if you only want to use the positioning functions)

To test the synchronization functions, proceed as follows:

- a) If possible, disconnect the motors from the working machine.
- b) Ensure that the initial speed and position setpoints for the master and slave drives are "0".
- c) If possible, enable only one slave axis initially for operation with position and speed controller.
- d) Temporarily set an extremely slow acceleration/deceleration ramp on the machine ramp-function generator (e.g. [832.5]).
- e) Move the master axis carefully from velocity "0" to low values and check whether the slaves follow the motion correctly.

NOTE**Start position for synchronization**

If you want to start the synchronization from a defined start position, you must first approach the position in a positioning mode and stop the drive at this point. You can then start the synchronization from speed "0".

With the "offset setting" [837], you can perform the orientation on-the-fly with reference to a synchronization marker after starting the synchronization mode. Although the offset setting is already implemented, it is not yet released for software version V1.2.

9.8.22 Help, my axis won't start!

If your positioning axis won't start, the cause may be one of the following:

- ◆ The start command [STA] is not active or not wired correctly. Check in n541.01 [809.7] whether the start command is correctly applied to bit 2 of the positioning control word. A movement is always started on a 0 → 1 edge of the start command.
- ◆ The start enable [ST_EN] is missing. Check in n541.03 [811.7] whether the start enable is indicated in bit 10 of the positioning status word. A missing start enable can have the following causes:
 - The start command [STA] is still set to "1". The start enable is not activated until the start command is reset to "0".
 - The correct mode is not selected (see below)
 - A positioning warning is active A129...A255 (see display parameter n540.26 [818.5] and the section entitled "Warnings, faults"). Remedy the cause of the warning and acknowledge it with a 0 → 1 edge of basic unit control bit 7 "acknowledge fault" [ACK_F] [180] or with the "P" button on the PMU.
 - The "reset technology" [RST] command is active. Check bit 6 of the positioning control word in n541.01 [809.7].
 - The "follow-up mode" command [FUM] is active. Check bit 5 of the positioning control word in n541.01 [809.7].
- ◆ The velocity override is = 0. Check n540.11 [809.8].
- ◆ The velocity setpoint defined in the NC block = 0.
- ◆ The correct mode was not selected with [MODE_IN]. Check the mode status [MODE_OUT] in n540.15 [811.4].
- ◆ An operating condition is missing (warning A130...A135). Check in r550 [180.7] that control bits OFF1, OFF2, and OFF3 are set to "0" and the inverter enable [ENC] is set to "1". You should also check the present converter status in r000.

- ◆ The axis is already in position. You can recognize this condition from the fact that the "function terminated" [FUT] status signal changes to "1" (or remains at "1") immediately after the 0 → 1 edge of the start command [STA]; you can monitor [FUT] in bit 27 of the positioning status word via n541.04 (preferably on the PMU). This can be the case with a rotary axis, for example, if the position setpoint that has already been approached is calculated as the target position by G90 through the "modulo rotary axis function" (e.g. the axis is positioned at 5°, and a position of 365° is defined → no G90 movement is performed). Check the effective position setpoint in n549.02 (the setpoint including all offset values and modulo functions). If you want to move a rotary axis through several revolutions, use relative positioning with G91; in this case, no modulo calculation is performed.
- ◆ The position setpoint is not correctly defined or is not wired correctly. Check the current position setpoint in n540.12 [823.6] for MDI mode.
- ◆ There is a converter fault. Bit 3 is enabled in basic unit status word 1 [200]. You can monitor the current fault and warning numbers in connector K0250 [510.4].
- ◆ There is a mismatch between the speed/position encoder, encoder cable and sensor board. Follow the instructions in "Checking the speed/position controller".

9.8.23 General commissioning information

- ◆ Changes to the axis type **MD1** and **AVWF factor** do not become effective until the power supply to the electronics is switched off/on.
- ◆ Changes to **machine data** parameters MD1...MD50 do not become effective until they have been transferred (at a standstill) with U502 = 2 [804.3].
- ◆ If a **positioning fault A129...A255** is active, you cannot start a movement until you have remedied the cause of the fault and acknowledged the warning. Diagnostic parameter n540.26 [818] indicates whether a positioning fault is active. This parameter indicates the number A129...A255 or "0" if no positioning fault is active.
- ◆ Until you have optimized the position controller, it may be necessary to enter more tolerant settings in the following monitoring parameters:
 - **Following error monitoring** by increasing MD14/MD15 (affects **A140, A141**)
 - **"In-position monitoring"** by increasing MD16/MD17 (affects **A142**)
- ◆ If necessary, reduce the traversing velocity for the initial commissioning steps by changing the **velocity override** (in the factory setting with U708 [809.1]).

- ◆ **Display parameters** n540 and n541 provide important diagnostic information (see also the section below entitled "Faults, warnings, diagnostics").
- ◆ You should only change the **speed/velocity compensation** in P353, MD23 and P205 and the AVWF factor in exceptional circumstances, since this de-aligns the loop gain factor P204 [340], the speed setpoint K311 [817.6], the speed precontrol KK312 [817.6] and the acceleration/deceleration ramps MD41/MD42, thereby necessitating a series of commissioning steps.
- ◆ Use the trace function integrated in the MASTERDRIVES MC if you want to record the timing sequence of important internal signals. Any connector or binector can be recorded with the **trace function**. The curves can be monitored in DriveMonitor (oscilloscope function). See also the section above entitled "Measurement and diagnostics resources".
- ◆ No double-connectors can be wired from MASTERDRIVES to PROFIBUS with software versions V1.1 and older of the PROFIBUS-DP interface **CBP**. The software version can be read out in r069. In this case, a double-connector/connector converter must be used.
- ◆ A **torque increase** is only possible if P128 is increased simultaneously ==> increase P263, P264 and P128 together.

9.9 Faults, warnings, diagnostics

All faults and warnings of technology option F01 are described in the "Warnings, Faults" section of this Compendium.

The main faults and warnings generated by the positioning modes are shown on sheet [818]. On sheet [839.8] you will find warnings generated by the cam.

Faults

Technology option F01 generates fault F063 if an attempt is made to nest a technology block in a sampling time when the technology is not enabled [800.3].

Warnings

The technology also generates warnings A129...A255. When a warning is active, a movement cannot be started until the cause of the warning has been remedied. Unlike basic unit warnings, technology warnings must be acknowledged before the traversing movement can be started again. The warnings are acknowledged using the normal fault acknowledgement feature of MASTERDRIVES, i.e. via bit 7 of basic unit control word 1 [180.4], e.g. from a digital input, serial interface (if so wired) or the PMU.

Diagnostic parameters

The following display parameters provide you with important diagnostic information:

- ◆ n500 : Error number of machine data check [804.07]
- ◆ n540.01....40 : Centralized diagnostic parameters of positioning modes [809, 815, 817, 818, 826]
- ◆ n540.26 : Current positioning warning
- ◆ n541.01...04 : Control and status signals [809, 811]
- ◆ n542.01...02 : Status of digital I/Os for positioning
- ◆ n668 : Status of cam table [839.3]
- ◆ r750 : Receive signals from SIMOLINK [150.5]
- ◆ r733 : Receive signals from communication interface, e.g. from PROFIBUS-DP [120.5]

The axis won't start

If your axis won't start, please consult the section entitled "Help, my axis won't start!" in "Commissioning the technology".

9.10 Hardware and software replacement measures

Hardware replacement measures.

If you replace the hardware – e.g. on repair – technology function F01 must be enabled on the MASTERDRIVES MC unit using the PIN number. Please consult sheet [850] in the function diagram to find out how to check whether the technology option is enabled and, if not, how to enable it.

Before replacing the hardware, copy all the MASTERDRIVES parameters into a download file using DriveMonitor or keep your archived download file at hand. After installing the new hardware, perform a "parameter reset to the factory settings" - with reference to the section of the same name in the "Parameterizing Steps" section of the Compendium. Then download the file with DriveMonitor, in order to restore the original parameter settings.

Please ensure if you use special signal sources that the switches on sensor boards SBx and terminal expansion boards EB1 and EB2 have the same settings as the old hardware.

Software replacement measures (booting new firmware)

Once technology option F01 has been enabled by PIN number [850], it remains enabled even if the software on the unit is replaced/updated. The PIN number is stored in a protected area of the parameter memory. It is therefore not necessary to enable technology option F01 again after replacing the software.

If you obtain new firmware, use DriveMonitor to copy all the MASTERDRIVES parameters into a download file before loading the firmware into the flash EPROM of the MASTERDRIVES unit via the "boot cable". Alternatively, keep your archived download file at hand.

After the booting procedure, perform a "parameter reset to the factory settings" - with reference to the section of the same name in the "Parameterizing Steps" section of the Compendium. Then download the file with DriveMonitor, in order to restore the original parameter settings.

The parameters in new software are normally fully backwardly compatible with all software versions. However, there are incompatibilities in the pilot versions before the release of the first official version of technology option F01 together with basic unit software version V1.2; see the section entitled "Modification history of technology option F01".

9.11 Modification history of technology option F01

The specified software versions are basic unit software versions. There are no special version numbers for technology option F01. The following section only lists software versions in which changes have been made to technology option F01.

9.11.1 Software version V1.0

(Start of Shipping: November 97)
Original version, for pilot customers only

9.11.2 Software version V1.1

(Start of Shipping: February 98)
V1.1 was only supplied to pilot customers.

- Modified parameters**
- ◆ U529.01....02: These parameters have been omitted and no longer need to be set. Basic unit connectors K030 (control word 1), K032 (status word 1) and K250 (fault/warning number) are now connected permanently to the positioning function (see "**Positioning input signals**").
 - ◆ U511...U520: These triple-index parameters are no longer used for **permanent MDI blocks 1...10**. Instead, they are stored in parameters U550...U559.
- New parameters**
- ◆ U550...U559: **Permanent MDI blocks 1...10**

9.11.3 Software version V1.2

(Start of Shipping: May 18th, 1998)

First officially released software version of technology option F01

Modified binector/connector numbers for position correction

The output binectors/connectors used for synchronization as a free block are now the same as when synchronization is nested as mode 11. It is only necessary to reconfigure the parameters when upgrading from V1.1 to V1.2 if synchronization is nested as a free block (i.e. if $U953.33 < 20$).

Modification	Reconfiguration Required when Upgrading from V1.1 ==> V1.2 (if Motor Encoder Sensor Board Used in Slot C)
KK310 replaces the old KK801 (corrected path setpoint)	P190 = 310 instead of 801
KK312 replaces the old K802 (slave velocity setpoint)	P209 = 312 instead of 802
K803 is omitted (slave acceleration)	--
KK301 replaces the old KK810 (position correction KOR)	P174 = 301 instead of 810
B307 new binector (enable measured value memory)	--
B304 replaces the old B801 (POV; position correction +)	P175.01 = 304 instead of 801
B303 replaces the old B802 (NOV; position correction -)	P175.02 = 303 instead of 802

Modified normalization in the synchronization function

All path and velocity signals are now fully normalized in the synchronization function, with the virtual master axis and in the positioning function, i.e.:

- ◆ The velocity signals are now normalized in [10 LU/min] instead of in [increments/sec].
- ◆ The path signals are now normalized in [LU] instead of in [increments].

Modified parameters in the synchronization function

Certain parameters of the synchronization function have been omitted and the corresponding parameters from the machine data (U501) used instead. This eliminates redundancy in the parameters (i.e. the representation of the same variable in two parameters).

The following parameters are affected:

- ◆ U501.11 replaces the old U670 (slave axis cycle now MD11)
- ◆ U501.23 replaces the old U668.1 (rated velocity now = MD23)

Modified parameters in positioning function

- ◆ U502 The machine data transfer now takes place with U502 = 2 instead of U502 = 1

Effect of the new normalization on synchronization parameters

(LU= Length Unit = the length unit based on the actual value weighting factor (AVWF))

- ◆ Velocities of virtual master
 $1000 \text{ [inc/sec]} \leftrightarrow 6000 \text{ [10 LU/min]}$
 => all parameters previously set have to be multiplied by 6; the value output or indicated in the monitoring parameters and connectors is greater by a factor of 6.
 This affects the following parameters and connectors:
 - U682
 - U679/KK818 or the connector connected via U680
 - Monitoring parameters n691 and n692
 - Connector KK816
- ◆ Acceleration of virtual master
 $1000 \text{ [inc/sec}^2] \leftrightarrow 1 \text{ [1000 LU/sec}^2]$
 => the parameter previously set (U685) should be divided by 1000
- ◆ Normalization of slave velocity setpoint ($\leftrightarrow$ speed/velocity precontrol)
 $1000 \text{ [inc/sec]} \leftrightarrow 60 \text{ [1000 LU/min]}$
 => the parameter previously set should be multiplied by 0.06.

NOTE

Machine data 49 is defined in 1000 LU/min, unlike the velocity setpoints and actual values.

Various enhancements

A series of enhancements has been made that does not affect the compatibility with older software versions:

- ◆ The velocity override can now be defined by any connector selected in U709.
- ◆ A series of new display parameters has been introduced (e.g. n540, n541, n542).
- ◆ The positioning status bits are now connected to individual binectors (B351...B562 [811.3])

9.11.4 Software version V1.3

(Start of Shipping: December 1998)

**Automatic mode
[826, 828]**

The "Automatic" and "Automatic single-block" positioning modes are now released. Automatic programs are input via the SIMATIC S7 using the GMC-BASIC standard software or via parameters U571 to U591 [828].

Electronic cam [839]

The electronic cam is released. Table input is performed via the SIMATIC S7 using the GMC-BASIC standard software or via parameters U630 to U646 (e.g. from an EXCEL table).

Acceleration pre-control [817]

Output of the acceleration pre-control by "Setpoint output and enable" of the technology option via connector KK313 is now implemented.

New integrator for the virtual master axis using the comfort ramp-function generator

A special integrator for implementing a virtual master axis using the comfort ramp-function generator [790] is now available in the free blocks on function diagram sheet [791].

New connector for position correction [843]

The new connector KK826 has been introduced for position correction. The deviation of the position of the printing or reference index from the setpoint position is available at this connector.

New "on the fly" referencing function for synchronization [843]

The new "on the fly" referencing function enables synchronization to a reference index (BERO or similar) to take place "on the fly" . It is no longer necessary to approach the reference index in positioning mode beforehand and then to change over to synchronization mode from standstill.

New synchronizing function "Synchronization to master value" [841]

The zero position of the slave axis can be matched to that of the master axis with this function. The speed and acceleration of the compensation movement necessary for this can be set via the new parameters U697.2 and U697.1. Synchronization is started by the binector that can be selected via the new parameter U676.

Introduction of 3 alternatives of setting the displacement angle for synchronization [841]

An angle displacement for synchronization can now be conveniently set in the following 3 ways:

- ◆ Setting an **absolute displacement angle** via the connector selectable by the new parameter U678.01
- ◆ Setting a **relative displacement angle** via connectors or parameters, which can be supplemented either in a positive or a negative direction, depending on the current zero position
- ◆ Setting a **displacement angle in inching mode** with a selectable variable speed (similar to a motorized potentiometer)

These displacement angle settings can be made in any size. Overflows of more than one slave axis revolution can be coped with. The displacement angle setting can be used, for example, to set up an index control system for printing machines.

Catch-up [837]

The newly introduced catch-up function enables a drive to be uncoupled from an angular-locked synchronizing multi-motor system (e.g. shaftless printing press) and to be operated autonomously at its own speed setpoint ("isolated setpoint"). It also enables a drive to halt at a specified angular position.

The drive can catch up to the speed of the running machine from the halt position or from its current speed in autonomous operation. After the catch-up command has been set, the drive accelerates to the machine speed and can then be coupled back accurately into angular synchronization after reaching speed synchronization.

NOTE

The "Catchup" function is not released until SW version V1.32.

Modified parameters

- ◆ U501.23 Machine data MD23 "Maximum traversing speed":
Can now be set up to 20 000 000 instead of up to 1 000 0900 [x100 LU/min]
- ◆ U501.10 Machine data MD10 "Offset for absolute-value encoder":
Now remains in non-volatile storage also after electronics power supply has been de-energized/energized.

New parameters

- ◆ U422.01-03 Input position actual value, actual speed and position setting value for the real master
- ◆ U423 Input smoothing for position actual value input of the real master
- ◆ U424 Deadtime compensation for real master
- ◆ U425.01-02 Axis cycle length at the input and output of the real master
- ◆ U426 Selection of connector "Set output" for real master
- ◆ U427 Smoothing of the speed signal for the real master
- ◆ U428 Speed normalization in the real master
- ◆ U625.01-03 Catch-up control word
- ◆ U626.01-03 Catch-up setpoints (setpoint speed and halt position)
- ◆ U627.01-02 Rounding time constants for the halt and catch-up ramp in the catch-up function
- ◆ U628.01-02 Deceleration/acceleration for the halt and catch-up ramp in the catch-up function
- ◆ U672 Selection of the "Set displacement" binector in the displacement angle setting
- ◆ U675.01-02 Selection of the "Enable position correction" and "Enable referencing" binectors for synchronization (parameter now indexed, index .02 is new)
- ◆ U676 Selection of the "Synchronize to master value" binector
- ◆ U677.01-02 Fixed setpoints for absolute and relative displacement angle
- ◆ U678.01-03 Selection of the following connectors for displacement angle setting: "Absolute displacement angle", "Displacement setting value" and "Relative displacement angle".
- ◆ U688.01-02 Fixed setpoints for catch-up speed and catch-up position
- ◆ U694.01-02 Selection of "Start+" and "Start-" binectors for relative displacement angle setting
- ◆ U695.01-03 Speed, deceleration and acceleration for "Displacement angle inching"
- ◆ U696.01-02 Selection of "Inching+" and "Inching-" binectors for displacement angle setting
- ◆ U697.01-02 Acceleration and variable speed for compensation movement of displacement angle setting
- ◆ U698.01-02 Selection of connectors for variable speeds for displacement angle setting
- ◆ U699.01-02 Operating mode of synchronization to master value and absolute displacement angle setting

9.11.5 Software version V1.4

(Start of delivery: 12.99)

General synchronism	Concerning the Synchronism function, please note that the smallest permissible time slot is T4 (P2953.33 = 4).
Real master [833]	The "Real Master" function block is no longer part of Technology Option F01, but is a free block. Set the setting value. This is only effective for the output. Additional connector KK624 as speed output in %.
Electronic cam [839]	The electronic cam has been extended from 2 to 8 tables and from 200 to 400 support points. The assignments X101-X150 = U640; Y101-Y150 = U645; X151-X200 = U641; Y151-Y200 = U646; X201-X250 = U632; Y201-Y250 = U637; X251-X300 = U633; Y251-Y300 = U638. Amendment to the configuration table (U615): 0 = 1 table with 400 support points; 1 = 2 table with 200 support points each; 2 = 4 tables each with 100 support points; 3 = 8 tables each with 50 support points; 4 = max. 8 tables with a total of 400 support points. New status binector: Stop at table end B834
Additive offset	Supplementary in the case of synchronism: external position setpoint setting possibility (U460 and U461) To be used herewith:
New free blocks [794]	Free block 'Additive offset adjustment analogous to offset angle adjustment in synchronism, only external 'Offset adder with limitation' for modulo calculation of position setpoints.
Speed input [834]	Supplementary in the case of synchronism: Speed input in percent [%] (U600: 4 to 6), analog switchable with the position sources from the master setpoint source.
Visualization parameters	Supplementary in the case of synchronism: Visualization parameters n655 for position setpoint [LU], n653 for speed [%], n654 for gear factor, n466 for offset angle
Position correction [843]	Supplementary in the case of synchronism position correction: Parameter U467 for input of the setting speed in [1000LU/Min]
Start/Stop [834]	Supplementary in the case of synchronism: Status binectors start/stop (B831, B832; B833)

Synchronism mode 3 [834]	Supplementary in the case of synchronism: Mode as catch-up. Only to be used if the scaling speed master has been entered.
Master setpoint correction [845]	Technology option master setpoint correction: Function selection (U458) between master setpoint correction and master setpoint setting. Output (integrator) setting introduced. Connector output KK828 residual offset distance introduced.
Catch-up [837]	Supplementary in the case of catch-up: Deceleration delay/acceleration delay for positioning (U628.3 & 4); binector "Trigger transfer stop position" (U625.4); rounding mode introduced (U649)
Synchronism status signals [846]	Output of a synchronism status word in n450.1 (low word) and n450.2 (high word)
Homing procedure [821]	Homing procedure supplements: 1. Homing with homing switch only 2. Homing with encoder zero mark only 3. Use of a reversing switch with homing
Procedure: Referencing on the fly [822]	Homing on the fly: The assignment of the input for reference setting on the fly is by means of machine datum MD46 with identification, and is dynamically released via binector input U675.2. Behaves analogously to referencing in synchronous operation
Actual value dependent M output	Extensions of actual value dependent M (machine) output in the case of rotary axes; evaluation whether the residual travel distance to a given position is less than one revolution, and M output must be implemented.
NOTE	The additive offset angle setting and actual value dependent M (machine) output functions have not yet been released at software status V1.4.0. They shall be released with version V1.42.

Changed parameters

- ◆ U501.08 MD8 :
0 = Homing with zero and zero mark
1 = Homing with zero only
2 = Homing with zero mark only

- ◆ U501.45 MD45: Digital inputs - Function 1
0 = Without function
1 = Start OR-gated
2 = Start AND-gated
3 = Position-feedback setting on the fly
4 = External set change
5 = Measuring on the fly
6 = Collision
7 = Zero for homing
8 = Reversing cam for homing
9 = Release read-in, dependent on external program

- ◆ U627.3,4 Parameter Index 3 and 4 do not apply (not used)

- ◆ U628.3,4 Parameter Index 3 and 4 extended
Acceleration/deceleration for positioning

- ◆ n655.1...5 Position display parameter [LU] for synchronism extended

- ◆ n653.1...5 Speed display parameter [%] for synchronism extended

- ◆ n668.1...8 Status of tables extended to Tables 1-8

- ◆ U602 Synchronism mode extended from 0..2 to 0..3 (catch-up)

- ◆ U615 Table configuration extended from 1,2 to 0...4 for 8 tables

- ◆ U650.1..3 Binector for table selection extended for 8 tables.

New parameters	◆ U449	Rounding mode, catch-up
	◆ n459.1,2	Display parameters, tables of X/Y axis positions
	◆ U600.4-6	Master speed setpoint, master axis [%]
	◆ U461.1-2	Source, additive offset, slave
	◆ U607.2	Scaling speed, master
	◆ U607.2	Scaling speed, slave (alternative to MD23, here with two places after the decimal)
	◆ n654.1...2	Display of set gear factor (numerator/denominator)
	◆ n634	Display of free support points for the table, in variable table configuration mode (U615=4)
	◆ n639.1...16	Table information: Display parameters giving information as to which tables are to be found in which parameters. Beginning of table, end of table.
	◆ n466.1...2	Visualization parameters for offset angle adjustment and synchronization. Residual offset: Index 1 Current offset: Index 2
	◆ U467	Maximum correction speed in 1000LU/min, alternative to U667 maximum correction speed in LU/sampling time.

9.11.6 Software version V2.1

(delivery as of: 06.03)

Synchronism – generation of position setpoint [836]

Creation of a docking point between the basic positioner and the synchronous block so as to ensure that the drive can be operated in position control mode after the axis has been stopped via the catch-up function.

Setpoint source U886.1 for position setpoint [LU] and U886.2 for speed setpoint [%]

The function is activated via U885.

New parameters

- ◆ U885 Synchronism local ON
- ◆ U886.1...2 Setpoint source synchronism local

**Engaging/
disengaging**

The total length of the acceleration/deceleration ramp U610 must not be more than the engaging/disengaging length U611. The engaging/disengaging length U611 must be at least as long as the total length of the acceleration/deceleration ramp U610.

From MASTERDRIVES version 1.6 as an alternative to the engaging/disengaging length, a disengaging position (special configuration U475=1) can also be specified, at which the constant travel phase of the engaging action is completed (see also function diagram 834a, column 1, note <3>).

From MASTERDRIVES version 2.1, in addition to the disengaging position the acceleration and the deceleration ramp can also be specified differently (special configuration U475 = 11, see also function diagram 834b, column 1, note <4>).

If the disengaging position (DISENG) lies before the total of acceleration starting point (ENG) and deceleration ramp end (ACC-RA), the point ((ACC-RA') or (DEC-RA')) is determined as a result thereof (from MASTERDRIVES version 2.2 with special configuration U475=111, see also function diagram 834c, column 1, note <4>).

New parameters

- ◆ U474 "Src Variable Ramps" has been newly introduced
 - U474.1 Length of acceleration ramp [LU] WE:894
 - U474.2 Length of deceleration ramp [LU] WE:894

**Extended
parameters**

- ◆ U609 has been extended to two indices
 - U609.1 Source offset of engaging position [LU] WE:822
 - U609.2 Source disengaging position [LU] WE:821
- ◆ U610 has been extended to two indices
 - U610.1 Total length of acceleration/deceleration ramp [LU] WE:1
 - U610.2 Total length of acceleration/deceleration ramp [LU] WE:0
- ◆ U613 has been extended to two indices
 - U613.1 Offset of engaging position [LU] WE:0
 - U613.2 Disengaging position [LU] WE:0

New connectors

- ◆ KK0821 Disengaging position
- ◆ KK0894 Total length of acceleration ramp
- ◆ KK0895 Total length of ramp

**New binectors
(synchronism status
signals FD846)**

- ◆ B0801 In acceleration ramp (BIT20)
- ◆ B0802 In deceleration ramp (BIT21)
- ◆ B0814 Change to Src. Variable Ramps permitted (bit 22)

9.11.7 Software status V2.2

Master value correction The binector B0827 is active as long as the speed difference still to be reduced between master value 1 and master value 2 at KK0866 is not equal to 0.

New binector ♦ B0827 Speed adjustment active

9.11.8 Software status V2.3

Basic positioner Stop cam and connector inputs for the software limit switch.
See Compendium Chapter 7.2.3 and function diagrams 789a/b

Extended parameters

- ♦ U866 Control Word BP-SET
 - U866.14 SC_ON (Enable Stop Cam)
 - U866.15 SC_PLUS (Stop Cam Plus)
 - U866.16 SC_MINUS (Stop Cam Minus)
- ♦ U873 Fixed Connectors [%] for Basic Positioning:
 - U873.4 Fixed Connectors 896 Deceleration Stop Cam
- ♦ U850 Source Basic Positioning Position Setpoints
 - U850.7 BPos SWLS Plus (Software Limit Switch Plus)
 - U850.8 BPos SWLS Minus (Software Limit Switch Minus)

New connectors

- ♦ K0896 Fixed Connector Deceleration Stop Cam
- ♦ KK0897 Position Difference at Input: Delta S in LU
- ♦ KK0898 Fixed Connector Software Limit Switch Plus
- ♦ KK0899 Fixed Connector Software Limit Switch Minus

New binectors

- ♦ B0896 SC_PLUS_ACTIV (Stop Cam Plus active)
- ♦ B0897 SC_MINUS_ACTIV (Stop Cam Minus active)

9.11.9 Software status V2.4

Basic positioner Static STOP cam (see Section 7.2.3 and FD789a)

Register decoupling Two new free blocks (register displacement and register decoupling), see function diagram 842.

- New parameters**
- ◆ U700 Connector parameter
 - U700.1 Adaptation of circumference
 - U700.2 Decoupling from predecessor
 - U700.3 Setting value
 - U700.4 Register displacement offset angle
 - U700.5 Register displacement offset angle speed
 - U700.6 Decoupling
 - U700.7 Adjustment speed of printing cylinder
 - U700.8 Current angular speed of printing cylinder
 - U700.9 Current circumference
 - ◆ U701 Binector parameter
 - U701.1 Enable register displacement
 - U701.2 Set register displacement
 - U701.3 Enable register decoupling
 - ◆ U702 Function parameter - Adaptation
 - U702.1 Speed adaptation register displacement
 - U702.2 Speed adaptation register decoupling
 - U702.3 T constant adaptation register decoupling
 - ◆ U703 Function parameter - General parameters
 - U703.1 Register displacement axis cycle
 - U703.2 Circumference
 - U703.3 Normalization of circumference
 - U703.4 Distance printing position
 - ◆ U704 Function parameter - Normalization
 - U704.1 Normalization of speed
 - U704.2 Normalization of web speed
 - ◆ U705 Selection parameter
 - U705.1 Operating mode
 - U705.2 Characteristic
 - ◆ U887 BPos Mode Cam
 - 0 = Edge change
 - 1 = Static
- Extended parameters**
- ◆ n465 Visualization paramater
 - n465.2 Current adjusting speed including v_inching as a percentage referred to U607

New connectors

- ◆ KK0794 Register decoupling
- ◆ KK0795 Register displacement offset angle
- ◆ KK0796 Register displacement offset angle speed
- ◆ KK0797 Current circumference
- ◆ KK0798 Decoupling to successor
- ◆ KK0799 Adaptation of circumference
- ◆ KK0803 Current adjusting speed including v_{inching} as a percentage referred to U607

9.12 References, software products and accessories

/1/ "Motion Control for MASTERDRIVES MC and SIMATIC M7" manual including SIMATIC S7 "Motion Control Configuring Package" software on CD-ROM

- ◆ Order number German 6AT1880-0AA00-1AE0
- ◆ Order number English 6AT1880-0AA00-1BE0
- ◆ Internal Siemens order location: LZF Logistics Center Fürth

The configuring package also includes the GMC-BASIC standard software.

/2/ Motion Control HMI Package for SIMATIC S7

- ◆ Order number: 6AT1880-0AA10-1YA0

The HMI package also contains the GMC-OP-OAM standard software.

/3/

"DVA_S5" Option Package for SIMATIC S5		
Order number: 6DD1800-0SW0	German / English	◆ Communication software "PROFIBUS-DP" for • S5-95U / DP Master • S5-115 ... 155U with IM308-B/C ◆ Communication software "USS protocol" for • S5-95 / 100U with CP521Si • S5-115 ... 155U with CP524
Internal Siemens ordering location: A&D SE B1 TDL11 (Order form recipient G610B "WKF Fürth")		(3.5" diskette for S5-DOS including German/English User Manual)

/4/

Drive ES SIMATIC Software Package			
Ordering data			
Drive ES SIMATIC V5.1 Single license	6SW1700-5JC00-1AA0	CD-ROM, 1	Five standard languages
Drive ES SIMATIC V5.1 Copy license / Runtime license	6SW1700-5JC00-1AC0	Product certificate only (without SW and DOCU)	Five standard languages
Content of Drive ES SIMATIC package			
◆ "PROFIBUS-DP" communication software for S7-300 with CPUs with integrated DP interface (program libraries DRVDPS7, POSMO) S7-400 with CPUs with integrated DP interface or with CP443-5 (program library DRVDPS7, POSMO) S7-300 with CP342-5 (program library DRVDPS7C) ◆ "USS Protocol" communication software for S7-200 with CPU214 / CPU215 / CPU 216 (driver program DRVUSS2 for STEP7 Micro programming tool) S7-300 with CP340/341 and S7-400 with CP441 (program library DRVUSSS7) ◆ STEP7 Slave Object Manager for convenient configuring of drives and non-cyclic PROFIBUS-DP communication with drives, support for conversion of DVA_S7 to Drive ES projects (V5.1 and later only) ◆ SETUP program for installing software in the STEP7 environment			

/5/ 1 axis demonstration pack, order no. 6SX7000-0AF00 contents:

- ◆ 1FK6 synchronous motor with resolver
- ◆ 1 MASTERDRIVES MC Compact PLUS converter
- ◆ Brake resistor, RI suppression filter
- ◆ Operator panel
- ◆ Ready to connect the AC cable

Internal Siemens ordering location: A&D SE B8.4
("WKF Fürth", Tel. 4894)

/6/ 2-axis demonstration pack, order no. 6SX7000-0AF10 contents:

- ◆ 1FT6 synchronous motor with optical sin/cos encoder
- ◆ 1FK6 synchronous motor with resolver
- ◆ One toothed disc with position index per motor
- ◆ LED beam for checking the synchronization
- ◆ MASTERDRIVES MC Compact PLUS converters and inverters
- ◆ Brake resistor, RI suppression filter
- ◆ Operator panel
- ◆ Ready to connect the AC cable

Ordering location: same as 1-axis demonstration pack

10 Control Word and Status Word

10.1 Description of the control word bits

The operating statuses can be read in visualization parameter r001:
e.g. READY TO POWER-UP: r001 = 009

The function sequences are described in the sequence in which they are actually realized.

An overview of the control word can be found in function diagrams 180 and 190.

Bit 0: ON/OFF 1 command (↑ "ON") / (L "OFF1")

Condition	Positive edge change from L to H (L → H) in the READY TO POWER-UP condition (009).
Result	◆ PRECHARGING (010) The main contactor (option), if present, is closed. Precharging is carried out. After precharging, the bypass contactor, if present, is closed. ◆ READY (011) If the drive was last powered-down with "OFF2", the next condition is only selected after the de-energization time (P603) has expired since the last shutdown. ◆ RUN (014).
Condition	LOW signal
Result	◆ OFF1 (015), if the drive is in a status where the inverter is enabled. • For P290 = 0 and slave drive, the system waits until the higher-level open-loop/closed-loop control shuts down the drive. • For P290 = 0 and master drive as well as for P290 = 1 (v/f characteristic), the setpoint at the ramp-function generator input is inhibited (setpoint = 0), so that the drive decelerates along the parameterized down ramp (P464) to the OFF shutdown frequency (P800). After the OFF delay time (P801) has expired, the inverter pulses are inhibited, and the main contactor (option/bypass contactor), if available, are opened. If the OFF1 command is withdrawn again when the drive is ramping-down, (e.g. as the result of an ON command), ramp-down is interrupted, and the drive goes back into the RUN (014) condition. ◆ For PRECHARGING (010), READY (011), RESTART-ON-THE-FLY (013)¹ or MOT-ID-STANDSTILL (018), the inverter pulses are inhibited, and the main contactor (option)/bypass contactor, if available, is opened. ◆ POWER-UP INHIBIT (008) ◆ READY-TO-POWER-UP (009), if "OFF2" or "OFF3" are not present.

¹ The "Restart-on-the-fly" function has not been provided.

Bit 1: OFF2 command (L "OFF2") electrical**Condition** LOW signal**Result**

- ◆ The inverter pulses are inhibited, and the main contactor (option)/bypass contactor, if available, are opened.
- ◆ POWER-ON INHIBIT (008), until the command is removed.

NOTE

The **OFF2** command is simultaneously connected from three sources (P555, P556 and P557)!

Bit 2: OFF3 command (L "OFF3") (fast stop)**Condition**

LOW signal

Result

- ◆ This command has the following results:
 - When $P290 = 0$ (closed-loop current control), the drive is braked at the current limit by means of torque control (see function diagram 370). The sign before the braking torque is always opposite to that of the speed actual value. If the speed actual value reaches the OFF value $P800$ (see function diagram 480), the gating pulse is disabled. If OFF3 is used, the OFF time must be set to $P801 = 0.0$.
If brake control is used, $P801$ must be $> P617 + P607$. Brake control (function diagram 470) should not be used together with AUS3. With brake control the inverter must not be disabled until the brake is fully applied (i.e. after brake closure time $P607$ has exceeded and, if necessary, delay brake threshold $P617 \rightarrow P891 > 0$). During this time, the drive would necessarily hum on AUS3 because every sign inversion of the speed actual value (noise to n_{act} , average value of $n_{act} = 0$ if brake is applied) would also reverse the torque direction. The brake control should be used with the AUS1 command.
In the case of drives with a low moment of inertia in comparison to the motor moment of inertia, it may be necessary to increase the OFF value $P800$ by up to 1 to 5 %. If this is not sufficient in order to avoid fluctuation of the torque-forming current setpoint when $n \approx 0$, a speed-controlled shutdown with OFF1 must be carried out. The torque limits ($K0172$, $K0173$) have no effect during OFF3. The torque can only be limited by means of $P128$ (maximum current).
When $P290 = 1$ (v/f characteristic), the setpoint at the ramp-function-generator input is blocked so that the drive is powered down on the parameterized deceleration ramp ($P464$) to the OFF frequency (800).
After the OFF waiting time ($P801$) has expired, the inverter pulses are inhibited and the main/bypass contactor, if present, is opened. If the OFF3 command is withdrawn during deceleration, deceleration is nevertheless continued.

- ◆ For PRE-CHARGING (010), READY (011), RESTART-ON-THE-FLY (013)¹ or MOT-ID STANDSTILL (018), the inverter pulses are inhibited, and the main/bypass contactor, if used, is opened.
- ◆ If the drive operates as slave drive, when an OFF3 command is issued, it automatically switches-over to the master drive.
- ◆ POWER-ON inhibit (008), until the command is withdrawn.

NOTE

The **OFF3** command is simultaneously effective from three sources (P558, P559 and P560)!

Priority of the **OFF** commands: **OFF2 > OFF3 > OFF1**

Bit 3: Inverter enable command (H "inverter enable")/(L "inverter inhibit")

Condition	HIGH signal, READY (011) and the de-energization time (P603) has expired since the last time that the drive was shutdown.
Result	◆ RUN (014) The inverter pulses are enabled and the setpoint is approached via the ramp-function generator.
Condition	LOW signal
Result	◆ For RESTART-ON-THE-FLY (013)¹, RUN (014): The drive changes over into the READY (011), condition, and the inverter pulses are inhibited. ◆ If OFF1 is active (015), the inverter pulses are inhibited, the main/bypass contactor, if used, is opened, and the drive goes into the POWER-ON INHIBIT (008) condition. ◆ If OFF3 is active (016 / fast stop), the inverter inhibit command is ignored, fast stop is continued and, after shutdown (P800, P801), the inverter pulses are inhibited.

Bit 4: Ramp-function generator inhibit command (L "RFG inhibit")

Condition	LOW signal in the RUN (014) condition.
Result	◆ The ramp-function generator output is set to setpoint = 0.

Bit 5: Ramp-function generator hold command (L "RFG hold")

Condition	LOW signal in the RUN (014) condition.
Result	◆ The actual setpoint is "frozen at the ramp-function generator output".

Bit 6: Setpoint enable command (H "setpoint enable")

Condition	HIGH signal and the de-energization time have expired (P602).
Result	◆ The setpoint at the ramp-function generator input is enabled.

¹ The "Restart-on-the-fly" function has not been provided.

Bit 7: Acknowledge command (↑ "Acknowledge")

Condition	Rising (positive) edge change from L to H (L → H) in the FAULT condition (007).
Result	◆ All of the current faults are deleted after they have been previously transferred into the diagnostics memory. ◆ POWER-ON INHIBIT (008), if no actual faults are present. ◆ FAULT (007), if there are additional faults which cannot be acknowledged.
NOTE	The Acknowledge command is simultaneously effective from the three sources (P565, P566 and P567) and always from the PMU!

Bit 8: Inch 1.3 Bit 0, ON command (↑ "Inch 1.3 ON") / (L "Inch 1.3 OFF")

Condition	Positive (rising) edge change from L to H (L → H) in the READY TO POWER-UP (009) condition.
Result	◆ In the setpoint channel, an ON command (see control-word bit 0) is automatically executed and, with Bit 9 = 0, inching frequency 1 (P448) is enabled or, when bit 8 and bit 9 are activated at the same time, inching frequency 3 (P450) is enabled. The ON/OFF1 command (bit 0) is ignored for active inching operation! The system must wait until the de-energization time (P603) has expired. (See also function diagram 310)
Condition	LOW signal
Result	◆ An OFF1 command is automatically executed (refer to control word bit 0).

Bit 9: Inch 2.3 Bit 1, ON command (↑ "Inch 2.3 ON") / (L "Inch 2.3 OFF")

Condition	Rising (positive) edge change from L to H (L → H) in the READY TO POWER-UP (009) condition.
Result	◆ In the setpoint channel, an ON command (see control-word bit 0) is automatically executed and, with Bit 8 = 0, inching frequency 2 (P449) is enabled or, when bit 8 and bit 9 are activated at the same time, inching frequency 3 (P450) is enabled. The ON/OFF1 command (bit 0) is ignored if inching is active. The system must wait until the de-energization time (P603) has expired. (See also function diagram 310)
Condition	LOW signal
Result	◆ An OFF1 command is automatically executed (refer to control word bit 0).

Refer to the function diagram "Setpoint processing (Part 1)" (310)

Bit 10: Control from the PLC command (H "control from the PLC")

Condition	HIGH signal; the process data PZD (control word, setpoints) are only evaluated if the command has been accepted; this data is sent via the SST1/2 interface of the CUPM, the CB/TB interface (option) and the SCB2 interface (option). In the case of Compact Plus, the SST1/2 interface is on the basic board; the TB and SCB2 interfaces do not exist.
Result	◆ If several interfaces are used, only the process data of the interfaces are evaluated, which send an H signal. ◆ For an L signal, the last values are received in the appropriate dual port RAM of the interface.
NOTE	An H signal appears in the visualization parameter r550 "control word 1", if one of the interfaces sends an H signal!

Bit 11: Clockwise rotating field command (H "clockwise rotating field")

Condition	HIGH signal
Result	◆ The setpoint is influenced in conjunction with bit 12 "counter-clockwise rotating field".

Refer to the function diagram "Setpoint processing (Part 1)" (310)

Bit 12: Counter-clockwise rotating field command (H "counter-clockwise rotating field")

Condition	HIGH signal
Result	◆ The setpoint is influenced in conjunction with bit 11 "clockwise-rotating field".

Refer to the function diagram "Setpoint processing (Part 1)" (310)

NOTE	The counter-clockwise rotating field and the clockwise rotating field command have no influence on supplementary setpoint 2, which is added after the ramp-function generator (RFG)!

Bit 13: Command to raise the motorized potentiometer (H "raise motorized potentiometer")

Condition	HIGH signal
Result	◆ The motorized potentiometer in the setpoint channel is driven in conjunction with bit 14 "motorized potentiometer, lower".

Refer to the function diagram "Motorized potentiometer" (300)

Bit 14: Command to lower the motorized potentiometer (H "lower motorized potentiometer")

Condition	HIGH signal
Result	◆ The motorized potentiometer in the setpoint channel is driven in conjunction with bit 13 "raisemotorized potentiometer".

Refer to the function diagram "Motorized potentiometer" (300)

Bit 15: Command external fault 1 (L "External fault 1")**Condition** LOW signal**Result**

- ◆ FAULT (007) and fault message (F035).
The inverter pulses are inhibited, the main contactor/bypass contactor, if used, is opened.

Refer to the section "Fault- and alarm messages"

Bit 16: Function data set FDS bit 0 command**Result**

- ◆ In conjunction with bit 17 "FDS BIT 1" one of the four possible function data sets is energized.

Bit 17: Function data set FDS bit 1 command**Result**

- ◆ In conjunction with bit 16 "FDS BIT 0" one of the four possible function data sets is energized.

Bit 18, 19: Reserve**Bit 20: Fixed setpoint FSW bit 0 command****Result**

- ◆ In conjunction with bit 21 "FSW BIT 1", one of the four possible fixed setpoints is energized to input as percentage fixed setpoints, referred to the reference frequency P352 or reference speed P353.

Refer to the function diagram "Fixed setpoints" (290), also refer to FSW bit 2 and bit 3, parameter P417, P418

Bit 21: Fixed setpoint FSW bit 1 command**Result**

- ◆ In conjunction with bit 20 "FSW BIT 0" one of the four possible fixed setpoints is energized for input as percentage fixed setpoints, referred to the reference frequency P352 or the reference speed P353.

Refer to the function diagram "Fixed setpoints" (290), also refer to FSW bit 2 and bit 3, parameters P417, P418

Bit 22: Reserve

Bit 23: Reserve**Bit 24: Enable-droop command (H "Droop enable")****Condition** HIGH signal

Result ♦ This command enables the droop function if P290 = 0 has been assigned, parameter P246 is $\neq 0$ and the inverter pulses of the converter have been enabled.
The speed-controller output negatively linked back to the speed setpoint can be set via parameters P245 (droop) and P246 (droop gain).

Refer to function diagram "Speed controller" 360

Bit 25: Controller enable command (H "controller enable")**Condition** HIGH signal and the drive converter inverter pulses are enabled.

Result ♦ The output of the speed controller is enabled when P290 = 0 (current control).

Refer to function diagram 360

Bit 26: Command, external fault 2 (L "External fault 2")**Condition** LOW signal; it is only activated from the READY (011) condition onwards and after an additional time delay of 200 ms.

Result ♦ FAULT (007) and fault message (F036).
The inverter pulses are inhibited, the main contactor, if available, is opened.

Bit 27: Slave/master drive command (H "Slave drive")/(L "Master drive")**Condition** HIGH signal, P290 = 0 and enabling of the inverter pulses of the converter.

Result ♦ Slave drive: The closed-loop control acts as closed-loop torque control (M closed-loop control).

Condition LOW signal, P290 = 0 and enabling of the inverter pulses of the converter.

Result ♦ Master drive: The closed-loop control function acts as a closed-loop speed control function (n-control).

Refer to the function diagrams 360, 370

Bit 28: Command, external alarm 1 (L "External alarm 1")**Condition** LOW signal**Result** ♦ The operating status is maintained. An alarm message is issued (A015).**Bit 29: Command, external alarm 2 (L "External alarm 2")** ♦**Condition** LOW signal**Result** ♦ The operating status is maintained. An alarm message is issued (A016).**Bit 30: Select, BICO data sets (H "data set 2") / (L "data set 1")****Condition** HIGH signal**Result** ♦ The parameter settings of data set 2 for all binector and connector commands and signals, are activated.**Condition** LOW signal**Result** ♦ The parameter settings of data set 1 for all binector and connector commands and signals, are activated.**Bit 31: Main contactor checkback signal command (H "main contactor checkback signal")****Condition** HIGH signal, corresponding to the wiring and parameterization of the main contactor (option).**Result** ♦ Checkback signal, "main contactor energized" (closed).

10.2 Description of the status word bits

An overview of the status word can be found in function diagrams 200 and 210.

Bit 0: Message, "Ready to power-up" (H)

HIGH signal	POWER-ON INHIBIT (008) or READY TO POWER-UP (009) status
Significance	◆ The power supply, the open- and closed-loop control are operational. ◆ The inverter pulses are inhibited. ◆ If an external power supply and a main contactor (option)/bypass contactor exist, it is possible that the DC link is still dead in this converter status!

Bit 1: Message, "Ready" (H)

HIGH signal	PRE-CHARGING (010) or READY (011) status
Significance	◆ The power supply, the open-loop and the closed-loop control are operational. ◆ The unit is powered-up. ◆ Pre-charging has been completed. ◆ The DC link has reached the rated voltage. ◆ The inverter pulses are still inhibited.

Bit 2: Message, "Run" (H)

HIGH signal	RESTART-ON-THE-FLY (013) ¹ , RUN (014), OFF1 (015) or OFF3 (016)
Significance	◆ The unit is functioning. ◆ The inverter pulses are enabled. ◆ The output terminals are live.

Bit 3: Message "Fault" (H)

HIGH signal	Fault (007) status
Significance	◆ A fault has occurred. <i>Output at the terminal strip with L signal.</i>

¹ The "Restart-on-the-fly" function has not been provided.

Bit 4: Message "OFF2" (L)

LOW signal	OFF2 command available
Significance	◆ The OFF2 command was output (control word bit 1).

Bit 5: Message "OFF3" (L)

LOW signal	OFF3 (016) status, and/or OFF3 command available
Significance	◆ The OFF3 command was output (control word bit 2).

Bit 6: Message "Power-on inhibit" (H)

HIGH signal	POWER-ON INHIBIT (008) status
Significance	◆ The power supply, open-loop and closed-loop control are operational. ◆ If an external power supply and a main contactor (option)/bypass contactor are available, it is possible to bring the DC link voltage in this drive converter status into a no-voltage condition! ◆ The message is available as long as an OFF2 command is present via control word bit 1 or an OFF3 command is available via control word bit 2 after the setpoint has been ramped-down, or an ON command is available via control word bit 0 (edge evaluation).

Output at the terminal strip with L signal.

Bit 7: Message, "Alarm" (H)

HIGH signal	Alarm (Axxx)
Significance	◆ An alarm has been issued. ◆ The signal is present until the cause has been resolved.

Output at the terminal strip with L signal.

Bit 8: Message "Setpoint/actual-value deviation" (L)

LOW signal	Alarm, "Setpoint-actual value deviation" (A034)
Significance	◆ A deviation of the actual value compared to the comparison setpoint has occurred which is larger than P792 (set/act dev) and lasts longer than P794 (set/act dev. time). (See also function diagram 480) ◆ The bit is again set as H signal, if the deviation is less than parameter value P792.

Bit 9: Message "PZD control requested" (H)

HIGH signal	Still present.

Bit 10: Message "Comparison value reached" (H)

HIGH signal	The parameterized comparison value has been reached.
Significance	◆ The actual value is greater or equal to the parameterized comparison value (P796). ◆ The bit is re-set to the L signal as soon as the actual value falls below the comparison value (P796) minus the parameterized comparison-value hysteresis (P797 in % in relation to the comparison value (P796)). (See also function diagram 480).

Bit 11: Message "Fault, undervoltage" (H)

HIGH signal	"Undervoltage in the DC link"
Significance	◆ The DC link voltage has fallen below the permissible limit value. From drive state (°011) there is an additional fault message (F008) "DC link undervoltage". Refer to the Section "Fault- and alarm messages" <i>Output at the terminal strip with L signal.</i>

Bit 12: Message "Main contactor energized" (H)

HIGH signal	The main contactor/bypass contactor (option) is energized.
Significance	◆ The main contactor/bypass contactor (option) can be driven with the appropriate wiring and parameterization.

Bit 13: Message "RFG active" (H)

HIGH signal	Ramp-function generator active
Significance	◆ The ramp-function generator output (KK0073) is not equal to the ramp-function generator input (KK0072).

Bit 14: Message, "Clockwise rotating field" (H)/"Counter-clockwise rotating field" (L)

HIGH signal	Clockwise rotating field
Significance	◆ The speed setpoint for closed-loop control (speed setpoint r472 / KK0075) is greater or equal to 0.
LOW signal	Counter-clockwise rotating field
Significance	◆ The speed setpoint for closed-loop control (speed setpoint, r472 / KK0075) is smaller than 0.

Bit 15: Reserve

Bit 16: Message "Restart-on-the-fly active" (H) ¹

HIGH signal The restart-on-the-fly function is active, or the de-energization time (P602) is running.

Significance ♦ The excitation time (magnetization time) is active.

Bit 17: Reserve**Bit 18: Message "Overspeed" (L)**

LOW signal Alarm "Overspeed" (A033)

Significance ♦ The speed actual value is either greater than the maximum speed for the clockwise-rotating field (P452) or is smaller than the maximum speed for the counter-clockwise rotating field (P453).
 ♦ The bit is set to the H signal again as soon as the speed actual value is smaller than or equal to the corresponding maximum speed. (See also function diagram 480).

Bit 19: Message "External fault 1" (H)

HIGH signal "External fault 1"

Significance ♦ A "External fault 1" is present in control word, bit 15.

Output at the terminal strip with L signal.

Bit 20: Message "External fault 2" (H)

HIGH signal "External fault 2"

Significance ♦ A "External fault 2" is present in control word bit 26.

Output at the terminal strip with L signal.

Bit 21: Message "External alarm" (H)

HIGH signal "External alarm"

Significance ♦ An "external alarm 1" is present in control word bit 28, or, "external alarm 2" in control word bit 29.

Output at the terminal strip with L signal.

¹ The "Restart-on-the-fly" function has not been provided.

Bit 22: Message "Alarm i²t drive converter" (H)**HIGH signal**Alarm "i²t alarm, inverter" (A025)**Significance**

- ◆ If the instantaneous load status is maintained, then the drive converter will be thermally overloaded.
(See also function diagram 480)

*Output at the terminal strip with L signal.***Bit 23: Message "Fault, converter overtemperature" (H)****HIGH signal**

"Inverter temperature too high" fault (F023)

Significance

- ◆ The limiting inverter temperature has been exceeded.

*Output at the terminal strip with L signal.***Bit 24: Message "Alarm, converter overtemperature" (H)****HIGH signal**

Alarm, "inverter temperature too high" (A022)

Significance

- ◆ The inverter temperature threshold to release an alarm has been exceeded.

*Output at the terminal strip with L signal.***Bit 25: Message "Alarm, motor overtemperature" (H)****HIGH signal**

Alarm "Motor overtemperature"

Significance

- ◆ This is an overtemperature alarm triggered by the KTY (P380 > 0).
- ◆ The precondition for the alarm is satisfied by a measurement with the KTY84 sensor (r009 / K0245).
- ◆ Parameters involved in the calculation:
P380 (mot. temp. alarm).

*Output at the terminal strip with L signal.***Bit 26: Message "Fault, motor overtemperature" (H)****HIGH signal**

Fault, "Motor overtemperature"

Significance

- ◆ This is an overtemperature fault detected by the KTY (P381 > 1).

*Output at the terminal strip with L signal.***Bit 27: Reserve**

Bit 28: Message, "Fault, motor stalled/locked" (H)**HIGH signal**

Fault, "Motor stalled or blocked" (F015)

Significance

- ◆ The drive has either stalled or is locked.

*Output at the terminal strip with L signal.***Bit 29: Message "Bypass contactor energized" (H)****HIGH signal**

The bypass contactor is energized.

Significance

- ◆ With appropriate wiring and parameterization, an external bypass contactor (option) can be energized (only for DC devices).

Bit 30: Reserve**Bit 31: Message "Pre-charging active" (H)****HIGH signal**

PRE-CHARGING (010) condition

Significance

- ◆ Pre-charging is realized after an ON command.

11 Engineering Information

General

Servo drives are mostly cycle-type drives, i.e. drives which perform particular sequences of movement within a fixed cycle of motion. These movements can be linear or rotational. In addition, the motion sequence usually involves approaching a pre-defined position and all movements must be carried out in the shortest time possible. As a consequence, drives have to meet specific requirements. They must be

- ◆ dynamic, i.e. move to the desired position in an optimum time and without overshoot
- ◆ overload-capable, i.e. have a high acceleration reserve
- ◆ and must have a large control range, i.e. high resolution for exact positioning.

The planning notes below refer to servo drives with 1FK6/1FT6 synchronous servomotors or with 1PA6 induction servomotors. The type of motor which is selected, either synchronous or induction, depends on the requirements of the drive and on the required drive power. Synchronous servomotors are preferred where a small unit volume, low rotor inertia, high overload capability and thus fast response levels are important. Induction servomotors are of simpler design and are therefore very sturdy. They only require a pulse encoder instead of an encoder or a resolver. Induction servomotors are offered with ratings up to 160 kW.

Which components are selected from the Motion Control system depends on the type of drive configuration used. The drives can be operated individually, for example, as single-axis drives or together as multi-axis drives. For connecting the drives to a PLC, via PROFIBUS for example, supplementary boards may be necessary. Decentralized provision of technology functions within the Motion Control system is possible with special software or the functions can be provided centrally by means of a PLC. Drives can be coupled, e.g. for angular synchronism, via SIMOLINK.

A typical planning sequence

The planning of a cycle-type drive, i.e. the selection of the motor, converter/inverter and perhaps the rectifier unit, is carried out in the following sequence:

1. Clarification of the type of drive, technical data and other border conditions
2. Specification of the travel curve
3. Calculation of the maximum speed under load and the maximum load torque, selection of the gear
4. Selection of the motor
5. Selection of the converter or inverter
6. Selection of the rectifier unit if multi-axis drives are used
7. Selection of the braking unit and the braking resistor
8. Selection of other components.

NOTE

Steps 2 to 8 of the above sequence can be conveniently carried out with the aid of the "PFAD" planning program. In particular, this method simplifies the optimization of the drive, which would otherwise entail a great amount of calculation work.

11.1 Clarification of the type of drive, technical data and other border conditions

The procedure for calculating the load torque depends on the type of drive. It may be a travel drive, a lifting drive, a turntable drive or a spindle drive. In the case of linear motion, for example, power can be transmitted via a toothed belt, a gear rack, a spindle or via friction. Normally, a further set of gears is also needed for adapting the motor speed and the motor torque to the load conditions.

For this calculation, the necessary technical data must be available, such as the moving masses, diameter of the drive wheel/pinion or the diameter and pitch of the spindle, details of the frictional resistance, mechanical efficiency, maximum speed, maximum acceleration and maximum deceleration, travel distances and times and the accuracy levels for positioning. If the drive consists of several motors with the same load distribution which are each operated as individual drives on a converter/inverter, the conditions for one motor have to be taken into consideration in each case when dimensioning the drive (moving masses, load-side moments of inertia, additional forces/torques divided by the number of motors).

11.2 Specification of the travel curve

The travel curve - namely the v, t diagram when a linear drive is being used - is determined from the information relating to travel distances, maximum speed, acceleration, deceleration and the cycle time. If multi-axis drives are used, the interdependence of the individual travel curves must be taken into account. The travel curve is needed for deciding on the thermal rating of the motor and the type of braking resistors. It should therefore represent a "worst-case scenario" for the particular type of motor and resistors chosen.

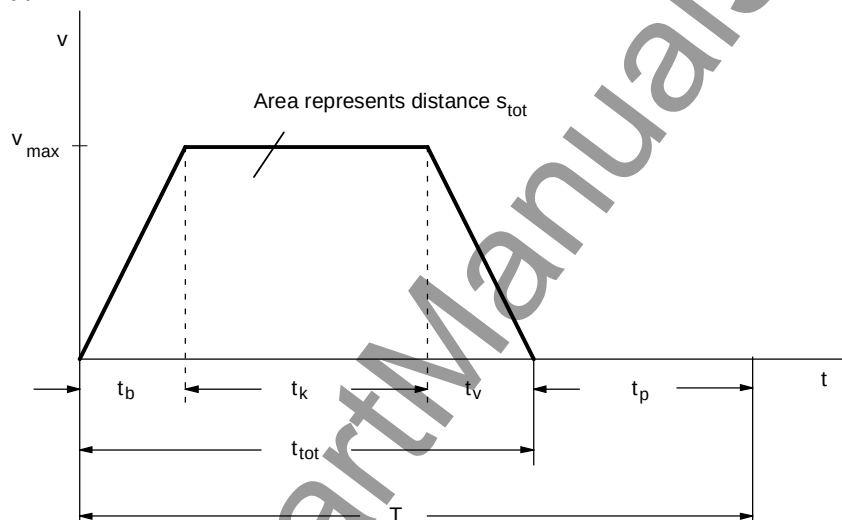


Fig. 11-1 Example of a simple travel curve

◆ Acceleration (b) time [s] $t_b = \frac{v_{\max}}{a_b}$

◆ Deceleration (v) time [s] $t_v = \frac{v_{\max}}{a_v}$

$v_{\max}$ Maximum speed [m/s]

$a_{b,v}$ Acceleration, deceleration [m/s^2]

◆ Time for constant (k) travel [s] $t_k = \frac{s_{\text{tot}} - v_{\max} \cdot \frac{t_b}{2} - v_{\max} \cdot \frac{t_v}{2}}{v_{\max}}$

s_{tot} Travel distance [m]

◆ Travel time [s] $t_{\text{tot}} = t_b + t_k + t_v$

NOTE

On rotating drives (turning mechanisms), the values $\omega_{\max}$, $\alpha_{b,v}$, ϕ_{tot} have to be applied instead of $v_{\max}$, $a_{b,v}$, s_{tot} .

11.3 Calculation of the maximum speed under load and the maximum load torque, selection of the gear

Information on the mechanical factors involved is used to calculate the maximum speed under load and the maximum load torque. The following specifies the calculation formulae for simple drive tasks:

Horizontal travel drive

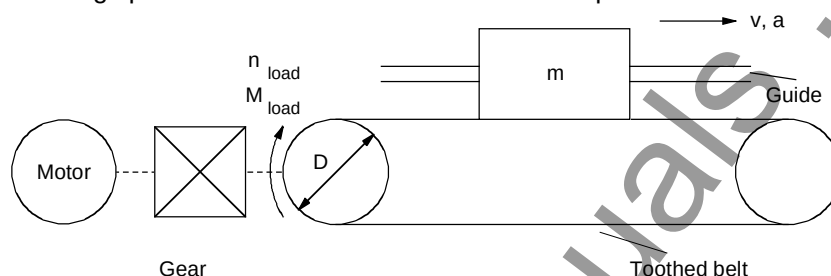


Fig. 11-2 Horizontal travel drive

- ◆ Speed under load [rpm]

$$n_{\text{load}} = \frac{v \cdot 60}{p \cdot D}$$

v Travel speed [m/s]
 D Diameter of load wheel/pinion [m]

- ◆ Travel resistance / frictional force [N]

$$F_W = m \cdot g \cdot w_F$$

w_F specific travel resistance

- ◆ Resistance/frictional torque [Nm]

$$M_W = F_W \cdot \frac{D}{2}$$

- ◆ Angular acceleration and deceleration at the load wheel/pinion [s⁻²]

$$\alpha_{b,v \text{ load}} = a_{b,v} \cdot \frac{2}{D}$$

$a_{b,v}$ Acceleration, deceleration [m/s²]

- ◆ Load inertia [kgm²]

$$J_{\text{load}} = m \cdot \left(\frac{D}{2}\right)^2$$

- ◆ Acceleration torque and deceleration torque for the load [Nm]

$$M_{b,v \text{ load}} = J_{\text{load}} \cdot \alpha_{b,v \text{ load}}$$

- ◆ Load torque at the drive wheel/pinion [Nm]

$$M_{\text{load}} = (M_{b,v \text{ load}} + M_W) \cdot \frac{1}{\eta_{\text{mech}} \cdot \text{Sign}(M_{b,v \text{ load}} + M_W)}$$

η_{mech} Mech. efficiency of the travel drive

$M_{b,v \text{ load}}$ has to be applied with the correct sign (accelerate = +, decelerate = -)

If the deceleration is equal to the acceleration, the load torque is at a maximum during the acceleration phase.

Lifting drive

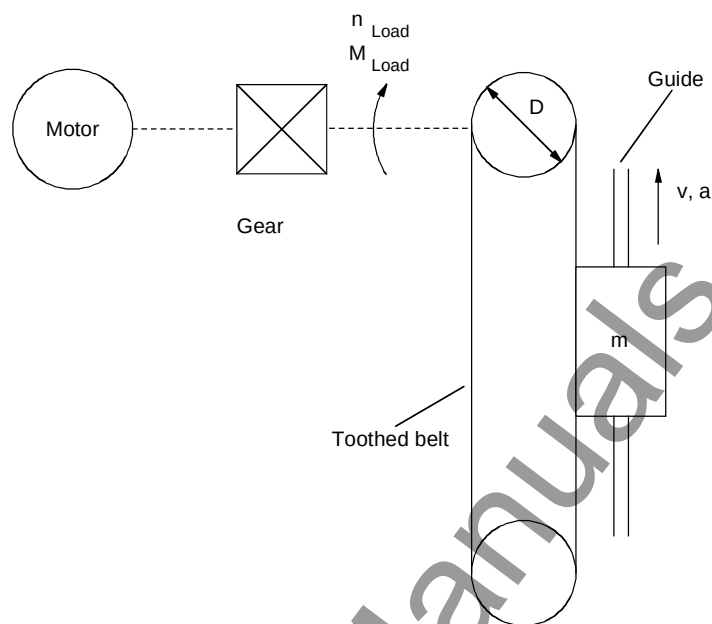


Fig. 11-3 Lifting drive

- ◆ $n_{\text{load}}, \alpha_{\text{b,v load}}, J_{\text{load}}, M_{\text{b,v load}}$ See "Horizontal travel drive"

- ◆ Lifting force [N]

$$F_H = m \cdot g$$

- ◆ Lifting torque [Nm]

$$M_H = F_H \cdot \frac{D}{2}$$

- ◆ Lifting torque at the drive wheel/pinion [Nm]

$$M_{\text{load up}} = (M_{\text{b,v load}} + M_H) \cdot \frac{1}{\frac{\text{Sign}(M_{\text{b,v load}} + M_H)}{\eta_{\text{mech}}}}$$

$$M_{\text{load down}} = (M_{\text{b,v load}} + M_H) \cdot \frac{\text{Sign}(M_{\text{b,v load}} + M_H)}{\eta_{\text{mech}}}$$

η_{mech} Mech. efficiency of the lifting drive

$M_{\text{b,v load}}$ has to be applied with the correct sign (acceleration up, deceleration down = + , deceleration up, acceleration down = -)

If the deceleration is equal to the acceleration, the load torque is at a maximum during the upwards acceleration phase.

Turning drive

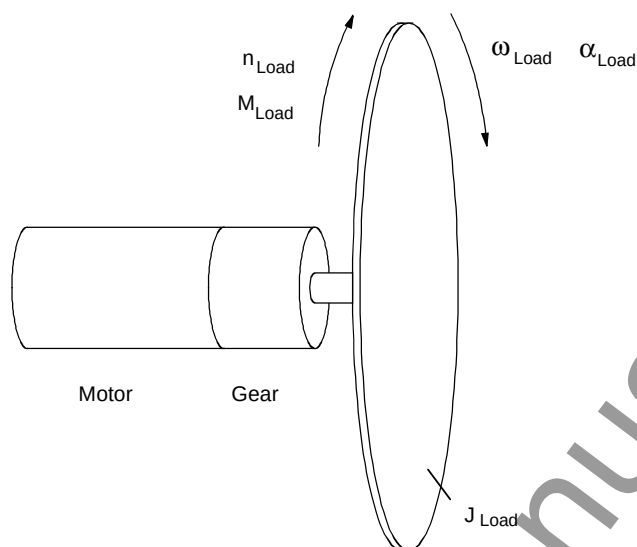


Fig. 11-4 Turning drive

- ◆ Speed under load [rpm]

$$n_{load} = \frac{\omega_{load} \cdot 60}{2 \cdot \pi}$$

ω_{load} Angular speed of the load [s^{-1}]

$\alpha_{b,v load}$ Angular acceleration/deceleration of the load [s^{-2}]

- ◆ Load torque [Nm]

$$M_{load} = J_{load} \cdot \alpha_{b,v load} \cdot \frac{1}{\eta_{mech} \cdot \text{sign}(\alpha_{b,v load})}$$

η_{mech} Mech. efficiency of the turning drive

$\alpha_{b,v load}$ has to be applied with the correct sign (accelerate = +, decelerate = -)

If the deceleration is equal to the acceleration, the load torque is at a maximum during the acceleration phase.

Horizontal spindle drive

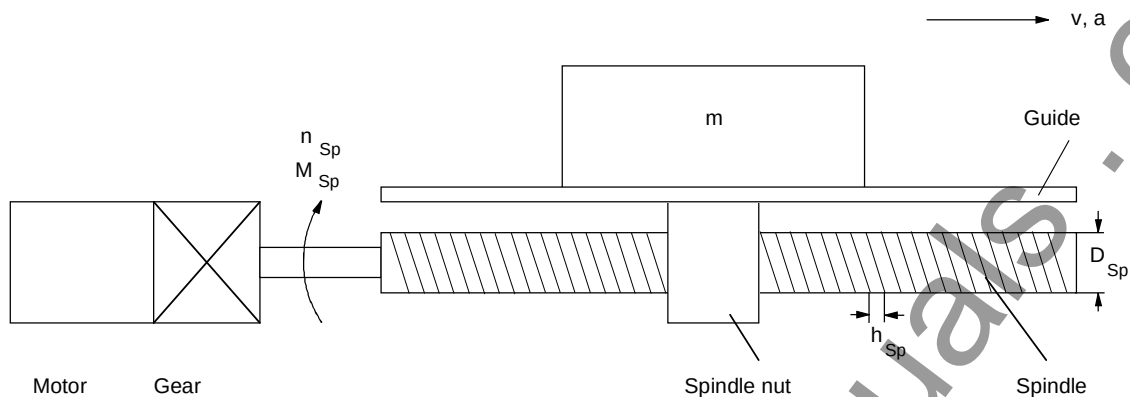


Fig. 11-5 Horizontal spindle drive

- ◆ Spindle speed [rpm]

$$n_{Sp} = \frac{v \cdot 60}{h_{Sp}}$$

v Speed [m/s]
 h_{Sp} Spindle pitch [m]

- ◆ Angle of pitch of spindle [wheel]

$$\alpha_{SW} = \arctan\left(\frac{h_{Sp}}{\pi \cdot D_{Sp}}\right)$$

D_{Sp} Spindle diameter [m]

- ◆ Frictional angle of spindle [wheel]

$$\rho = \arctan\left(\frac{\tan(\alpha_{SW})}{\eta_{Sp}}\right) - \alpha_{SW}$$

η_{Sp} Spindle efficiency

- ◆ Angular acceleration and deceleration of the spindle [s⁻²]

$$\alpha_{b,v Sp} = a_{b,v} \cdot \frac{2 \cdot \pi}{h_{Sp}}$$

- ◆ Frictional force of guide [N]

$$F_W = m \cdot g \cdot w_F$$

w_F Specific travel resistance

- ◆ Acceleration force [N]

$$F_{b,v} = m \cdot a_{b,v}$$

- ◆ Acceleration and deceleration torque for spindle [Nm]

$$M_{b,v \text{ Sp}} = J_{\text{Sp}} \cdot \alpha_{b,v \text{ Sp}}$$

$$J_{\text{Sp}} \quad \text{Inertia of spindle [kgm}^2\text{]}$$

- ◆ Load torque at the spindle [Nm]

$$M_{\text{Sp}} = M_{b,v \text{ Sp}} + (F_{b,v} + F_W) \cdot \tan(\alpha_{\text{SW}} + \rho \cdot \text{sign}(F_{b,v} + F_W)) \cdot \frac{D_{\text{Sp}}}{2}$$

$M_{b,v \text{ Sp}}, F_{b,v}$ has to be applied with the correct sign
(accelerate = +, decelerate = -)

If the deceleration is equal to the acceleration, the load torque is at a maximum during the acceleration phase.

Vertical spindle drive

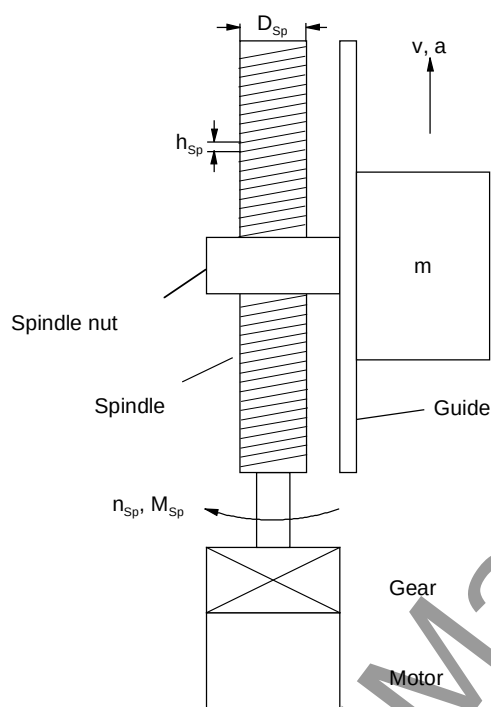


Fig. 11-6 Vertical spindle drive

◆ n_{Sp} , α_{SW} , ρ , $\alpha_{b,v Sp}$ see "Horizontal spindle drive"

◆ $F_{b,v}$, $M_{b,v Sp}$ see "Horizontal spindle drive"

◆ Lifting force [N]

$$F_H = m \cdot g$$

◆ Load torque at the spindle [Nm]:

$$M_{Sp up} = M_{b,v Sp} + (F_{b,v} + F_H) \cdot \tan(\alpha_{SW} + \rho \cdot \text{sign}(F_{b,v} + F_H)) \cdot \frac{D_{Sp}}{2}$$

$$M_{Sp down} = M_{b,v Sp} + (F_{b,v} + F_H) \cdot \tan(\alpha_{SW} - \rho \cdot \text{sign}(F_{b,v} + F_H)) \cdot \frac{D_{Sp}}{2}$$

$M_{b,v Sp}$, $F_{b,v}$ has to be applied with the correct sign (acceleration up, deceleration down = +, deceleration up, acceleration down = -)

If the deceleration is equal to the acceleration, the load torque is at a maximum during the upwards acceleration phase.

For selecting the gear, there are various other variables in addition to the maximum speed under load and the maximum load torque, e.g. size, efficiency, torsional play, torsional strength, moment of inertia, noise. Planetary gears are especially suitable for positioning tasks due to their low torsional play and high torsional stiffness. These gears also have a high power density, are highly efficient and produce a low amount of noise. When the gear transmission ratio is being selected, it should be borne in mind that higher motor speeds generally entail smaller motors. This must be checked, however, in each individual case. A higher gear transmission ratio has a favourable effect on positioning accuracy in relation to the encoder resolution. The positioning accuracy is calculated as follows from the components gears, encoder and mechanical system:

$$\Delta s_{\text{gear}} = \frac{D \cdot \pi}{360^\circ} \cdot \alpha_G \text{ [mm]}$$

$$\Delta s_{\text{encoder}} = \frac{D \cdot \pi}{i \cdot z} \text{ [mm] or}$$

$$\Delta s_{\text{encoder}} = \frac{h_{\text{Sp}}}{i \cdot z} \text{ with spindle drives [mm]}$$

$$\Delta s_{\text{total}} = \Delta s_{\text{gear}} + \Delta s_{\text{encoder}} + \Delta s_{\text{mech}} \text{ (steady-state) [mm]}$$

Δs_{mech} is the imprecision of the mechanical system such as due to expansion of the toothed belt in mm.

α_G Torsional angle of the gear [degree]

z Number of pulses per encoder revolution

D Drive wheel/pinion diameter [mm]

h_{Sp} Spindle pitch [mm]

i Gear transmission ratio

In the case of a pure acceleration drive without additional forces and torques, the optimum gear transmission ratio for the smallest motor torque and thus also for the smallest motor current can be calculated as follows for a given motor:

$$i_{\text{opt}} = \sqrt{\frac{J_{\text{load}}}{J_{\text{Mot}}}}$$

It is, however, not always possible to implement this optimum gear transmission ratio, e.g. if the resulting motor speed is too high.

11.4 Selection of the motor

- ◆ The motor is selected according to the following criteria:
- ◆ Adherence to the dynamic limits, i.e. all M, n points of the load cycle must be below the limit curve.
- ◆ The motor speed must be smaller than $n_{\max \text{ perm}}$. With synchronous servomotors, the maximum motor speed should not be greater than the rated speed. With induction servomotors, the maximum motor speed must not be more than 1.2 times the rated speed in the field weakening area.
- ◆ Adherence to the thermal limits, i.e. with synchronous servomotors, the motor rms torque at the mean motor speed resulting from the load cycle must be below the S1 curve. With induction servomotors, the rms value of the motor current within a load cycle must be smaller than the rated current of the motor.

When synchronous servomotors are used, it must be borne in mind that the maximum permissible motor torque at high speeds is reduced by the voltage-limit curve. In addition, a distance of about 10 % should be kept to from the voltage-limit curve as a protection against voltage fluctuations.

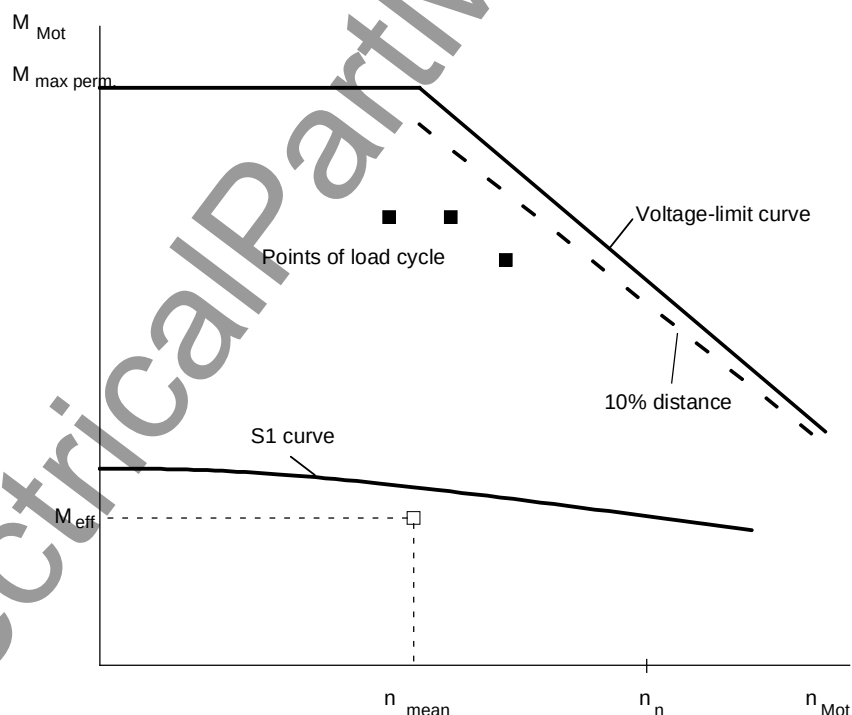


Fig. 11-7 Limit curves for 1FK6/1FT6 motors (synchronous servomotors)

If induction servomotors are used, the permissible motor torque in the field-weakening range is reduced by the stalling limit. Here, a distance of approximately 30 % should be kept to.

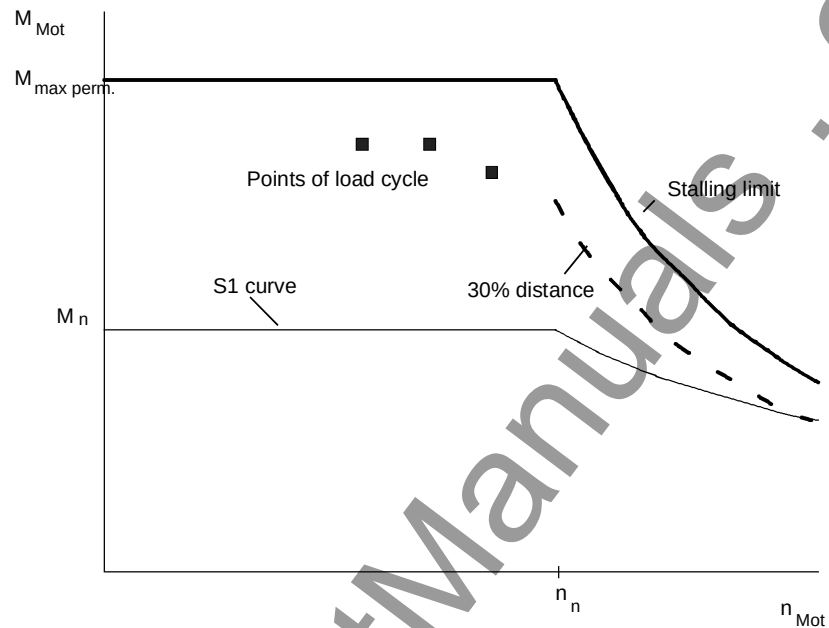
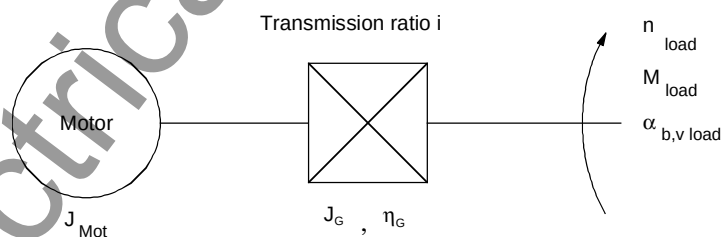


Fig. 11-8 Limit curves for 1PA6 motors (induction servomotors)

In order to keep a check on the dynamic limits, the relevant points of the torque curve must be calculated. In general, the motor torque at maximum speed during the acceleration phase is decisive. The motor torque and motor speed are calculated as follows when load torque, speed under load and angular acceleration/deceleration on the gear-output side are known:



$$M_{\text{Mot}} = J_{\text{Mot}} \cdot i \cdot \alpha_{b,v \text{ load}} + J_G^* \cdot i \cdot \alpha_{b,v \text{ load}} + M_{\text{load}} \cdot \frac{1}{i \cdot \eta_G \cdot \text{Sign}(M_{\text{load}})}$$

$$n_{\text{Mot}} = i \cdot n_{\text{load}}$$

J_{Mot} Motor inertia

J_G^* Gear moment of inertia referred to motor speed

η_G Gear efficiency

With the motor torque of lifting drives, a difference is made between upwards and downwards:

$$M_{\text{Mot up}} = J_{\text{Mot}} \cdot i \cdot \alpha_{b,v \text{ load}} + J_G^* \cdot i \cdot \alpha_{b,v \text{ load}} + M_{\text{load up}} \cdot \frac{1}{i \cdot \eta_G \cdot \text{Sign}(M_{\text{load up}})}$$

$$M_{\text{Mot down}} = J_{\text{Mot}} \cdot i \cdot \alpha_{b,v \text{ load}} + J_G^* \cdot i \cdot \alpha_{b,v \text{ load}} + M_{\text{load down}} \cdot \frac{\text{Sign}(M_{\text{load down}})}{i \cdot \eta_G}$$

$\alpha_{b,v \text{ load}}$ and M_{load} have to be applied with the correct signs (see also the examples under 11.3). If further moments of inertia are present on the motor side (e.g. coupling), these must also be taken into account.

In addition to the torque determined by the load and by the gears, the torque required for accelerating or decelerating the rotor's moment of inertia during dynamic processes is added to the motor torque.

$$M_{b,v \text{ Mot}} = J_{\text{Mot}} \cdot i \cdot \alpha_{b,v \text{ load}}$$

Now a motor has to be selected which fulfils the condition for the maximum motor torque in the required speed range. The proportion of the acceleration torque for the motor rotor in relation to the maximum motor torque depends on the motor's moment of inertia and the angular acceleration but also on the moment of inertia of the load, the gear transmission ratio and the static load torque.

A second point to be checked is whether the thermal limits are adhered to.

Synchronous servomotors

In order to calculate the rms torque, the motor torque must be determined in all parts of the travel curve. The following formula is used to calculate the rms torque and the mean motor speed:

$$M_{\text{eff}} = \sqrt{\frac{\sum M_{\text{Mot } i}^2 \cdot \Delta t_i}{T}}$$

$$n_{\text{mean}} = \frac{\sum \frac{|n_{\text{Mot } A} + n_{\text{Mot } E}|}{2} \cdot \Delta t_i}{T}$$

T Cycle time

$M_{\text{Mot } i}$ Motor torque in time segment Δt_i

$$\frac{|n_{\text{Mot } A} + n_{\text{Mot } E}|}{2}$$

Mean motor speed in time segment Δt_i

(A: initial value, E: final value)

When calculating the mean motor speed, it must be borne in mind that the initial value and the final value of the motor speed should not have different signs. Thus, an interpolation point must exist for every zero passage.

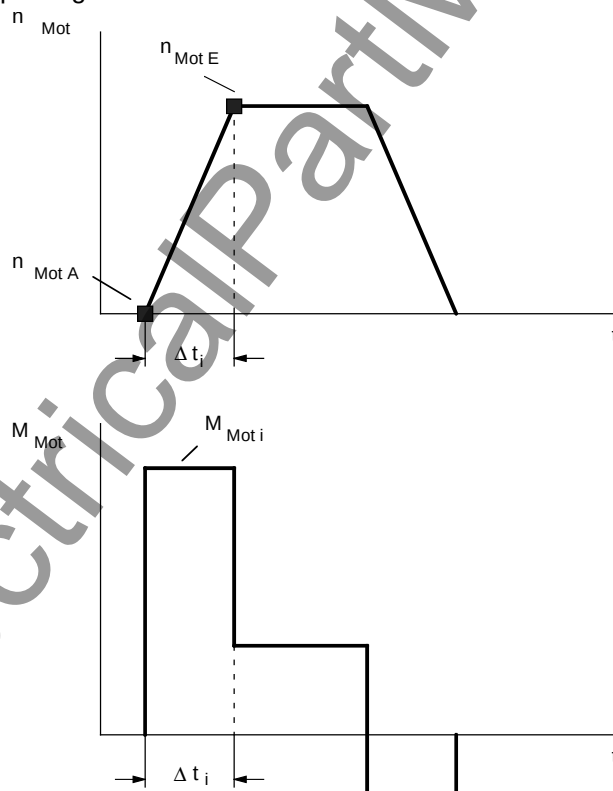


Fig. 11-9 Example of motor speed and motor torque in a time segment Δt_i

If the rms torque at the mean motor speed is below the S1 curve and the dynamic limits are being adhered to, the selected synchronous servomotor can be used.

Induction servomotors

In order to calculate the motor's rms current, the motor torque in all parts of the travel curve must first be determined. The motor current is thus calculated as follows:

$$I_{\text{Mot}} = I_n \cdot \sqrt{\left(\frac{M_{\text{Mot}}}{M_n}\right)^2 \cdot \left(1 - \left(\frac{I_{\mu n}}{I_n}\right)^2\right) \cdot k_n^2 + \left(\frac{I_{\mu n}}{I_n}\right)^2 \cdot \frac{1}{k_n^2}}$$

$I_{\mu n}$ Rated magnetizing current

$k_n = 1$ In the constant flux range

$k_n = \frac{n}{n_n}$ In the field weakening range

The rms value of the motor current is calculated as follows:

$$I_{\text{eff}} = \sqrt{\frac{\sum \left(\frac{I_{\text{Mot A}} + I_{\text{Mot E}}}{2}\right)^2 \cdot \Delta t_i}{T}}$$

$$\frac{I_{\text{Mot A}} + I_{\text{Mot E}}}{2}$$

Mean motor current in time segment Δt_i

(A: initial value, E: final value)

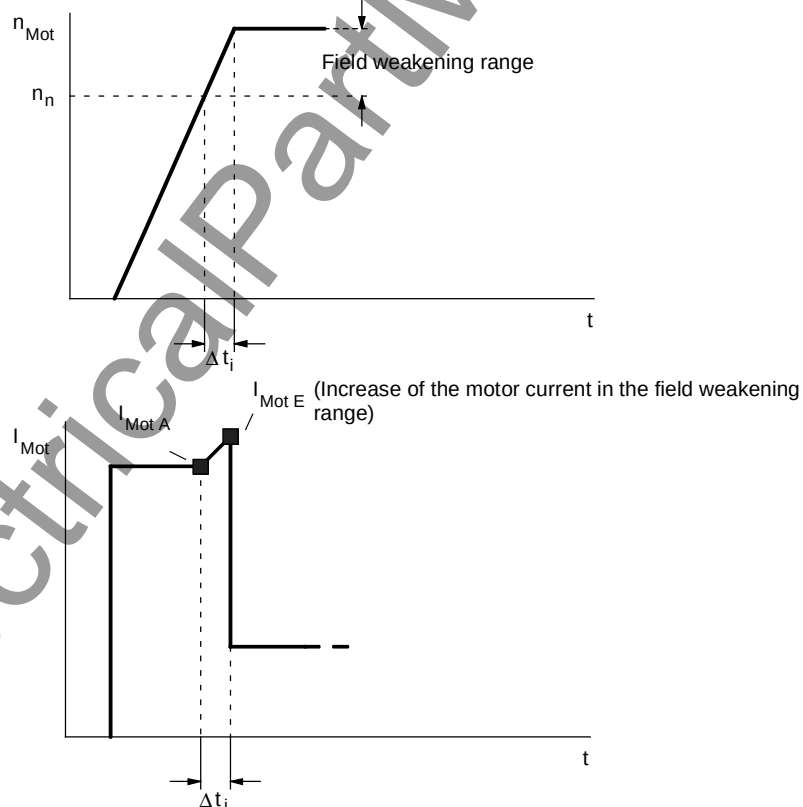


Fig. 11-10 Example of motor speed and motor current in a time segment Δt_i

If the dynamic limits are being adhered to and the rms value of the motor current is smaller than the motor's rated current, the selected induction servomotor can be used.

Encoders

Which encoder is selected depends on the requirements in each case. Encoders provide high resolution and extremely true running at the lowest speeds. They are thus especially suitable for highly accurate positioning tasks. Resolvers are robust and inexpensive and provide good resolution. Absolute-value encoders maintain the absolute position even after the power supply is de-energized which means that a new approach to reference point is not necessary with positioning drives. Contrary to synchronous servomotors, induction servomotors do not require a rotor position sensor such as an encoder or a resolver for motor control. A pulse encoder is adequate here.

11.5 Selection of converters or inverters

With single-axis drives, a converter now has to be selected and, with multi-axis drives, an inverter is necessary. The selection criteria are the same for both:

- ◆ The maximum motor current must be smaller than the maximum permissible output current of the converter/inverter. If three times the rated current is utilized in the case of the Compact PLUS unit (3 times at a 5 kHz pulse frequency, 2.1 times at a 10 kHz pulse frequency), this current must not flow for longer than 250 ms and then a pause of 750 ms has to be observed with only 0.91 times the rated current, otherwise 1.6 times the rated current is permissible for 60 s (see technical data).
- ◆ The arithmetic mean value of the motor current must be smaller than the rated current of the converter/inverter with a maximum cycle time of 300 s.

The second condition arises from the fact that the switching losses and forward losses in the inverter are approximately proportional to the output current. The rms value can also be calculated instead of the arithmetic mean value. One is then more on the safe side, but it requires more calculation work.

In order to determine the motor current at a given motor torque, the following formula is used:

- ◆ For synchronous servomotors

$$I_{\text{Mot}} = \frac{M_{\text{Mot}}}{kT_n} \quad \text{for } M_{\text{Mot}} \leq M_0$$

kT_n Torque constant in Nm/A
 M_0 Standstill torque

In general, the maximum motor current occurs during the acceleration phase. At motor torques $> M_0$, the motor current may possibly be higher than calculated with kTn due to saturation effects. In this case, the motor current is calculated as follows:

$$I_{\text{Mot}} = \frac{M_{\text{Mot}}}{kTn \cdot \left(1 - \left(\frac{M_{\text{Mot}} - M_0}{M_{\text{max}} - M_0}\right)^2 \cdot \left(1 - \frac{M_{\text{max}} \cdot I_0}{M_0 \cdot I_{\text{max}}}\right)\right)} \quad \text{for } M_{\text{Mot}} > M_0$$

I_0 Standstill current

M_{max} Maximum permissible motor torque

I_{max} Maximum permissible motor current

◆ For induction servomotors

Calculation of the motor current is as described under 11.4.

Acceleration into the field-weakening range with a constant motor torque results in the maximum motor current in the field-weakening range at maximum speed.

The following formula is used to calculate the arithmetic mean of the motor current:

- For synchronous servomotors

$$I_{\text{Mot mean}} \approx \frac{\sum |M_{\text{Mot } i}| \cdot \Delta t_i}{kTn \cdot T}$$

$M_{\text{Mot } i}$ Motor torque in time segment Δt_i

T Cycle time

- For induction servomotors

$$I_{\text{Mot mean}} = \frac{\sum \frac{I_{\text{Mot A}} + I_{\text{Mot E}}}{2} \cdot \Delta t_i}{T}$$

$\frac{I_{\text{Mot A}} + I_{\text{Mot E}}}{2}$ Mean motor current in time segment Δt_i
(A: initial value, E: final value)

11.6 Selection of the rectifier unit for multi-axis drives

When multi-axis drives are used, several inverters are supplied with power by a rectifier unit. When the rectifier unit is being selected, it must be determined whether all the drives can work at the same time. The criteria for making the selection are as follows:

- ◆ The maximum DC link current occurring must be smaller than the maximum permissible output current of the rectifier unit. In the case of a Compact PLUS rectifier unit, if three times the rated current is utilized, this current must not flow for longer than 250 ms, otherwise 1.6 times the rated current is permissible for 30 s (see technical data). If a Compact PLUS rectifier unit is not used, the maximum output current must not exceed 1.36 times the rated current for a time of 60 s (see technical data).
- ◆ The arithmetic mean value of the DC link current must be smaller than the rated value of the DC link current of the rectifier unit when the maximum cycle time is 300 s.

The second condition arises from the fact that the forward losses in the rectifier are approximately proportional to the DC link current. The rms value can also be calculated instead of the arithmetic mean value. One is then more on the safe side, but it requires more calculation work.

The DC link current is calculated as follows:

$$I_{\text{Link Rect}} = \sum I_{\text{Link Inv}}$$

$$I_{\text{Link Inv}} = \frac{P_{\text{Mot}}}{\eta_{\text{Mot}} \cdot \eta_{\text{Inv}} \cdot U_{\text{Link}}} \quad \text{DC link current of an inverter in motor operation}$$

$$U_{\text{Link}} = 1.35 \cdot U_{\text{Line}} \quad \text{DC link voltage}$$

$$P_{\text{Mot}} = \frac{M_{\text{Mot}} \cdot n_{\text{Mot}}}{9.55} \quad \text{Motor output in W}$$

M_{Mot} Motor torque in Nm

n_{Mot} Motor speed in rpm

η_{Mot} Motor efficiency

η_{Inv} Inverter efficiency (≈ 0.98)

When the rectifier is being selected, only motor operation needs to be considered. The maximum DC link current occurs when all the motors connected to the inverters have to simultaneously produce the maximum motor output. If this is not the case, the rectifier unit can be smaller. The total number of connected inverters, however, must not be too large because, otherwise, precharging of the rectifier unit can be overloaded (see technical data).

In order to determine the arithmetic mean value of the DC link current, the mean values of the individual inverters are added together. For one inverter, the calculation is as follows:

$$I_{\text{Link Inv mean}} = \frac{P_{\text{Mot mean}}}{\eta_{\text{Mot}} \cdot \eta_{\text{Inv}} \cdot U_{\text{Link}}}$$

$$P_{\text{Mot mean}} = \frac{\sum \frac{P_{\text{Mot A}} + P_{\text{Mot E}}}{2} \cdot \Delta t_i}{T}$$

$$\frac{P_{\text{Mot A}} + P_{\text{Mot E}}}{2}$$

Mean motor output in time segment Δt_i [W]

(A: initial value, E: final value)

T Cycle time

Only positive motor outputs are evaluated. When calculating the mean motor output, one must make sure that the initial value and the final value of the motor speed do not have different signs. An interpolation point must therefore exist for every zero passage.

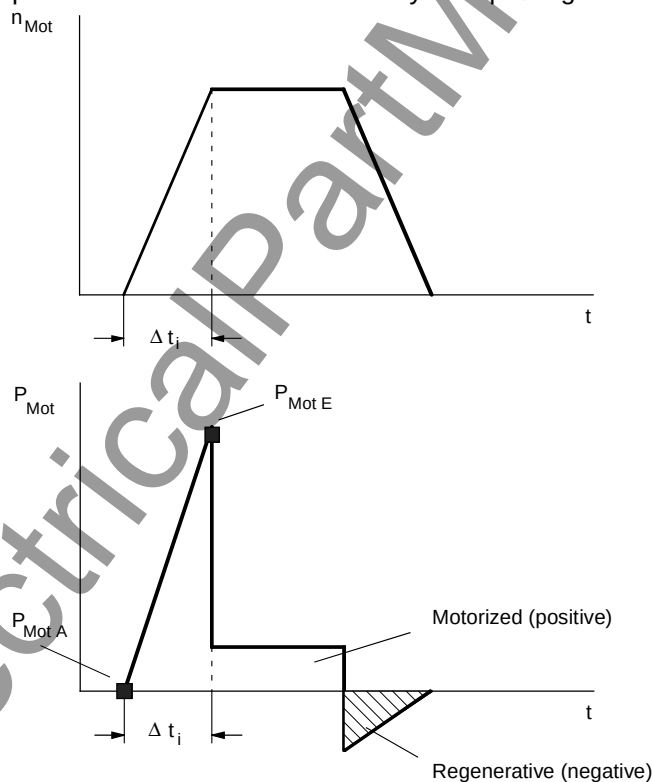


Fig. 11-11 Example of motor speed and motor output in a time segment Δt_i

Adding the mean values for the individual inverters gives the mean value for the rectifiers as follows:

$$I_{\text{Link Rect mean}} = \sum I_{\text{Link Inv mean}}$$

11.7 Selection of the braking units and braking resistors

Compact PLUS

On Compact PLUS units, the choppers for the braking resistors are provided in the converters and in the rectifier unit (in the case of multi-axis drives with several inverters).

For further information regarding the selection of the braking resistor, please refer to the MASTERDRIVES Motion Control Catalog DA65.11, Chapter 3.

The following criteria apply to the braking resistors:

- ◆ The maximum braking power which occurs must be smaller than $15 \cdot P_{20}$. This power must not occur for more than 3 s (see technical data).
- ◆ The mean braking power must be smaller than $P_{20} / 4.5$ with a maximum cycle time of 90 s

Compact and chassis type units

The braking units for Compact and chassis type units are autonomous components. The braking units, up to a power of $P_{20} = 20$ kW, have an internal braking resistor. Instead of the internal braking resistor, an external braking resistor can be used to increase the continuous power output. The following criteria apply to the selection process:

- ◆ The maximum braking power which occurs must be smaller than $15 \cdot P_{20}$. This power must not occur for longer than 0.4 s when there is an internal braking resistor or 3 s when there is an external braking resistor (see technical data).
- ◆ The mean braking power must be smaller than $P_{20} / 36$ when an internal braking resistor is used or smaller than $P_{20} / 4.5$ when an external braking resistor is used. The maximum cycle time is 90 s.

For further information regarding the selection of the braking resistor, please refer to the MASTERDRIVES Motion Control Catalog DA65.11, Chapter 3.

Braking power

The braking power is calculated as follows:

$$P_{br} = P_{Mot v} \cdot \eta_{Mot} \cdot \eta_{Inv}$$

$$P_{Mot v} = \frac{M_{Mot v} \cdot n_{Mot}}{9550}$$

Motor output during braking in kW

$M_{Mot v}$ Motor torque during braking in Nm

n_{Mot} Motor speed in rpm

The maximum motor braking power $P_{Mot v \max}$ generally occurs at the beginning of deceleration when the motor is running at maximum speed. If several inverters are operated from one rectifier unit, a check must be made to see whether several drives can brake simultaneously. In the event of an emergency stop, all drives may have to be shut down at the same time.

The mean braking power is calculated as follows:

$$P_{br \text{ mean}} = \frac{\sum \frac{P_{Mot \text{ v } A} + P_{Mot \text{ v } E}}{2} \cdot \Delta t_i}{T} \cdot \eta_{Mot} \cdot \eta_{Inv}$$

$\frac{P_{Mot \text{ v } A} + P_{Mot \text{ v } E}}{2}$ Mean motor braking power in time segment Δt_i
 (A: initial value, E: final value)

T Cycle time

Only negative motor outputs are evaluated. When calculating the mean motor output, it must be borne in mind that the initial value and the final value of the motor speed do not have different signs. An interpolation point must therefore exist for every zero passage.

If several inverters are connected to one rectifier unit, the mean value is calculated by adding together the individual mean values for the inverters.

11.8 Selection of other components

The selection tables in Catalog DA65.11 are used to make a list of the other components needed on the rectifier side and on the load side.

Rectifier side	Load side
Line fuses	Output reactor
Line switch	
Line contactor	
Line reactor	
Line filter	

Table 11-1 Selection of other components

Line fuses

Line fuses, or circuit-breakers in the lower output range, are generally always necessary. In addition to line protection, fuses with gR characteristics also protect semiconductors (rectifier). Fuses with gL characteristics or circuit-breakers are only for line protection; in the event of a fault in the rectifier or in the DC link, the semiconductors of the rectifier are not protected. Fuses with gL characteristics or circuit-breakers are therefore appropriate if units have to be replaced in the event of a fault. If repairs have to be carried out locally, e.g. for large outputs, the use of fuses with gR characteristics is recommended.

Line switches	Line switches are used to disconnect the voltage on converters or rectifier units. Depending on the customer's requirements, line switches can be executed as main and emergency OFF switches (for installing in doors), as load disconnectors with and without fuses, or as fuse switch disconnectors.
Line contactor	The converter or the rectifier unit can be disconnected from the voltage supply in the event of a fault by means of the line contactor or also via the OFF command. The use of a line contactor prevents other components, for example, pre-charging resistors, braking resistors, from being damaged if there is a fault.
Line reactor	A line reactor reduces the harmonics of the system on the one hand and protects the DC link capacitors from excessive current spikes on the other. A line reactor with 2 % u_k is necessary from a ratio: System fault power > 33 x rated converter output or if a rectifier unit is used together with inverters: System fault power > 33 x total rated inverter outputs
Line filters	Line filters are necessary if a certain radio interference level in accordance with EN 55011 has to be maintained (class A1 for chassis type units and B1 for Compact and Compact PLUS type units). The A1 or B1 radio interference level can only be maintained in conjunction with a 2 % u_k line reactor and shielded motor cables. On the Compact PLUS type unit, the line reactor is contained in the line filter.
Output reactors, sinusoidal filters, dv/dt filters	It is not permissible to use output reactors, sinusoidal filters and dv/dt filters in the case of MASTERDRIVES Motion Control.

Notes regarding the use of a power back-up module

The power back-up module is for increasing the capacity of the DC link. This can bridge a short-time power-system failure, on the one hand, and also enables intermediate storage of braking energy, on the other.

- ◆ Storage capacity in the event of a power failure:

$$W = \frac{1}{2} \cdot C \cdot (U_{\text{Link n}}^2 - U_{\text{Link min}}^2)$$

With a 400 V supply voltage and when $C = 5.1 \text{ mF}$ and $U_{\text{Link min}} = 400 \text{ V}$, for example, the storage capacity is calculated as follows:

$$W = \frac{1}{2} \cdot 5.1 \cdot 10^{-3} \cdot ((135 \cdot 400)^2 - 400^2) = 336 \text{ Ws}$$

With a 460 V supply voltage, the storage capacity increases to 575 W. The possible bridging time t_{bri} is calculated with the output power P as follows:

$$t_{\text{bri}} = \frac{W}{P}$$

The storage capacity during regenerative operation is calculated as follows:

$$W = \frac{1}{2} \cdot C \cdot (U_{\text{Link max}}^2 - U_{\text{Link n}}^2)$$

With a 400 V supply voltage and when $U_{\text{Link max}} = 750 \text{ V}$:

$$W = \frac{1}{2} \cdot 5.1 \cdot 10^{-3} \cdot (750^2 - (135 \cdot 400)^2) = 691 \text{ Ws}$$

During braking, for example, from maximum speed to 0 within time t_v the braking energy is calculated as follows:

$$W_{\text{br}} = \frac{1}{2} \cdot P_{\text{br max}} \cdot t_v$$

with maximum motor braking power in W

$$P_{\text{br max}} = \frac{M_{\text{Mot v max}} \cdot \eta_{\text{Mot max}}}{9.55} \cdot \eta_{\text{Mot}} \cdot \eta_{\text{Inv}}$$

$M_{\text{Mot v max}}$ Maximum motor torque during braking in Nm

$\eta_{\text{Mot max}}$ Maximum motor speed during braking in rpm

- ◆ Maximum number of power back-up modules which can be connected on a Compact PLUS unit is

- Two power back-up modules for rectifier units
- One power back-up module for converters

Notes on pulse frequency

The level of the pulse frequency basically affects the dynamic response. Accordingly, when a high level of dynamic response is required, the pulse frequency should be set to 10 kHz. Use of the Compact PLUS does not therefore entail derating. Compact units and chassis-type units require derating of 6 or 3 Hz and upwards, depending on their power output (see technical data). A reduction of the permissible rated current entails a reduction of the permissible maximum current to the same amount. In addition, the maximum pulse frequency with chassis-type units is lower than 10 kHz (see technical data).

11.9 Calculating example

A three-axis conveyor vehicle is to be designed. The x-axis is the main propelling drive, the y-axis is the fork drive and the z-axis is the lifting drive. The propelling drive and the lifting drive can be operated simultaneously whereas the fork drive only operates alone. The x-axis and the y-axis are driven via toothed belts. The z-axis is driven via a gear rack. Three inverters are to be used on one rectifier unit. Positioning is to be carried out non-centrally in the inverter. The Profibus is to be used for connection to a PLC.

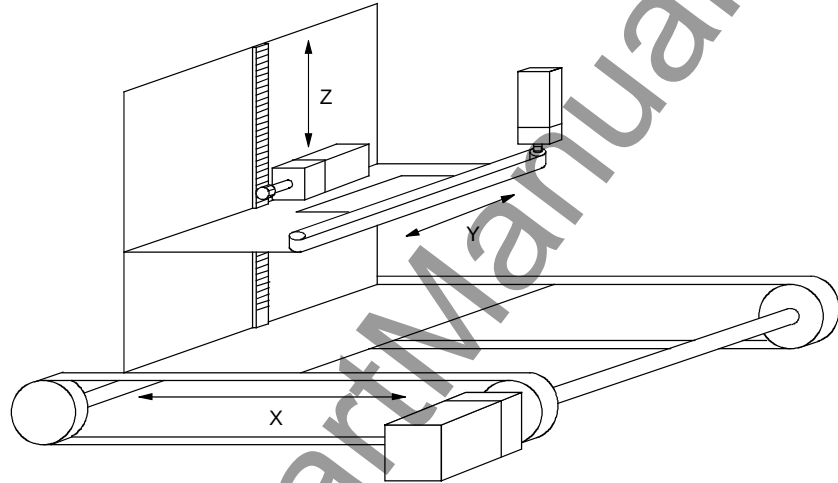


Fig. 11-12 Line drawing of a three-axis conveyor vehicle

11.9.1 Calculation of the x-axis as the travel gear

1. Data of the drive	◆ Mass to be transported	$m = 400 \text{ kg}$
	◆ Diameter of drive wheel	$D = 0.14 \text{ m}$
	◆ Max. speed	$v_{\max} = 1.6 \text{ m/s}$
	◆ Max. acceleration and deceleration	$a_{\max} = 6.4 \text{ m/s}^2$
	◆ Distance travelled	$s = 2 \text{ m}$
	◆ Cycle time	$T = 7 \text{ s}$
	◆ Mech. efficiency	$\eta_{\text{mech}} = 0.9$
	◆ Specific travelling resistance	$w_f = 0.1$
	◆ Mech. accuracy	$\Delta s_{\text{mech}} = \pm 0.1 \text{ mm}$
	◆ Overall accuracy required	$\Delta s_{\text{tot}} = \pm 0.2 \text{ mm}$

2. Travel curve

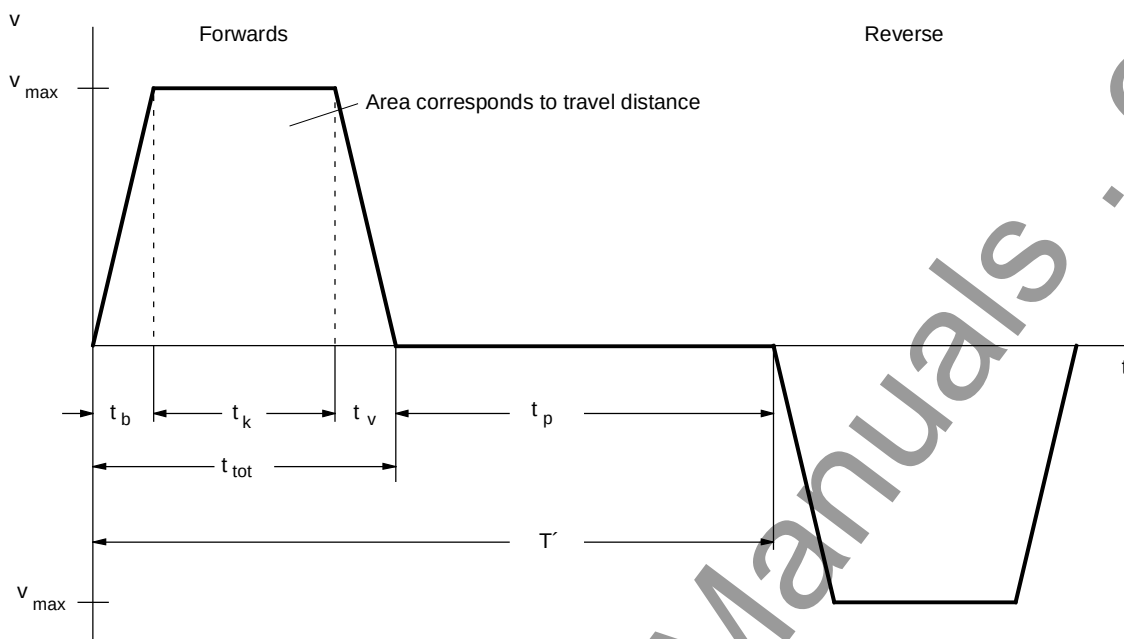


Fig. 11-13 Travel curve for forwards and reverse travel

It is sufficient to only consider forwards travel because the conditions are the same for forwards and reverse travel.

- ◆ The new cycle time is therefore:

$$T' = \frac{T}{2}$$

- ◆ For the remaining values of the travel curve, the following is obtained:

$$t_b = t_v = \frac{v_{\max}}{a_{\max}} = \frac{16}{6.4} = 0.25 \text{ s}$$

$$t_k = \frac{s - v_{\max} \cdot \frac{t_b}{2} - v_{\max} \cdot \frac{t_v}{2}}{v_{\max}} = \frac{2 - 16 \cdot \frac{0.25}{2} - 16 \cdot \frac{0.25}{2}}{16} = 1 \text{ s}$$

$$t_{\text{tot}} = t_b + t_k + t_v = 0.25 + 1 + 0.25 = 1.5 \text{ s}$$

$$t_p = T' - t_{\text{tot}} = 3.5 - 1.5 = 2 \text{ s}$$

3. Max. speed under load, max. load torque, selection of the gear unit

- ◆ Max. speed under load at the drive wheel

$$n_{\text{Load max}} = \frac{v_{\text{max}} \cdot 60}{\pi \cdot D} = \frac{16 \cdot 60}{\pi \cdot 0.14} = 218.27 \text{ rpm}$$

A gear transmission ratio of $i=10$ is selected here. A synchronous servomotor can thus be used with a rated speed of 3000 rpm.

$$n_{\text{Mot max}} = i \cdot n_{\text{Load max}} = 10 \cdot 218.27 = 2182.7 \text{ rpm}$$

- ◆ Resistance torque

$$M_W = m \cdot g \cdot w_f \cdot \frac{D}{2} = 400 \cdot 9.81 \cdot 0.1 \cdot \frac{0.14}{2} = 27.47 \text{ Nm}$$

- ◆ Acceleration and deceleration torque for the load

$$\alpha_{\text{load}} = a_{\text{max}} \cdot \frac{2}{D} = 6.4 \cdot \frac{2}{0.14} = 914 \text{ s}^{-2}$$

$$J_{\text{load}} = m \cdot \left(\frac{D}{2}\right)^2 = 400 \cdot \left(\frac{0.14}{2}\right)^2 = 196 \text{ kgm}^2$$

$$M_{b,v \text{ load}} = J_{\text{load}} \cdot \alpha_{\text{load}} = 196 \cdot 914 = 179.2 \text{ Nm}$$

- ◆ Max. torque on the output side of the gear unit

$$M_{\text{load max}} = (M_{b \text{ load}} + M_W) \cdot \frac{1}{\eta_{\text{mech}}}$$

$$= (179.2 + 27.47) \cdot \frac{1}{0.9} = 229.6 \text{ Nm}$$

An SPG140-M1 planetary gear unit for mounting on 1FT6 motors is therefore used where

$$M_{\text{max}} = 400 \text{ Nm at } i=10$$

$$J_G^* = 0.001 \text{ kgm}^2 \quad \text{moment of inertia referred to motor}$$

$$\eta_G = 0.95 \quad \text{gear unit efficiency}$$

$$\alpha_G = 3' \quad \text{torsional play}$$

- ◆ Acceleration and deceleration torque for the gear unit

$$M_{b,v G} = J_G^* \cdot \alpha_{\text{load}} \cdot i = 0.001 \cdot 914 \cdot 10 = 0.914 \text{ Nm}$$

- ◆ Positioning accuracy

$$\Delta s_{\text{gear}} = \frac{D \cdot \pi}{360^\circ} \cdot \frac{\alpha_G}{60} = \frac{0.14 \cdot \pi}{360} \cdot \frac{3}{60} = 0.061 \text{ mm}$$

i.e. $\pm 0.0305 \text{ mm}$

$$\Delta s_{\text{encoder}} = \frac{D \cdot \pi}{i \cdot z} = \frac{0.14 \cdot \pi}{10 \cdot 4096} = \pm 0.01 \text{ mm with an 8-pole resolver}$$

$$\Delta s_{\text{tot}} = \Delta s_{\text{mech}} + \Delta s_{\text{gear}} + \Delta s_{\text{encoder}}$$

$$= 0.1 + 0.0305 + 0.01 = 0.1405 < 0.2 \text{ mm}$$

The required accuracy is thus complied with.

4. Selection of the motor

Selection with regard to the dynamic limit curve

- ◆ The maximum motor torque occurs here because the deceleration is equal to the acceleration.

$$\begin{aligned} M_{\text{Mot max}} &= M_{\text{b Mot}} + M_{\text{b G}} + (M_{\text{b load}} + M_{\text{W}}) \cdot \frac{1}{i \cdot \eta_{\text{mech}} \cdot \eta_{\text{G}}} \\ &= M_{\text{b Mot}} + 0.914 + (179.2 + 27.47) \cdot \frac{1}{10 \cdot 0.9 \cdot 0.95} \\ &= M_{\text{b Mot}} + 25.08 \text{ Nm} \end{aligned}$$

$$\text{where } M_{\text{b Mot}} = J_{\text{Mot}} \cdot \alpha_{\text{load}} \cdot i = J_{\text{Mot}} \cdot 914 \cdot 10 = J_{\text{Mot}} \cdot 914 \text{ s}^{-2}$$

The first 1FT6 motor with $n_n=3000 \text{ rpm}$, which satisfies the condition or the dynamic limit curve, is the 1FT6084-8AF7 with $P_n=4.6 \text{ kW}$, $M_n=14.7 \text{ Nm}$, $M_{\text{max perm}}=65 \text{ Nm}$, $J_{\text{Mot}}=0.0065 \text{ kgm}^2$ (with brake), $k_{Tn100}=1.34 \text{ Nm/A}$, $\eta_{\text{Mot}}=0.92$; $M_0=20 \text{ Nm}$

- ◆ The acceleration and deceleration torque for the motor rotor is thus:

$$M_{\text{b,v Mot}} = 0.0065 \cdot 914 = 5.94 \text{ Nm}$$

- ◆ The maximum motor torque is equal to the motor torque during acceleration:

$$M_{\text{Mot max}} = M_{\text{Mot b}} = 5.94 + 25.08 = 31.03 \text{ Nm}$$

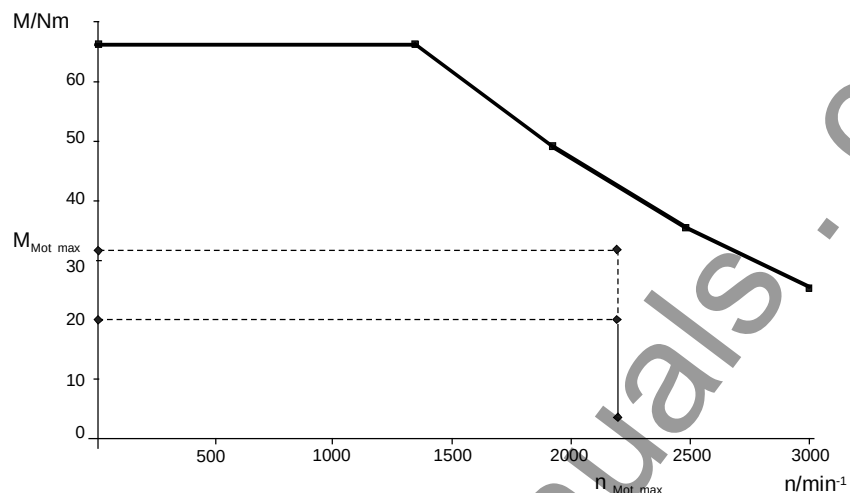


Fig. 11-14 Dynamic limit curve for the 1FT6084-8AF7 with the points of the load cycle

As a check on the thermal limits, the effective motor torque is calculated. This is done by determining all the motor torques within the travel curve in addition to the motor torque during acceleration.

- ♦ Motor torque during constant travel

$$M_{\text{Mot k}} = M_W \cdot \frac{1}{i \cdot \eta_{\text{mech}} \cdot \eta_G} = 27.47 \cdot \frac{1}{10 \cdot 0.9 \cdot 0.95} = 3.21 \text{ Nm}$$

- ♦ Motor torque during deceleration

$$M_{\text{Mot v}} = -M_{\text{v Mot}} - M_{\text{v G}} + (-M_{\text{v Load}} + M_W) \cdot \frac{1}{i \cdot (\eta_{\text{mech}} \cdot \eta_G) \cdot \text{sign}(-M_{\text{v load}} + M_W)}$$

$$= -5.94 - 0.914 + (-179.2 + 27.47) \cdot \frac{0.9 \cdot 0.95}{10} = -19.83 \text{ Nm}$$

Here, the proportion of deceleration torque outweighs the resistance torque. Regenerative operation occurs. In this case, the efficiency levels are above the line (the sign before the bracketed term " $-M_{\text{v load}} + M_W$ " is negative).

The torque characteristic can be determined using the values calculated for the motor torque.

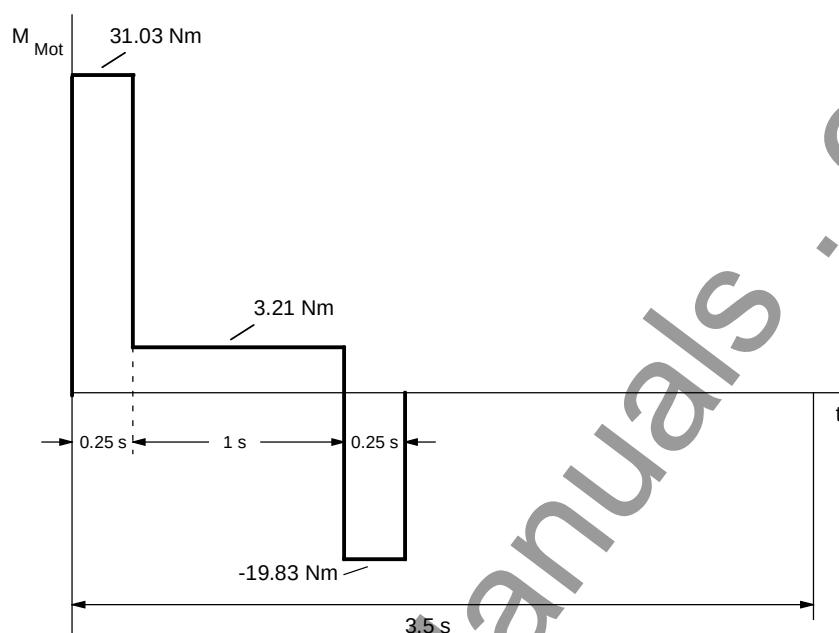


Fig. 11-15 Torque characteristic for forwards travel

- ◆ The effective motor torque is obtained from the torque characteristic as follows:

$$M_{eff} = \sqrt{\frac{\sum M_{Mot i}^2 \cdot \Delta t_i}{T'}}$$

$$= \sqrt{\frac{3103^2 \cdot 0.25 + 3.21^2 \cdot 1 + 19.83^2 \cdot 0.25}{3.5}} = 10 \text{ Nm}$$

- ◆ By using the travel curve, which is proportional to the speed, the mean motor speed is obtained:

$$n_{mean} = \frac{\sum \frac{|n_A + n_E|}{2} \cdot \Delta t_i}{T'}$$

$$= \frac{\frac{2182.7}{2} \cdot 0.25 + 2182.7 \cdot 1 + \frac{2182.7}{2} \cdot 0.25}{3.5} = 779.5 \text{ rpm}$$

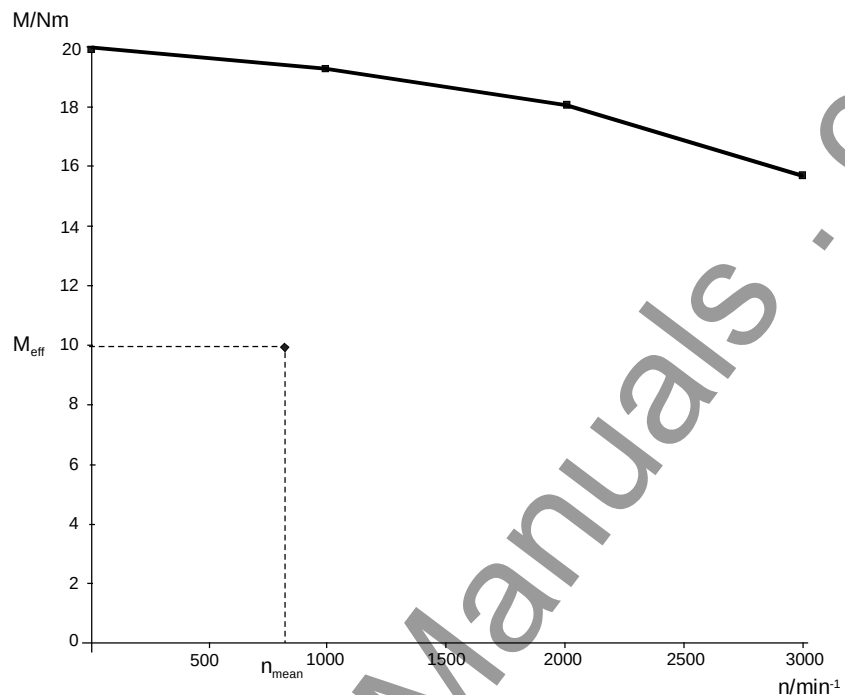


Fig. 11-16 S1 curve for the 1FT6084-8AF

The effective motor torque calculated is n_{mean} below the S1 curve. The motor is therefore suitable.

5. Selection of the inverter

The inverter is selected according to the maximum motor current and the mean value of the motor current.

- ◆ Maximum motor current (the saturation influence can be neglected here)

$$I_{\text{Mot max}} \approx \frac{M_{\text{Mot max}}}{k_{Tn100}} = \frac{3103}{134} = 23.16 \text{ A}$$

- ◆ Mean value of the motor current obtained from the magnitude of the torque characteristic

$$I_{\text{Mot mean}} \approx \frac{\sum |M_{\text{Mot } i}| \cdot \Delta t_i}{k_{Tn100} \cdot T'}$$

$$= \frac{3103 \cdot 0.25 + 3.21 \cdot 1 + 19.83 \cdot 0.25}{134 \cdot 3.5} = 3.4 \text{ A}$$

Because the accelerating and decelerating times are $\leq 0.25 \text{ s}$ and the time between is $\geq 0.75 \text{ s}$, a check is now made to see if three-times the rated current of a Compact PLUS inverter can be utilized when $I_{Un}=10.2 \text{ A}$.

- ◆ The following applies to the motor current during constant travel:

$$I_{\text{Mot k}} = \frac{M_{\text{Mot k}}}{k_{\text{Tn100}}} = \frac{3.21}{134} = 2.4 \text{ A}$$

- ◆ Thus:

$$I_{\text{Mot max}} = 23.16 \text{ A} < 3 \cdot I_{\text{Un}} = 30 \text{ A}$$

$$I_{\text{Mot mean}} = 3.4 \text{ A} < I_{\text{Un}} = 10.2 \text{ A}$$

$$I_{\text{Mot k}} = 2.4 \text{ A} < 0.91 \cdot I_{\text{Un}} = 9.3 \text{ A}$$

The 6SE7021-0TP50 Compact PLUS inverter can therefore be used when $I_{\text{Un}}=10.2 \text{ A}$.

6. Determination of the DC link currents

The maximum DC link current and the mean value of the DC link current for the inverter which occur during motor operation must be determined for later rating of the rectifier unit. To do this, all motor power output levels within the travel curve first have to be calculated.

- ◆ Max. power output of motor during acceleration

$$P_{\text{Mot b max}} = \frac{M_{\text{Mot b}} \cdot \eta_{\text{Mot max}}}{9550} = \frac{3103 \cdot 2182.7}{9550} = 7.09 \text{ kW}$$

- ◆ Power output of motor during constant travel

$$P_{\text{Mot k}} = \frac{M_{\text{Mot k}} \cdot \eta_{\text{Mot max}}}{9550} = \frac{3.21 \cdot 2182.7}{9550} = 0.734 \text{ kW}$$

- ◆ Max. power output of motor during deceleration

$$P_{\text{Mot v max}} = \frac{M_{\text{Mot v}} \cdot \eta_{\text{Mot max}}}{9550} = \frac{-19.83 \cdot 2182.7}{9550} = -4.53 \text{ kW}$$

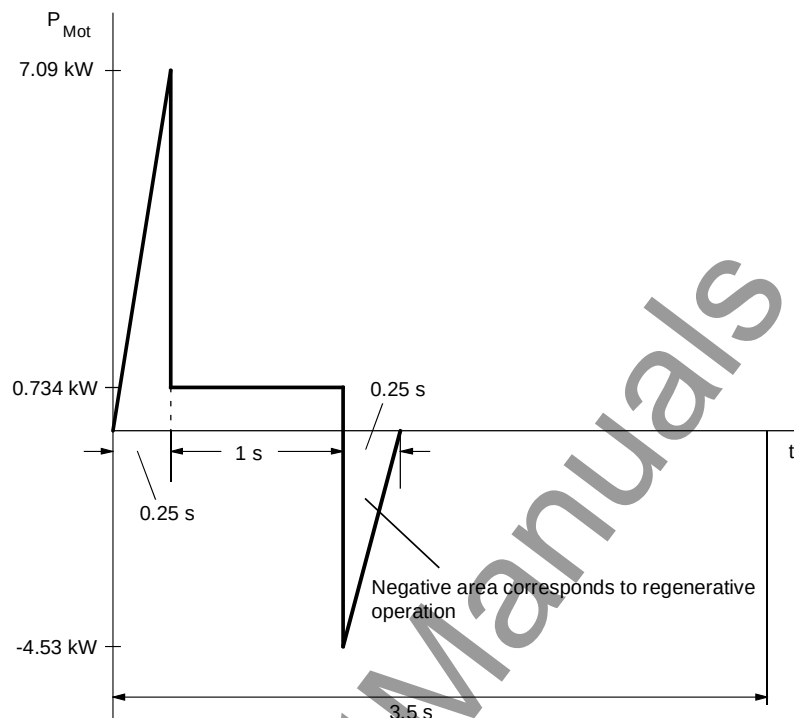


Fig. 11-17 Characteristic of the motor output for forwards travel

- ◆ The maximum DC link current during operation of the motor during acceleration is

$$I_{\text{Link Inv max}} = \frac{P_{\text{Mot max}}}{\eta_{\text{Mot}} \cdot \eta_{\text{Inv}} \cdot 1.35 \cdot U_{\text{line}}}$$

$$= \frac{7090}{0.92 \cdot 0.98 \cdot 1.35 \cdot 400} = 14.56 \text{ A}$$

- ◆ The mean motor power output during operation of the motor is calculated from the positive characteristic of the motor power output as follows:

$$P_{\text{Mot mean}} = \frac{\sum \frac{P_{\text{Mot A}} + P_{\text{Mot E}}}{2} \cdot \Delta t_i}{T'}$$

$$= \frac{\frac{1}{2} \cdot 7.09 \cdot 0.25 + 0.734 \cdot 1}{3.5} = 0.463 \text{ kW}$$

- ◆ The mean value of the DC link current is therefore:

$$I_{\text{Link mean}} = \frac{P_{\text{Mot mean}}}{\eta_{\text{Mot}} \cdot \eta_{\text{Inv}} \cdot 1.35 \cdot U_{\text{Line}}}$$

$$= \frac{463}{0.92 \cdot 0.98 \cdot 1.35 \cdot 400} = 0.95 \text{ A}$$

7. Determination of braking power

The maximum braking power and the mean braking power have to be calculated for later rating of the braking resistors. The maximum power output of the motor during braking has already been calculated (see 6.).

- ◆ The maximum braking power is therefore:

$$P_{br \max} = P_{Mot \ v \ max} \cdot \eta_{Mot} \cdot \eta_{Inv} = -4.53 \cdot 0.92 \cdot 0.98 = -4.08 \text{ kW}$$

- ◆ The mean braking power is obtained from the negative characteristic of the motor power output as follows:

$$P_{br \text{ mean}} = \frac{\sum \frac{P_{Mot \ v \ A} + P_{Mot \ v \ E}}{2} \cdot \Delta t_i}{T'} \cdot \eta_{Mot} \cdot \eta_{Inv}$$

$$= \frac{\frac{1}{2} \cdot (-4.53) \cdot 0.25}{3.5} \cdot 0.92 \cdot 0.98 = -0.146 \text{ kW}$$

11.9.2 Calculating the y-axis as the travel gear

1. Data of the drive	◆ Mass to be transported	$m = 100 \text{ kg}$
	◆ Diameter of drive wheel	$D = 0.1 \text{ m}$
	◆ Max. speed	$v_{\max} = 1 \text{ m/s}$
	◆ Max. acceleration and deceleration	$a_{\max} = 2.5 \text{ m/s}^2$
	◆ Distance travelled	$s = 0.5 \text{ m}$
	◆ Cycle time	$T = 7 \text{ s}$
	◆ Mech. efficiency	$\eta_{\text{mech}} = 0.9$
	◆ Specific travelling resistance	$w_f = 0.1$
	◆ Mech. accuracy	$\Delta s_{\text{mech}} = \pm 0.1 \text{ mm}$
	◆ Overall accuracy required	$\Delta s_{\text{tot}} = \pm 0.2 \text{ mm}$

NOTE

The same calculating procedures apply to the y-axis as the propelling drive as to the x-axis. This calculation is therefore dispensed with.

With $i=10$, the motor selected is a 1FT6041-4AF7 motor with a SPG75-M1 gear unit and the smallest 6SE7012-0TP50 Compact PLUS inverter with $I_{UN}=2 \text{ A}$. Because the drive of the y-axis always runs alone and, with regard to its power, is small in comparison to the drives of the x-axis and the z-axis, it is not taken into account in the rating of the rectifier unit and the braking resistor.

11.9.3 Calculating the z-axis as the lifting drive

1. Drive data

◆ Mass to be transported	$m = 200 \text{ kg}$
◆ Pinion diameter	$D = 0.1 \text{ m}$
◆ Max. speed	$v_{\max} = 1.5 \text{ m/s}$
◆ Max. acceleration and deceleration	$a_{\max} = 2.5 \text{ m/s}^2$
◆ Lifting height	$h = 1.35 \text{ m}$
◆ Cycle time	$T = 7 \text{ s}$
◆ Mech. efficiency	$\eta_{\text{mech}} = 0.9$
◆ Mech. accuracy	$\Delta s_{\text{mech}} = \pm 0.1 \text{ mm}$
◆ Overall accuracy required	$\Delta s_{\text{tot}} = \pm 0.2 \text{ mm}$

2. Travel curve

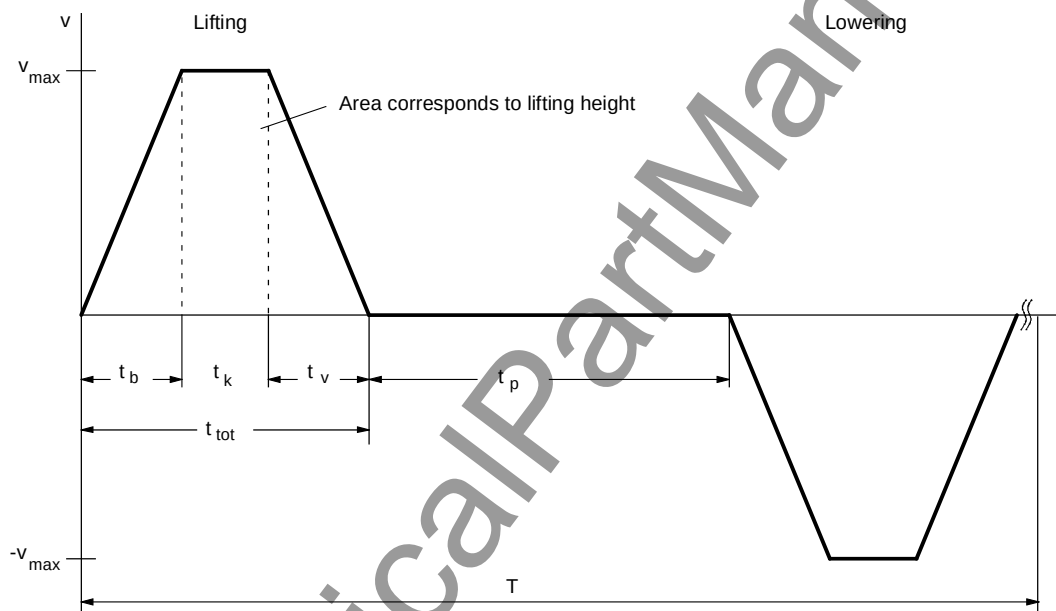


Fig. 11-18 Travel curve for lifting and lowering

The travel curve for lifting and lowering is symmetrical. Since the lifting torque and the lowering torque are different, however, the whole travel curve has to be considered.

- ◆ The following is obtained for the missing values of the travel curve:

$$t_b = t_v = \frac{v_{\max}}{a_{\max}} = \frac{1.5}{2.5} = 0.6 \text{ s}$$

$$t_k = \frac{h - v_{\max} \cdot \frac{t_b}{2} - v_{\max} \cdot \frac{t_v}{2}}{v_{\max}} = \frac{1.35 - 1.5 \cdot \frac{0.6}{2} - 1.5 \cdot \frac{0.6}{2}}{1.5} = 0.3 \text{ s}$$

$$t_{\text{tot}} = t_b + t_k + t_v = 0.6 + 0.3 + 0.6 = 1.5 \text{ s}$$

$$t_p = \frac{T}{2} - t_{\text{tot}} = 3.5 - 1.5 = 2 \text{ s}$$

3. Max. speed under load, max. torque under load, selection of gear unit

- ◆ Max. speed under load at the pinion

$$n_{\text{load max}} = \frac{v_{\text{max}} \cdot 60}{\pi \cdot D} = \frac{15 \cdot 60}{\pi \cdot 0.1} = 286.5 \text{ rpm}$$

Here, a gear transmission ratio of $i=10$ is selected. A synchronous servomotor with a rated speed of 3000 rpm can therefore be used.

$$n_{\text{Mot max}} = i \cdot n_{\text{load max}} = 10 \cdot 286.5 = 2865 \text{ rpm}$$

- ◆ Lifting torque

$$M_H = m \cdot g \cdot \frac{D}{2} = 200 \cdot 9.81 \cdot \frac{0.1}{2} = 98.1 \text{ Nm}$$

- ◆ Acceleration and deceleration torque for the load

$$\alpha_{\text{load}} = a_{\text{max}} \cdot \frac{2}{D} = 2.5 \cdot \frac{2}{0.1} = 50 \text{ s}^{-2}$$

$$J_{\text{load}} = m \cdot \left(\frac{D}{2}\right)^2 = 200 \cdot \left(\frac{0.1}{2}\right)^2 = 0.5 \text{ kgm}^2$$

$$M_{b,v \text{ load}} = J_{\text{load}} \cdot \alpha_{\text{load}} = 0.5 \cdot 50 = 25 \text{ Nm}$$

- ◆ Max. torque on the output side of the gear unit

$$M_{\text{load max}} = (M_{b \text{ load}} + M_H) \cdot \frac{1}{\eta_{\text{mech}}} = (25 + 98.1) \cdot \frac{1}{0.9} = 136.8 \text{ Nm}$$

A SPG140-M1 planetary gear unit for mounting on 1FT6 motors is selected with

$$M_{\text{max}} = 400 \text{ Nm when } i=10$$

$$J_G^* = 0.001 \text{ kgm}^2 \quad \text{Moment of inertia referred to the motor}$$

$$\eta_G = 0.95 \quad \text{Gear unit efficiency}$$

$$\alpha_G = 3' \quad \text{Torsional play}$$

- ◆ Acceleration and deceleration torque for the gear unit

$$M_{b,v G} = J_G^* \cdot \alpha_{\text{Load}} \cdot i = 0.001 \cdot 50 \cdot 10 = 0.5 \text{ Nm}$$

- ◆ Positioning accuracy

$$\Delta S_{\text{Gear}} = \frac{D \cdot \pi}{360^\circ} \cdot \frac{\alpha_G}{60} = \frac{0.1 \cdot \pi}{360} \cdot \frac{3}{60} = 0.0436 \text{ mm}$$

$$\text{i.e. } \pm 0.0218 \text{ mm}$$

$$\Delta S_{\text{Encoder}} = \frac{D \cdot \pi}{i \cdot z} = \frac{0.1 \cdot \pi}{10 \cdot 4096} = \pm 0.0077 \text{ mm, with an 8-pole resolver}$$

$$\Delta S_{\text{tot}} = \Delta S_{\text{mech}} + \Delta S_{\text{Gear}} + \Delta S_{\text{Encoder}}$$

$$= 0.1 + 0.0218 + 0.0077 = 0.1295 < 0.2 \text{ mm}$$

The accuracy requirement is thus satisfied.

4. Selection of motor

Selection in relation to the dynamic limit curve

- ◆ The max. motor torque here occurs during acceleration upwards since the deceleration is equal to the acceleration and the drive, during lifting, also has to overcome the levels of efficiency.

$$M_{\text{Mot max}} = M_{\text{b Mot}} + M_{\text{b G}} + (M_{\text{b Load}} + M_{\text{H}}) \cdot \frac{1}{i \cdot \eta_{\text{mech}} \cdot \eta_{\text{G}}}$$

$$= M_{\text{b Mot}} + 0.5 + (25 + 98.1) \cdot \frac{1}{10 \cdot 0.9 \cdot 0.95} = M_{\text{b Mot}} + 14.9 \text{ Nm}$$

$$\text{with } M_{\text{b Mot}} = J_{\text{Mot}} \cdot \alpha_{\text{Load}} \cdot i = J_{\text{Mot}} \cdot 50 \cdot 10 = J_{\text{Mot}} \cdot 500 \text{ s}^{-2}$$

The first 1FT6 motor with $n_n=3000$ rpm, which satisfies the condition or matches the dynamic limit curve, is the 1FT6082-8AF7 with $P_n=3.2$ kW, $M_n=10.3$ Nm, $M_{\text{max perm}}=42$ Nm, $J_{\text{Mot}}=0.00335$ kgm² (with brake), $k_{Tn100}=1.18$ Nm/A, $\eta_{\text{Mot}}=0.89$, $M_0=13$ Nm

- ◆ The acceleration and deceleration torque for the motor rotor is thus

$$M_{\text{b,v Mot}} = 0.00335 \cdot 500 = 168 \text{ Nm}$$

- ◆ The max. motor torque is equal to the motor torque during acceleration:

$$M_{\text{Mot max}} = M_{\text{Mot b up}} = 168 + 14.9 = 16.58 \text{ Nm}$$

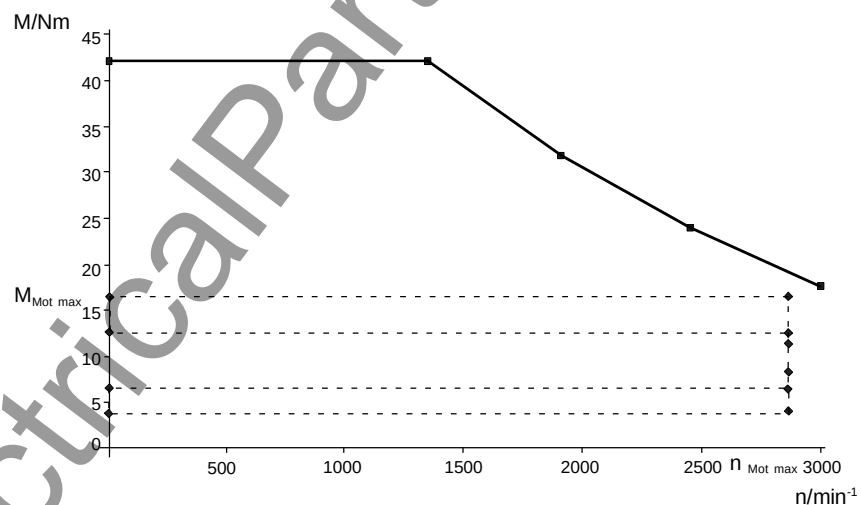


Fig. 11-19

Fig. 11-19 Dynamic limit curve for the 1FT6082-8AF7 with the points of the load cycle

In order to check the thermal limits, the effective motor torque is calculated. For this purpose, all other motor torques within the travel curve have to be calculated, in addition to the motor torque during acceleration.

- ♦ Lifting of the load, motor torque during constant travel

$$M_{\text{Mot k up}} = M_H \cdot \frac{1}{i \cdot \eta_{\text{mech}} \cdot \eta_G} = 98.1 \cdot \frac{1}{10 \cdot 0.9 \cdot 0.95} = 1147 \text{ Nm}$$

- ♦ Lowering of the load, motor torque during constant travel

$$M_{\text{Mot k down}} = M_H \cdot \frac{\eta_{\text{mech}} \cdot \eta_G}{i} = 98.1 \cdot \frac{0.9 \cdot 0.95}{10} = 8.39 \text{ Nm}$$

- ♦ Lifting of the load, motor torque during deceleration

$$M_{\text{Mot up}} = -M_{\text{v Mot}} - M_{\text{v G}} + (-M_{\text{v Load}} + M_H) \cdot \frac{1}{i \cdot (\eta_{\text{mech}} \cdot \eta_G) \cdot \text{Sign}(-M_{\text{v Load}} + M_H)}$$

$$= -168 - 0.5 + (-25 + 98.1) \cdot \frac{1}{10 \cdot 0.9 \cdot 0.95} = 6.37 \text{ Nm}$$

Lowering of the load, motor torque during acceleration

$$M_{\text{Mot b down}} = -M_{\text{b Mot}} - M_{\text{b G}} + (-M_{\text{b Load}} + M_H) \cdot \frac{(\eta_{\text{mech}} \cdot \eta_G) \cdot \text{Sign}(-M_{\text{b Load}} + M_H)}{i}$$

$$= -168 - 0.5 + (-25 + 98.1) \cdot \frac{0.9 \cdot 0.95}{10} = 4.08 \text{ Nm}$$

- ♦ Lowering of the load, motor torque during deceleration

$$M_{\text{Mot v down}} = M_{\text{v Mot}} + M_{\text{v G}} + (M_{\text{v Load}} + M_H) \cdot \frac{\eta_{\text{mech}} \cdot \eta_G}{i}$$

$$= 168 + 0.5 + (25 + 98.1) \cdot \frac{0.9 \cdot 0.95}{10} = 12.7 \text{ Nm}$$

The motor curve can be determined with the help of the values calculate for the motor torque.

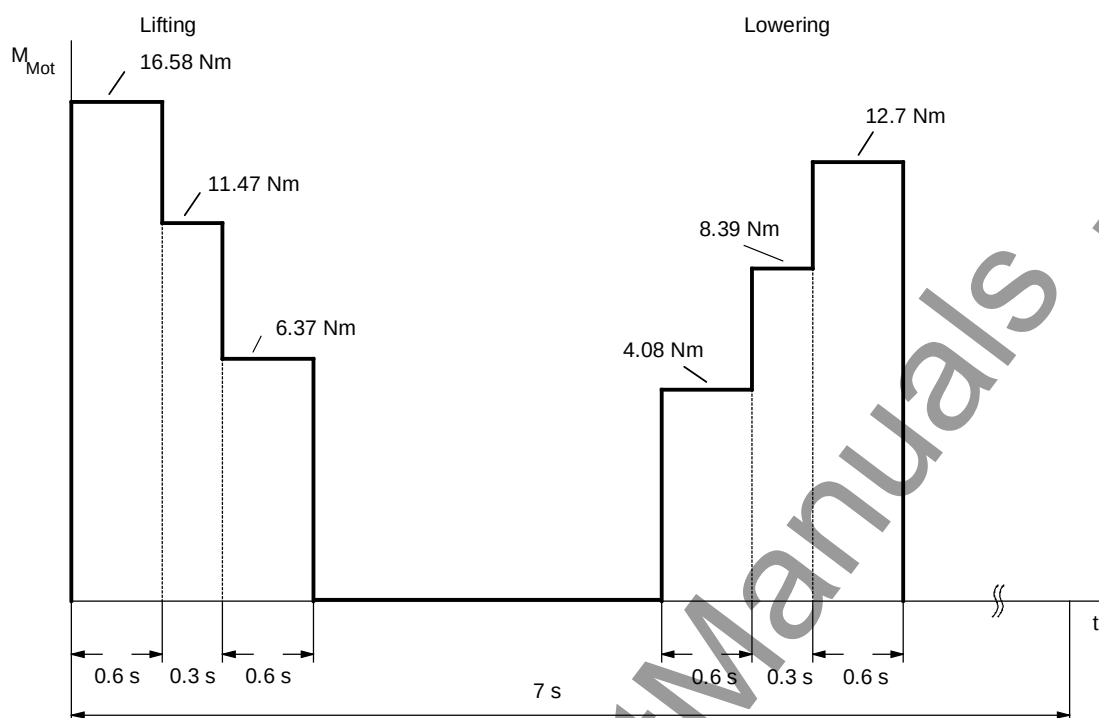


Fig. 11-20 Torque characteristic for lifting and lowering

- ◆ The effective motor torque is obtained from the torque characteristic as follows:

$$M_{\text{eff}} = \sqrt{\frac{\sum M_{\text{Mot } i}^2 \cdot \Delta t_i}{T}}$$

$$= \sqrt{\frac{16.58^2 \cdot 0.6 + 11.47^2 \cdot 0.3 + 6.37^2 \cdot 0.6 + 4.08^2 \cdot 0.6 + 8.39^2 \cdot 0.3 + 12.7^2 \cdot 0.6}{7}}$$

$$= 7.14 \text{ Nm}$$

- ◆ The speed-proportional travel curve is used to obtain the mean motor speed as follows:

$$n_{\text{mean}} = \frac{\sum \frac{|n_A + n_E|}{2} \cdot \Delta t_i}{T}$$

$$= \frac{(\frac{2865}{2} \cdot 0.6 + 2865 \cdot 0.3 + \frac{2865}{2} \cdot 0.6) \cdot 2}{7} = 736.7 \text{ rpm}$$

(due to the symmetry of the travel curve, the component for lifting is multiplied by 2)

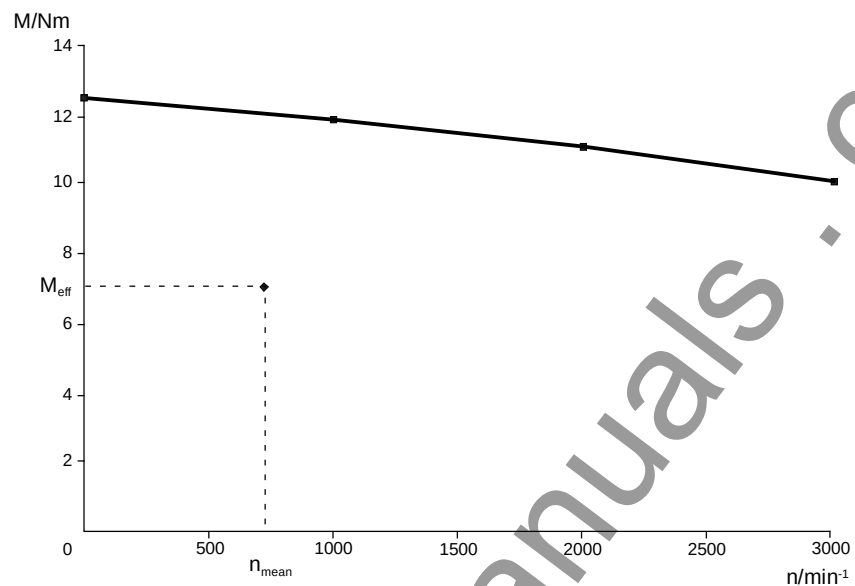


Fig. 11-21 S1 curve for the 1FT6082-8AF7

The calculated effective motor torque at n_{mean} is below the S1 curve. The motor is therefore suitable.

5. Selection of the inverter

The inverter is selected according to the maximum motor current and the mean motor current.

- ◆ Maximum motor current (the saturation influence here can be ignored)

$$I_{\text{Mot max}} \approx \frac{M_{\text{Mot max}}}{k_{Tn100}} = \frac{16.57}{118} = 14 \text{ A}$$

- ◆ Mean motor current, obtained from the magnitude of the torque characteristic

$$I_{\text{Mot mean}} \approx \frac{\sum |M_{\text{Mot } i}| \cdot \Delta t_i}{k_{Tn100} \cdot T}$$

$$= \frac{16.58 \cdot 0.6 + 11.47 \cdot 0.3 + 6.37 \cdot 0.6 + 4.08 \cdot 0.6 + 8.39 \cdot 0.3 + 12.7 \cdot 0.6}{118 \cdot 7} = 3.6 \text{ A}$$

- ◆ A 6SE7021-0TP50 Compact PLUS inverter is necessary with $I_{UN}=10.2 \text{ A}$. Since the acceleration and deceleration times are $> 0.25 \text{ s}$, only 1.6 times the rated current can be utilized. Thus

$$I_{\text{Mot max}} = 14 \text{ A} < 1.6 \cdot I_{UN} = 16 \text{ A}$$

$$I_{\text{Mot mean}} = 3.6 \text{ A} < I_{UN} = 10.2 \text{ A}$$

6. Determination of the DC link currents

The maximum DC link current occurring during motor operation and the mean DC link current for the inverter have to be determined for later rating of the rectifier unit. To do this, all power outputs of the motor within the travel curve first have to be calculated.

- ◆ Lifting of the load, max. power output of motor during acceleration

$$P_{\text{Mot b up max}} = \frac{M_{\text{Mot b up}} \cdot \eta_{\text{Mot max}}}{9550} = \frac{16.58 \cdot 2865}{9550} = 4.97 \text{ kW}$$

- ◆ Lifting of the load, power output of motor during constant travel

$$P_{\text{Mot k up}} = \frac{M_{\text{Mot k up}} \cdot \eta_{\text{Mot max}}}{9550} = \frac{11.47 \cdot 2865}{9550} = 3.44 \text{ kW}$$

- ◆ Lifting of the load, max. power output of motor during deceleration

$$P_{\text{Mot v up max}} = \frac{M_{\text{Mot v up}} \cdot \eta_{\text{Mot max}}}{9550} = \frac{6.37 \cdot 2865}{9550} = 1.91 \text{ kW}$$

- ◆ Lowering of the load, max. power output of motor during acceleration

$$P_{\text{Mot b down max}} = \frac{M_{\text{Mot b down}} \cdot \eta_{\text{Mot max}}}{9550} = \frac{4.08 \cdot (-2865)}{9550} = -1.22 \text{ kW}$$

- ◆ Lowering of the load, power output of motor during constant travel

$$P_{\text{Mot k down}} = \frac{M_{\text{Mot k down}} \cdot \eta_{\text{Mot max}}}{9550} = \frac{8.39 \cdot (-2865)}{9550} = -2.52 \text{ kW}$$

- ◆ Lowering of the load, max. power output of motor during deceleration

$$P_{\text{Mot v down max}} = \frac{M_{\text{Mot v down}} \cdot \eta_{\text{Mot max}}}{9550} = \frac{12.7 \cdot (-2865)}{9550} = -3.81 \text{ kW}$$

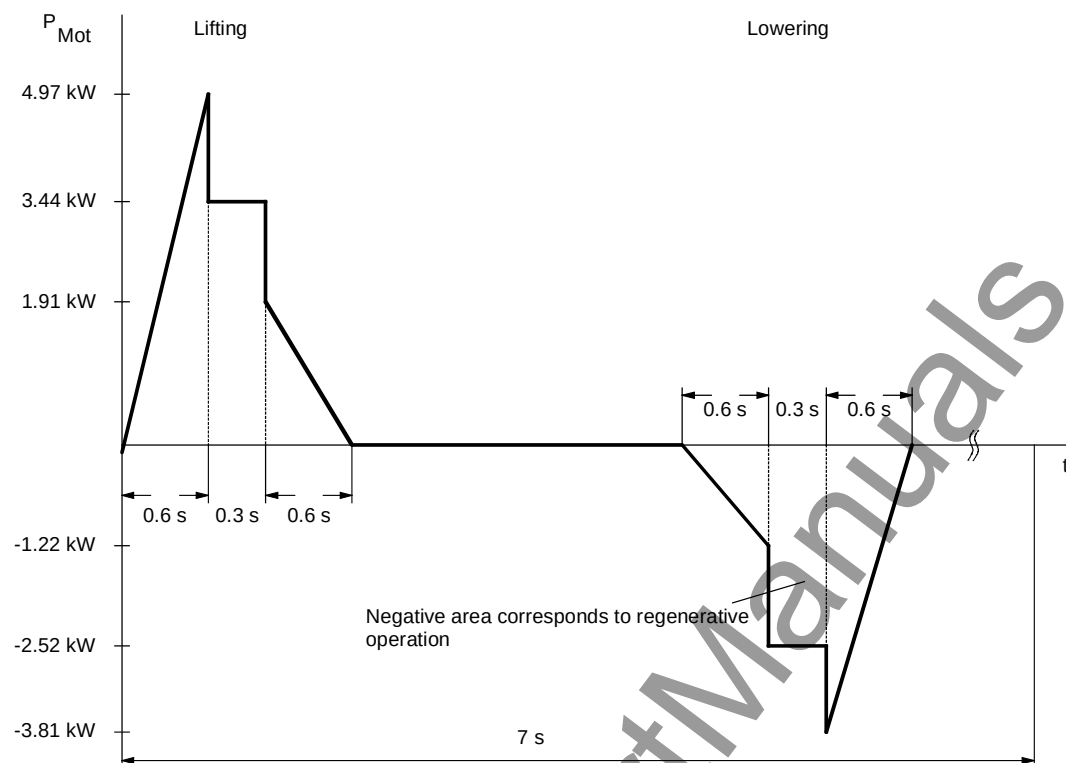


Fig. 11-22 Curve of motor power output for lifting and lowering

- ◆ The maximum DC link current during motor operation during acceleration upwards is as follows:

$$I_{\text{Link Inv max}} = \frac{P_{\text{Mot max}}}{\eta_{\text{Mot}} \cdot \eta_{\text{Inv}} \cdot 135 \cdot U_{\text{Line}}}$$

$$= \frac{4970}{0.89 \cdot 0.98 \cdot 135 \cdot 400} = 10.55 \text{ A}$$

- ◆ The mean power output of the motor during motor operation is calculated from the positive characteristic of the motor power output as follows:

$$P_{\text{Mot mean}} = \frac{\sum \frac{P_{\text{Mot A}} + P_{\text{Mot E}}}{2} \cdot \Delta t_i}{T}$$

$$= \frac{\frac{1}{2} \cdot 4.97 \cdot 0.6 + 3.44 \cdot 0.3 + \frac{1}{2} \cdot 1.91 \cdot 0.6}{7} = 0.442 \text{ kW}$$

- ◆ The mean DC link current is therefore:

$$I_{\text{Link mean}} = \frac{P_{\text{Mot mean}}}{\eta_{\text{Mot}} \cdot \eta_{\text{Inv}} \cdot 135 \cdot U_{\text{Line}}}$$

$$= \frac{442}{0.89 \cdot 0.98 \cdot 135 \cdot 400} = 0.938 \text{ A}$$

7. Determination of braking power

The maximum braking power and the mean braking power have to be determined for later rating of the braking resistors. The maximum motor power output during braking has already been calculated in 6.

- ◆ The maximum braking power is thus:

$$P_{br \max} = P_{Mot \text{ v down max}} \cdot \eta_{Mot} \cdot \eta_{Inv} = -3.81 \cdot 0.89 \cdot 0.98 = -3.32 \text{ kW}$$

- ◆ The mean braking power is obtained from the negative characteristic of the motor power output as follows:

$$P_{br \text{ mean}} = \frac{\sum \frac{P_{Mot \text{ v A}} + P_{Mot \text{ v E}}}{2} \cdot \Delta t_i}{T} \cdot \eta_{Mot} \cdot \eta_{Inv}$$

$$= \frac{\frac{1}{2} \cdot (-122) \cdot 0.6 + (-2.52) \cdot 0.3 + \frac{1}{2} \cdot (-3.81) \cdot 0.6}{7} \cdot 0.89 \cdot 0.98 = -0.28 \text{ kW}$$

11.9.4 Selection of the rectifier unit

Now that the drives of the x, y and z axes have been calculated, the rectifier unit can be selected. Here, it is assumed that the drives of the x and z axes can operate simultaneously.

- ◆ The maximum DC link currents of the two inverters during motor operation are therefore added together.

$$I_{Link \text{ Rect max}} = \sum I_{Link \text{ Inv max}} = 14.56 \text{ A} + 10.55 \text{ A} = 25.11 \text{ A}$$

- ◆ In order to determine the mean value of the DC link current, the mean values of the two inverters are added together.

$$I_{Link \text{ Rect mean}} = \sum I_{Link \text{ Inv mean}} = 0.95 \text{ A} + 0.938 \text{ A} = 1.89 \text{ A}$$

- ◆ The 15 kW rectifier unit, 6SE7024-1EP85-0AA0, with $I_{ZK \text{ n}} = 41 \text{ A}$ is sufficient.

$$I_{Link \text{ Rect max}} = 25.11 \text{ A} < 1.6 \cdot I_{Link \text{ n}} = 65.6 \text{ A}$$

$$I_{Link \text{ Rect mean}} = 1.89 \text{ A} < I_{Link \text{ n}} = 41 \text{ A}$$

11.9.5 Selection of the braking resistor

The braking resistor is connected to the chopper of the rectifier unit. During rating, it is assumed that the drives of the x and z axes can brake simultaneously.

- ◆ The maximum braking power levels of the two inverters are therefore added together.

$$P_{br \max} = \sum P_{br \text{ Inv}} = -4.08 \text{ kW} - 3.32 \text{ kW} = -7.4 \text{ kW}$$

- ◆ For the mean braking power, the individual mean values are also added together.

$$P_{br \text{ mean}} = \sum P_{br \text{ Inv mean}} = -0.146 \text{ kW} - 0.28 \text{ kW} = -0.426 \text{ kW}$$

- ◆ A 6SE7018-0ES87-2DC0 braking resistor of 80 Ω with $P_{20} = 5 \text{ kW}$ is necessary.

$$P_{br \max} = 7.4 \text{ kW} < 15 \cdot P_{20} = 7.5 \text{ kW}$$

$$P_{br \text{ mean}} = 0.426 \text{ kW} < P_{20} / 4.5 = 1.11 \text{ kW}$$

11.10 Power Extension PIN F02 (from firmware version 2.20 and higher)

Enabling the Power Extension PIN F02

Halving of the pulse frequency for operating power sections over 250 kW can be automatically selected via parameter P357 = 1 (pulse frequency ratio = 2:1).

If the F02 option has not been enabled (n978.2 = 0), it is not possible to select power sections > 250 kW. This can, however, be carried out by means of a subsequent enable.

**MFLB subsequent enable F02:
6SW1700-5AD00-2XX0**

To enable:

When ordering, it is imperative to specify the FID modules (product identification number, 2x4 digit number). The FID can be read out at parameters U976.1 and U976.2.

Save the F02 PIN in parameters U977.3 and U977.4.

You can check whether option F02 is present via the display parameter n978.2:

n978.2 = 1 ==> Option F02 is enabled

n978.2 = 0 ==> Option F02 is blocked

By halving the pulse frequency on units from 75 kW to 250 kW it is possible to expand the output by changing the derating (see Chapter "Technical Data").

Example

Higher output by a lower pulse frequency!

If the pulse frequency of Performance 2 units higher than 75 kW is set to 2.5 kHz, the output current of MASTERDRIVES VC units is then available to the user!

6SE7031-8EF70

75 kW unit rating Performance 2 with pulse frequency 5 kHz

Rated output current = 155 A

6SE7031-8EF70-Z-F02

75 kW Performance 2 with pulse frequency 2.5 kHz

Rated output current (MASTERDRIVES VC 90 kW unit rating = 186 A)

Comparison:

6SE7032-1EG50

90 kW unit rating standard MASTERDRIVES MC

Rated output current = 175 A

Overview

Unit rating [kW]	Standard	Basic unit with F02 as additional option									
Unit rating [kW] with F02 as add. option	Rated output current I _{Un} [A]	Short-time/overload current I _{max} [A]	Rated DC link current I _{ZKn} [A]	Line current (only for converters) [A]	Order No. Converter	Order No. Inverter	Total power loss Converter [kW]	Total power loss Inverter [kW]	Weight kg		
75	90	186	254	221	205	6SE7031-8EF70	6SE7031-8TF70	2.17	1.7	75	
90	110	210	287	250	231	6SE7032-1EG70	6SE7032-1TG70	2.68	2.18	160	
110	132	260	355	309	286	6SE7032-6EG70	6SE7032-6TG70	3.40	2.75	160	
132	160	315	430	375	346	6SE7033-2EG70	6SE7033-2TG70	4.30	3.47	180	
160	200	370	503	440	407	6SE7033-7EG70	6SE7033-7TG70	5.05	4.05	180	
200	250	510	694	607	—	—	6SE7035-1TJ70	—	5.8	350	
200	250	510	694	607	561	6SE7035-1EK70	—	7.10	—	400	
250	315	590	802	702	—	—	6SE7036-0TJ70	—	6.6	350	
250	315	590	802	702	649	6SE7036-0EK70	—	8.20	—	400	
	*400	690	938	821	—	—	6SE7037-0TJ70	—	8.8	350	
	*400	690	938	821	759	6SE7037-0EK70	—	10.20	—	400	
	*500	860	1170	1023	—	—	6SE7038-6TK70	—	11.9	520	
	*630	1100	1496	1310	—	—	6SE7041-1TK70	—	13.4	520	
	*710	1300	1768	1547	—	—	6SE7041-3TL70	—	14.5	625	
* Performance extension: Only operable with P357 = 1 (max. 2.7 kHz) (F02 as add. option is included)											

Table 11-2 Overview of units which can be operated with F02 as an additional option