

Drive Based Synchronous Operating manual

en-US
02/2021
ID 443046.02

Table of contents

1	Foreword	5
2	User information	6
2.1	Note on gender	6
2.2	Storage and transfer	6
2.3	Described product.....	6
2.4	Timeliness	6
2.5	Original language	6
2.6	Limitation of liability	6
2.7	Formatting conventions.....	7
2.7.1	Use of symbols.....	7
2.7.2	Markup of text elements	8
2.7.3	Mathematics and formulas.....	8
2.8	Trademarks	9
3	What you should know before commissioning	10
3.1	Structure of the program interface.....	10
3.1.1	Individualized workspace.....	12
3.1.2	Navigation using sensitive circuit diagrams	12
3.2	Meaning of parameters	13
3.2.1	Parameter groups	13
3.2.2	Parameter types and data types.....	14
3.2.3	Parameter types	15
3.2.4	Parameter structure	15
3.2.5	Parameter visibility	16
3.3	Signal transmission and fieldbus mapping.....	17
3.4	Power-loss protected storage.....	18
4	Commissioning	19
4.1	Initiating the project	20
4.1.1	Projecting the drive controller and axis	20
4.1.2	Configuring safety technology	21
4.1.3	Creating other modules and drive controllers.....	21
4.1.4	Specifying a module.....	22
4.1.5	Specifying the project	22
4.2	Mapping the mechanical axis model	23
4.2.1	Parameterizing a STOBER motor	23
4.2.2	Parameterizing the axis model	23
4.3	Referencing absolute position	27
4.3.1	Defining the referencing method	27
4.3.2	Parameterizing the reference switch.....	27

4.3.3	Setting the reference	28
4.3.4	Parameterizing the limit switch	28
4.4	Parameterizing Drive Based device control	29
4.4.1	Parameterizing transition conditions.....	29
4.5	Configuring synchronous operation.....	30
4.5.1	Model 1: Virtual master or master encoder, multiple slave axes	31
4.5.2	Model 2: Master encoder, individual slave axis.....	38
4.5.3	Model 3: Master axis, one or more slave axes	42
4.6	Configuring the IGB motion bus network	45
4.6.1	Projecting the IGB motion bus network	45
4.6.2	Assigning master and slave axes	46
4.6.3	Monitoring synchronous operation	47
4.7	Transmitting a configuration to drive controllers.....	48
4.7.1	Transmitting and saving the configuration	48
4.8	Testing the configuration	50
4.8.1	Testing jog mode	50
4.8.2	Testing the motion commands	52
4.8.3	Testing synchronous operation	53
4.9	Parameterizing the Drive Based synchronous application.....	58
4.9.1	General motion variables and signal sources	59
4.9.2	Parameterizing motion block operating mode	62
4.9.3	Parameterizing command operating mode	80
5	More on Drive Based synchronous?.....	85
5.1	Drive Based synchronous – Concept.....	85
5.1.1	Operating modes	86
5.1.2	Jog mode.....	86
5.1.3	Control panels.....	87
5.1.4	Data sources	87
5.1.5	Additional functions.....	87
5.1.6	Motion core	87
5.2	IGB motion bus network	88
5.3	Mechanical axis model.....	89
5.3.1	Rotational drives	89
5.3.2	Translational drives.....	90
5.4	Limit switches	92
5.4.1	Real axes	92
5.4.2	Virtual master	97
5.4.3	Special cases	98
5.5	Referencing.....	99
5.5.1	Referencing methods.....	100
5.5.2	Reference position.....	125
5.6	Drive Based device control	126

5.6.1	Drive Based device state machine	126
5.6.2	States, transitions and conditions.....	127
5.7	Motion commands.....	131
5.8	Electronic nameplate	132
5.9	Operating modes – Block diagrams	133
5.9.1	Motion block.....	133
5.9.2	Command	137
5.10	Cycle times	139
6	Appendix.....	140
6.1	Standard mapping of PROFINET and Drive Based synchronous	140
6.1.1	SC6, SI6: RxPZD	140
6.1.2	SC6, SI6: TxPZD	141
6.1.3	SD6: RxPZD.....	142
6.1.4	SD6: TxPZD	142
6.2	Detailed information.....	143
6.3	Abbreviations.....	144
7	Contact.....	145
7.1	Consultation, service and address	145
7.2	Your opinion is important to us	145
7.3	Close to customers around the world.....	146
	Glossary	149

1 Foreword

The DriveControlSuite software of the 6th STOBER drive controller generation offers convenient functions for efficient project configuration and commissioning for synchronous operation of multi-axis systems or single-axis applications.

The Drive Based synchronous application package contained in the software provides universal solutions for drive-based motion control of SD6, SC6 and SI6 series single-axis and double-axis controllers using position, velocity and torque/force control types. The standard commands associated with these control types are categorized into two operating modes: command and motion block.

Additional functions for monitoring process variables such as positions or velocities offer added convenience in monitoring.

This documentation describes the general functions of the Drive Based synchronous application and guides you step by step through the setup and project configuration of your drive project in the individual operating modes.

2 User information

This documentation supports you in the setup and project configuration of your drive system with the Drive Based synchronous application in conjunction with Drive Based device control.

Technical knowledge

In order to be able to commission one or more drive controllers—in combination with a controller, where applicable—using Drive Based synchronous, you should have experience in project configuration for drive-based motion profiles.

Technical requirements

Before you begin operating your drive system, you need to have wired the drive controllers involved and initially check that they are functioning correctly. To do so, follow the instructions in the respective commissioning instructions.

2.1 Note on gender

Note on gender

For reasons of improved readability, gendered differentiation is not used. In the sense of equal treatment, the corresponding terms apply equally to all genders. The shortened form does not imply any value judgment, but is purely for editorial reasons.

2.2 Storage and transfer

As this documentation contains important information for handling the product safely and efficiently, it must be stored in the immediate vicinity of the product until product disposal and be accessible to qualified personnel at all times.

Also pass on this documentation if the product is transferred or sold to a third party.

2.3 Described product

This documentation is binding for:

SD6, SC6 or SI6 series drive controllers in conjunction with the DriveControlSuite software (DS6) in V 6.5-A or higher and associated firmware in V 6.5-A or higher.

2.4 Timeliness

Check whether this document is the latest version of the documentation. We make the latest document versions for our products available for download on our website:

<http://www.stoeber.de/en/downloads/>.

2.5 Original language

The original language of this documentation is German; all other language versions are derived from the original language.

2.6 Limitation of liability

This documentation was created taking into account the applicable standards and regulations as well as the current state of technology.

No warranty or liability claims for damage shall result from failure to comply with the documentation or from use that deviates from the intended use of the product. This is especially true for damage caused by individual technical modifications to the product or the project configuration and operation of the product by unqualified personnel.

2.7 Formatting conventions

Orientation guides in the form of signal words, symbols and special text markups are used to emphasize specific information so that you are able identify it in this documentation quickly.

2.7.1 Use of symbols

Safety instructions are identified with the following symbols. They indicate special risks when handling the product and are accompanied by relevant signal words that express the extent of the risk. Furthermore, useful tips and recommendations for efficient, error-free operation are specially highlighted.

ATTENTION!

Attention

This indicates that damage to property may occur

- if the stated precautionary measures are not taken.

⚠ CAUTION!

Caution

This word with a warning triangle indicates that minor personal injury may occur

- if the stated precautionary measures are not taken.

⚠ WARNING!

Warning

This word with a warning triangle means there may be a considerable risk of fatal injury

- if the stated precautionary measures are not taken.

⚠ DANGER!

Danger

This word with a warning triangle indicates that there is a considerable risk of fatal injury

- if the stated precautionary measures are not taken.

Information

Information indicates important information about the product or serves to emphasize a section in the documentation that deserves special attention from the reader.

2.7.2 Markup of text elements

Certain elements of the continuous text are distinguished as follows.

Important information	Words or expressions with a special meaning
Interpolated position mode	Optional: File or product name or other name
<u>Detailed information</u>	Internal cross-reference
http://www.samplelink.com	External cross-reference

Software and other displays

The following formatting is used to identify the various information content of elements referenced by the software interface or a drive controller display, as well as any user entries.

Main menu Settings	Window names, dialog box names, page names or buttons, combined proper nouns, functions referenced by the interface
Select Referencing method A	Predefined entry
Save your <own IP address>	User-defined entry
EVENT 52: COMMUNICATION	Displays (status, messages, warnings, faults) for status information referenced by the interface

Keyboard shortcuts and command sequences or paths are represented as follows.

[CTRL], [CTRL] + [S]	Key, shortcut
Table > Insert table	Navigation to menus/submenus (path specification)

2.7.3 Mathematics and formulas

The following signs are used to represent mathematical relationships and formulas.

-	Subtraction
+	Addition
×	Multiplication
÷	Division
	Amount

2.8 Trademarks

The following names used in connection with the device, its optional equipment and its accessories are trademarks or registered trademarks of other companies:

CANopen [®] , CiA [®]	CANopen [®] and CiA [®] are registered European Union trademarks of CAN in AUTOMATION e.V., Nuremberg, Germany.
EnDat [®]	EnDat [®] and the EnDat [®] logo are registered trademarks of Dr. Johannes Heidenhain GmbH, Traunreut, Germany.
EtherCAT [®] , Safety over EtherCAT [®] , TwinCAT [®]	EtherCAT [®] , Safety over EtherCAT [®] and TwinCAT [®] are registered trademarks of patented technologies licensed by Beckhoff Automation GmbH, Verl, Germany.
HIPERFACE [®]	HIPERFACE [®] and the HIPERFACE DSL [®] logo are registered trademarks of SICK STEGMANN GmbH, Donaueschingen, Germany.
PLCopen [®]	PLCopen [®] is a registered trademark of the PLCopen Organisation, Gorinchem, Netherlands.
PROFIBUS [®] , PROFINET [®]	The PROFIBUS and the PROFINET logo are registered trademarks of PROFIBUS Nutzerorganisation e.V., Karlsruhe, Germany.

All other trademarks not listed here are the property of their respective owners.

Products that are registered as trademarks are not specially indicated in this documentation. Existing property rights (patents, trademarks, protection of utility models) are to be observed.

3 What you should know before commissioning

The following chapters provide a quick introduction to the structure of the program interface and accompanying window designations as well as relevant information about parameters and generally saving your project configuration.

3.1 Structure of the program interface

The DriveControlSuite commissioning software (DS6) offers a graphic interface that you can use to project, parameterize and start up your axis model quickly and efficiently.

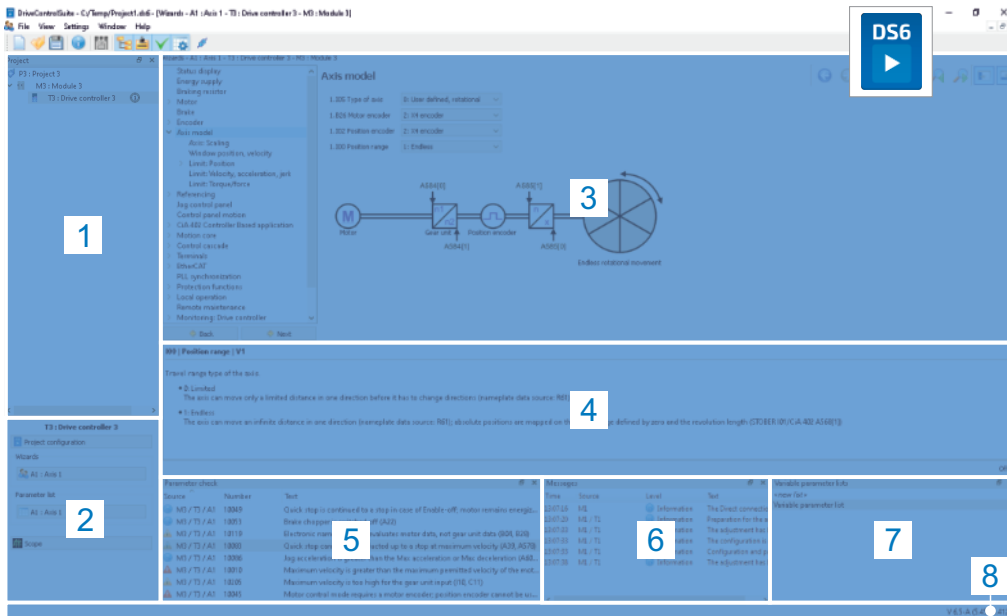


Fig. 1: DS6: Program interface

- 1 Project tree
- 2 Project menu
- 3 Workspace
- 4 Parameter description
- 5 Parameter check
- 6 Messages
- 7 Variable parameter lists
- 8 Status bar

Menus and toolbars

Using the main menus *File*, *View*, *Settings* and *Window*, you can open and save your projects, display and hide program windows, define the interface language and various access levels, or switch between different areas of the program interface.

1: Project tree

All of the drive controllers of your drive project are collected in the project tree under a project and organized into modules.

One project, one module and one drive controller are created in the project tree by default and can be configured individually; all additional elements must be added manually. If you would like to edit an element of the project tree, mark the element and switch over to the project menu.

2: Project menu

You can edit the elements of the project tree in the project menu. The project menu consists of the project, module and drive controller views.

Highlighting one of the project, module or drive controller elements in the project tree automatically activates the corresponding project menu view.

3: Workspace

The workspace is only active while configuring a drive controller (project tree: drive controller). Depending on the projected device control and application, the workspace allows quick access to the wizards and parameter list, the graphical programming as well as the scope function of the drive controller view in the project menu.

4: Parameter description

Each parameter on a wizard page of the workspace has a short description, with possible selection values where applicable. By clicking on a parameter, the associated information is displayed in the parameter description.

5: Parameter check

Generally, your parameter values, if calculable, are checked by the software against each other for plausibility. Abnormalities or discrepancies are output and weighted in this view.

6: Messages

The software logs the connection and communication state of the active drive components, incorrect inputs received on the system side and errors in the messages.

7: Variable parameter lists

You can use variable parameter lists to compile the parameters currently required for a quick overview in individual parameter lists.

8: Status bar

In the status bar, you can find the specification of the software version. When loading projects, it also offers additional information on the project file, the devices and the progress of loading or updating projects.

3.1.1 Individualized workspace

The project tree (1) and project menu (2) are connected and, like the parameter check and messages (5, 6), can also be docked at the left, right or bottom edge of the screen. This program window can also be displayed or hidden using the View menu.

The workspace (3) and parameter description (4) are also connected to each other and always positioned in the middle. Both areas can be minimized or maximized.

3.1.2 Navigation using sensitive circuit diagrams

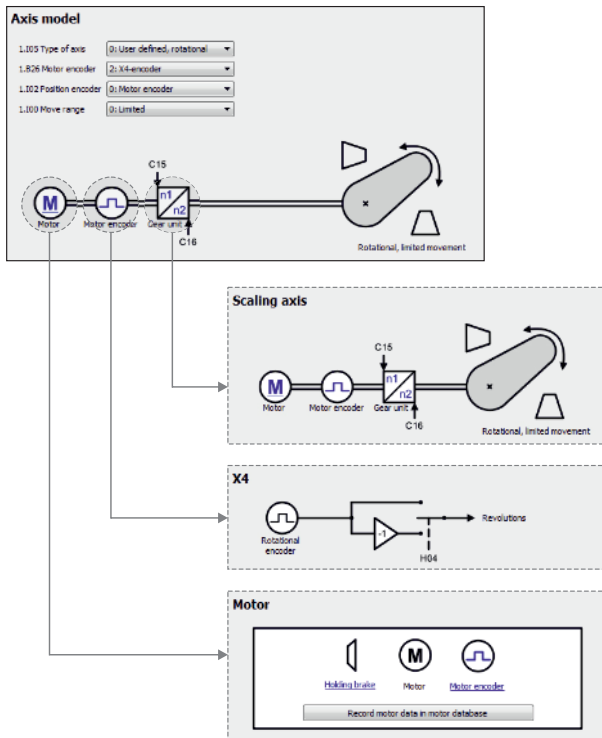


Fig. 2: DriveControlSuite: Navigation using text links and symbols

In order to illustrate graphically the processing sequence of actual and set values, the use of signals or certain drive component arrangements and to make configuring the accompanying parameters easier, they are displayed on the respective wizard pages of the workspace in the form of circuit diagrams.

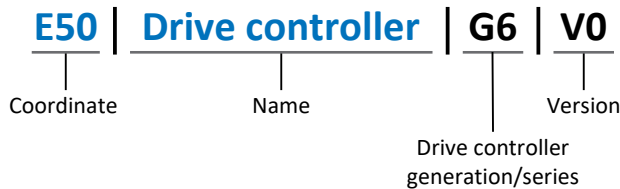
Blue text links or clickable icons indicate links within the program. These refer to the corresponding wizard pages and, as a result, allow you to reach additional helpful detail pages with just a click.

3.2 Meaning of parameters

You can adapt the function of the drive controller to your individual application using parameters. In addition, parameters visualize the current actual values (actual velocity, actual torque, etc.) and trigger general actions like Save values, Test phase, etc.

Interpretation of parameter identification

Parameter identification consists of the following elements, where short forms are also possible, i.e. only specifying a coordinate or the combination of coordinate and name.



3.2.1 Parameter groups

Parameters are assigned to individual groups by topic. The 6th generation of STOBER drive controllers differentiates between the following parameter groups.

Group	Topic
A	Drive controllers, communication, cycle times
B	Motor
C	Machine, velocity, torque/force, comparators
D	Set value
E	Display
F	Terminals, analog and digital inputs and outputs, brake
G	Technology – Part 1 (application-dependent)
H	Encoders
I	Motion (all motion settings)
J	Motion blocks
K	Control panel
L	Technology – Part 2 (application-dependent)
M	Profiles (application-dependent)
N	Additional functions (application-dependent; extended cam control unit)
P	Customer-specific parameters (programming)
Q	Customer-specific parameters, instance-dependent (programming)
R	Production data for the drive controller, motor, brakes, motor adapter, gear unit and geared motor
S	Safety (safety technology)
T	Scope
U	Protection functions
Z	Fault counter

Tab. 1: Parameter groups

3.2.2 Parameter types and data types

In addition to topic-based sorting in individual groups, all parameters belong to a certain data type and parameter type. The data type of a parameter is displayed in the parameter list, properties table. The connections between parameter types, data types and their value range can be found in the following table.

Type	Style	Length	Value range (decimal)
INT8	Integer or selection	1 byte (signed)	-128 – 127
INT16	Integer	2 bytes (1 word, signed)	-32768 – 32767
INT32	Integer or position	4 bytes (1 double word, signed)	-2147483648 – 2147483647
BOOL	Binary number	1 bit (internal: LSB in 1 byte)	0, 1
BYTE	Binary number	1 byte (unsigned)	0 – 255
WORD	Binary number	2 bytes (1 word, unsigned)	0 – 65535
DWORD	Binary number or parameter address	4 bytes (1 double word, unsigned)	0 – 4294967295
REAL32 (single type according to IEE754)	Floating-point number	4 bytes (1 double word, signed)	$-3.40282 \times 10^{38} - 3.40282 \times 10^{38}$
STR8	Text	8 characters	—
STR16	Text	16 characters	—
STR80	Text	80 characters	—

Tab. 2: Parameters – Data types, styles, possible values

Parameter types – Use

- Integer, floating-point number
For general computing processes
Example: Set and actual values
- Selection
Numeric value to which a direct meaning is assigned
Example: Sources for signals or set values
- Binary number
Bit-oriented parameter information that is collected in binary
Example: Control and status words
- Position
Integer combined with associated units and decimal places
Example: Actual and set values of positions
- Velocity, acceleration, deceleration, jerk
Floating-point number combined with the associated units and decimal places
Example: Actual and set values for velocity, acceleration, deceleration, jerk
- Parameter address
Corresponds to the storage location of another parameter
Example: Indirect read sources for analog and digital outputs and for fieldbus mapping
- Text
Outputs or messages

3.2.3 Parameter types

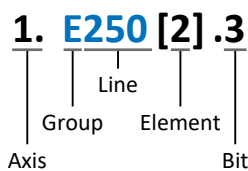
The following types of parameters are differentiated.

Parameter type	Description	Example
Simple parameters	Consist of one group and one line with a defined value.	A21 Brake resistor R: Value = 100 ohms
Array parameters	Consist of a group, a line and multiple sequential (listed) elements, which have the same properties but different values.	A10 Access level <ul style="list-style-type: none"> ▪ A10[0] access level: Value = Access level via operating unit ▪ A10[2] access level: Value = Access level via CANopen and EtherCAT ▪ A10[4] access level: Value = Access level via PROFINET
Structure parameters	Consist of a group, a line and multiple sequential (listed) elements, which can have different properties and different values.	A00 Save values <ul style="list-style-type: none"> ▪ A00[0] Start: Value = Start action ▪ A00[1] Progress: Value = Display action progress ▪ A00[2] Result: Value = Display action result

Tab. 3: Parameter types

3.2.4 Parameter structure

Every parameter has specific coordinates with the following pattern.



- **Axis**
The axis to which a parameter is assigned in the case of multiple axes (optional).
- **Group**
The thematic group to which a parameter belongs.
- **Line**
Distinguishes the parameters within a parameter group.
- **Element**
Elements of an array or structure parameter (optional).
- **Bit**
Selection of an individual bit for complete data addressing (optional).

3.2.5 Parameter visibility

The visibility of a parameter depends on the access level defined in the software, the dependency of other parameters, the selected application and the version of the associated firmware.

Access level

The access options for the individual software parameters are ranked hierarchically and divided into individual levels. This means that parameters can be hidden for a specific purpose and, relatedly, their configuration options can be locked starting from a specific level. The following levels are present:

- Level 0
Elementary parameters
- Level 1
Important parameters of an application
- Level 2
Important parameters for service with extensive diagnostic options
- Level 3
All parameters needed for commissioning and optimizing an application

Parameter A10 Access level controls general access to parameters:

- Over the SD6 drive controller display (A10[0])
- Over CANopen or EtherCAT (A10[2])
- Over PROFINET (A10[3])

Hiding functions

Hiding functions are used to hide parameters with regard to their logical relationships to other option modules or parameters.

For example, a drive controller can evaluate an encoder using terminal X120, provided that terminal module XI6 has been installed. The accompanying evaluation is activated using parameter H120. However, this parameter is visible only if terminal module XI6 was initially selected during the drive project configuration.

Applications

Applications generally differ in terms of functions and their activation. For this reason, different parameters are available with each application.

Firmware

A newer version of the firmware may introduce new parameters. Parameters that have been configured for files of an older firmware function may not be visible in newer versions. In such cases, the respective parameter description includes a corresponding note.

3.3 Signal transmission and fieldbus mapping

The transmission of control signals and set values in DriveControlSuite meets the following principles.

Signal transmission

Drive controllers are either controlled over a fieldbus, using mixed operation consisting of a fieldbus system and terminals or exclusively using terminals.

You can use the corresponding selection parameters, referred to as data sources, to configure whether the signals and set values are obtained over a fieldbus or using terminals.

In case of activation over a fieldbus, parameters that are selected as data sources for control signals or set values must be part of the subsequent fieldbus mapping; in the case of activation using terminals, the respective analog or digital inputs are specified directly.

Fieldbus mapping

If you are working with a fieldbus system and have selected the source parameters for signals and set values, configure the fieldbus-specific settings, e.g. the layout of process data channels for transmitting receive and transmit process data, as the last step.

The respective procedure can be found in the accompanying STOBER fieldbus manuals.

3.4 Power-loss protected storage

All project configurations, parameterizations and related changes to parameter values are in effect after transmission to the drive controller, but are not yet stored in non-volatile memory.

Saving to a drive controller

You have the following options for saving the configuration to a drive controller in non-volatile memory:

- Saving the configuration using the **Save values wizard**:
Project menu > Wizards area > Projected axis > Save values wizard: Select the Save values action
- Saving the configuration using the parameter list:
Project menu > Parameter list area > Projected axis > Group A: Drive controller > A00 Save values: Set the parameter A00[0] to the value 1: Active
- Saving the configuration using the operating unit:
SD6 drive controller > Operating unit: Press the save button for 3 seconds

Saving to all drive controllers within a project

You have the following options for saving the configuration to multiple drive controllers in non-volatile memory:

- Saving the configuration using the toolbar:
Toolbar > Save values symbol: Click on the Save values symbol
- Saving the configuration using the **Assignment and live firmware update window**:
Project menu > Assignment and live firmware update area > Projected axis > Assignment and live firmware update window: Click on Save values (A00)

Information

Do not shut off the drive controller while saving. If the supply voltage to the control unit is interrupted while saving, the drive controller starts without an executable configuration when it is next switched on. In this case, the configuration must be transferred to the drive controller again and saved in non-volatile memory.

4 Commissioning

This chapter describes how to commission your drive system using the DriveControlSuite DS6 software.

For the commissioning, we require that you use a motor in one of the following designs corresponding to your drive controller:

Drive controller	SD6	SC6	SI6
Motor	STOBER synchronous servo motor with optional brake	STOBER synchronous servo motor with optional brake, STOBER Lean motor	STOBER synchronous servo motor with optional brake, STOBER Lean motor
Encoders	EnDat 2.1/2.2 digital encoder, HIPERFACE DSL encoder	EnDat 2.2 digital encoder, HIPERFACE DSL encoder	EnDat 2.2 digital encoder, HIPERFACE DSL encoder

These motors together with all relevant data for the project configuration are saved in the motor database of DriveControlSuite as well as in the [electronic nameplate](#). By selecting a corresponding motor, all data is automatically transmitted to the corresponding parameters. There is no need for complex parameterization of the motor, brake or encoder.

If you use a different motor type, you must parameterize it manually.

Before you begin commissioning, make sure that the system nodes are hooked up and supplied with control voltage.

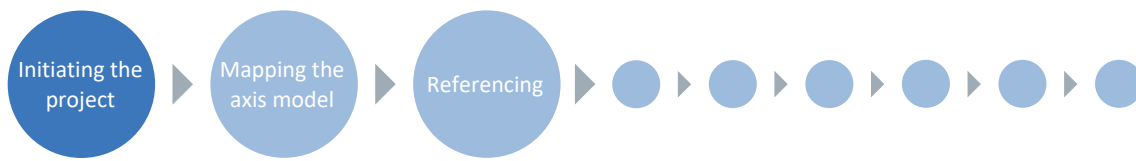
You can find more information about installing and commissioning your drive controllers in the corresponding commissioning instructions.

Information

Always perform the steps included in the following chapters in the specified order!

Some parameters of the DriveControlSuite are interdependent and do not become accessible to you until you have first configured certain settings. Follow the steps in the specified sequence so that you can finish the parameterization completely.

4.1 Initiating the project



To be able to use the functions of the Drive Based synchronous application, you need at least one drive controller of the SC6, SD6 or SI6 series. Record all drive controllers and axes of your drive system in a project to configure it using DriveControlSuite.

4.1.1 Projecting the drive controller and axis

Create a new project and project the first drive controller along with the accompanying axis.

Creating a new project

1. Start DriveControlSuite.
 2. Click Create new project.
- ⇒ The project configuration window opens and the Drive controller button is active.

Projecting the drive controller

1. **Properties tab:**
 Establish the relationship between your circuit diagram and the drive controller to be projected in DriveControlSuite.
 Reference: Specify the reference code (equipment code) of the drive controller.
 Designation: Give the drive controller a unique name.
 Version: Version your project configuration.
 Description: If necessary, specify additional supporting information, such as the change history of the project configuration.
2. **Drive controller tab:**
 Select the series, device type and firmware version of the drive controller.
3. **Option modules tab:**
 Depending on the drive controller series, various option modules are available for you to select.
 Communication module: If the drive controller communicates with a controller over a fieldbus, select the corresponding communication module.
 Terminal module: If you are controlling the drive controller using analog and digital inputs, select the corresponding terminal module (in addition to the communication module in mixed operation).
 Safety module: If the drive controller is part of a safety circuit, select the corresponding safety module.
4. **Device control tab:**
 Device control: Select Drive Based.
 Receive process data (Rx), transmit process data (Tx): If you control the drive controller using a fieldbus, select the fieldbus-specific receive and transmit process data.

Projecting the axis

1. Click on Axis 1.
 2. **Properties tab:**
Establish the connection between your circuit diagram and the axis to be projected in DriveControlSuite.
Reference: Specify the reference code (equipment code) of the axis.
Designation: Give the axis a unique name.
Version: Version your project configuration.
Description: If necessary, specify additional supporting information, such as the change history of the project configuration.
 3. **Application tab:**
Select Drive Based synchronous.
 4. **Motor tab:**
Select the type of motor operated using this axis.
 5. If you are using a double-axis controller, repeat the steps for the second axis.
 6. Confirm with OK.
- ⇒ The configuration dialog closes and the drive controller is created in a new module in the project tree.

4.1.2 Configuring safety technology

If the drive controller is part of a safety circuit, you must configure the safety technology in accordance with the commissioning steps outlined in the corresponding manual in the next step.

4.1.3 Creating other modules and drive controllers

In DriveControlSuite, all drive controllers within a project are grouped using modules. If you add a new drive controller to your project, be sure to always assign it to an existing module. Group drive controllers in a module if, for example, they are located in the same control cabinet, work together to operate the same machine part or are networked with each other over IGB motion bus.

Creating drive controllers

1. In the project tree, select your project P1 > module M1 > context menu Create new drive controller.
⇒ The drive controller is created in the project tree and the configuration dialog opens.
2. Project the drive controller as described in the chapter [Projecting the drive controller and axis \[► 20\]](#).
3. Repeat the steps for all other drive controllers that you want to project.

Creating a module

1. In the project tree, select your project P1 > context menu Create new module.
⇒ The module is created in the project tree.
2. Specify the module as described in the chapter [Specifying a module \[► 22\]](#).
3. Repeat the steps for all other modules that you want to project.

4.1.4 Specifying a module

After you have created and projected all the drive controllers that you would like to gather under one module, specify the module.

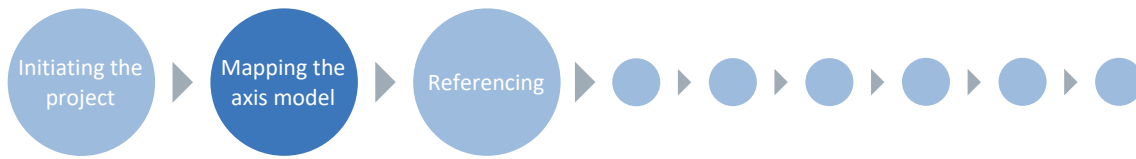
1. Highlight your M1 module in the project tree.
2. Change to the project menu and click Project configuration.
⇒ The Module window opens.
3. Establish the relationship between your circuit diagram and the newly created module in DriveControlSuite.
Reference: Specify the reference code (equipment code) of the module.
Designation: Give the module a unique name.
Version: Version the module.
Description: If necessary, specify additional supporting information, such as the change history of the module.
4. Confirm with OK.

4.1.5 Specifying the project

Finally, specify your project.

1. Highlight your P1 project in the project tree.
2. Change to the project menu and click Project configuration.
⇒ The Project window opens.
3. Establish the relationship between your circuit diagram and the newly created project in DriveControlSuite.
Reference: Specify the reference code (equipment code) of the project.
Designation: Give the project a unique name.
Version: Version the project.
Description: If necessary, specify additional supporting information, such as the change history of the project.
4. Confirm with OK.

4.2 Mapping the mechanical axis model



To be able to put your real drive train with one or more drive controllers into operation, you must map your complete mechanical environment in DriveControlSuite. STOBER drive controllers of the 6th generation are specially developed for communication between the drive controller and controller based on actual variables at the output (° or mm of real axis movement). The scaling of the axis model is calculated by the drive controller's firmware independently of the encoder model, without any rounding errors or drift.

4.2.1 Parameterizing a STOBER motor

You have projected a STOBER synchronous servo motor with EnDat encoder and integrated brake.

By projecting the corresponding motor, limiting values for currents and torques as well as associated temperature data are automatically transferred to the respective parameters of the individual wizards. At the same time, all additional data on the energy supply, brake and encoder is transferred.

4.2.2 Parameterizing the axis model

Parameterize the setup of your drive in this order:

- Define the axis model
- Scale the axis
- Parameterize the position and velocity window
- Limit the axis (optional)
 - Limit the position
 - Limit the velocity, acceleration and jerk
 - Limit the torque and force

Information

If you are using a double-axis controller with two projected axes, you must parameterize the axis model for each axis individually.

4.2.2.1 Define the axis model

1. Highlight the relevant drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
2. Select the Axis model wizard.
3. I05 Type of axis:
Define whether the axis type is rotational or translational.
 - 3.1. If you would like to configure the units of measure and the number of decimal places individually for specifying and displaying position set values, velocities, accelerations and jerk, select 0: User defined, rotational or 1: User defined, translational.
 - 3.2. If the units of measure and the number of decimal places for specifying and displaying position set values, velocities, accelerations and jerk are to be fixed, select 2: Rotational or 3: Translational.
4. B26 Motor encoder:
Define the interface to which the motor encoder is connected.
5. I02 Position encoder (optional):
Define the interface to which the position encoder is connected.
6. I00 Position range:
Define whether the travel range of the axis is limited or endless (modulo).
7. If you have selected 1: Endless for I00, parameterize a revolution length as described in the chapter [Scale the axis](#) [▶ 25].

Information

If you parameterize I05 Type of axis, you can either use the selections 0: User defined, rotational or 1: User defined, translational to configure units of measure and the number of decimal places for the axis model individually or use the selections 2: Rotational and 3: Translational to revert to default values.

Selecting 2: Rotational sets the following units of measure for the axis model: position in °, velocity in rpm, acceleration in rad/s^2 , jerk in rad/s^3 .

Selecting 3: Translational sets the following units of measure for the axis model: position in mm, velocity in m/s, acceleration in m/s^2 , jerk in m/s^3

Information

If you do not parameterize it differently for I02 Position encoder, B26 Motor encoder is used for position control as standard.

4.2.2.2 Scale the axis

1. Highlight the relevant drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
2. Select the Axis model wizard > Axis: Scaling.
3. Scale the axis by configuring the overall gear ratio between the motor and output.
To simplify this scaling for you, you are provided with the scaling calculator **Conversion of position, velocities, accelerations, torque/force**, which calculates the effects of changed movement variables on the entire system.
4. I01 Circular length:
If you have selected 1: Endless for I00 Position range, enter the revolution length.
5. I06 Decimal places position (optional):
If you have selected 0: User defined, rotational or 1: User defined, translational for I05 Type of axis, define the desired number of decimal places.
6. I09 Measure unit (optional):
If you have selected 0: User defined, rotational or 1: User defined, translational for I05 Type of axis, define the desired unit of measure.

Information

A change to parameter I06 moves the decimal separator for all axis-specific values!
Ideally, change I06 before parameterizing other axis-specific values and then check them afterwards.

Information

Parameter I297 Maximum speed position encoder must be parameterized according to your application case. If I297 is set too low, the permitted maximum speed is exceeded even at normal operating speeds. On the other hand, if I297 is set too high, measuring errors of the encoder can be overlooked.

I297 depends on the following parameters: I05 Type of axis, I06 Decimal places position, I09 Measure unit as well as I07 Distance factor numerator position and I08 Distance factor denominator position for Drive Based or A585 Feed constant for CiA 402. If you have made changes to one of the parameters listed, select I297 accordingly as well.

4.2.2.3 Parameterizing the position and velocity window

Enter position limits and velocity zones for set values. To do so, parameterize boundary values for reaching a position or velocity.

1. Select the Axis model wizard > Window position, velocity.
2. C40 Velocity window:
Parameterize a tolerance range for velocity tests.
3. I22 Target window:
Parameterize a tolerance range for position tests.
4. I87 Actual position in window time:
Parameterize how long a drive must stay in the specified position window before a corresponding status message is output.
5. I21 Maximal following error
Parameterize a tolerance range for lag tests.

4.2.2.4 Limiting the axis

If necessary, limit the movement variables for position, velocity, acceleration, jerk as well as torque/force according to the applicable conditions for your axis model.

Limiting the position (optional)

1. Highlight the relevant drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
2. Select the Axis model wizard > Limit: Position.
3. If necessary, limit the position of your axis using a software or hardware limit switch to secure the travel range.

Limiting velocity, acceleration, jerk (optional)

The specified default values are designed for slow velocities without gear units. For this reason, adapt the saved values.

Note that the velocity of the motor is parameterized in units other than that of the axis model. Verify the velocity of the motor against the velocity of the output accordingly.

1. Select the Motor wizard.
2. To determine the maximum velocity at the output, copy the value of the B13 Nominal motor speed parameter to the clipboard.
3. Select the Axis model wizard > Axis: Scaling > Conversion of positions, velocities, accelerations, torque/force area.
4. Velocity line:
Paste the copied value of the B13 parameter from the clipboard without the unit and confirm with ENTER.
⇒ The maximum velocity of the motor has been transferred to the output.
5. Select the Axis model wizard > Limit: Velocity, acceleration, jerk.
6. I10 Maximal speed:
Limit the maximum velocity of the output taking into account the configured Nominal motor speed in B13.
7. Determine the limiting values for acceleration and jerk if necessary and enter them into the associated parameters.

Limiting torque/force (optional)

The specified default values take into account the rated operation together with the overload reserves.

1. Select the Axis model wizard > Limit: Torque/force.
2. If the motor force must be limited, adapt the saved values as necessary.

4.3 Referencing absolute position



When commissioning a system with position measurement systems, the relation of a measured to an actual axis position must be determined.

If you are working with absolute positions, reference your drive now. Otherwise, proceed by parameterizing application-specific conditions and responses of the Drive Based device control.

Information

If you are using a double-axis controller with two projected axes, you must reference the absolute position for each axis individually.

4.3.1 Defining the referencing method

1. Highlight the relevant drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
2. Select the Referencing wizard.
3. I30 Referencing type, I31 Referencing direction, I35 Referencing with zero pulse:
The possible combinations of the named characteristics as well as detailed descriptions of the individual referencing methods can be found in the chapter [Referencing methods \[▶ 100\]](#).
4. I43 Move to reference position:
If the axis is to move to the referenced position after a reference run automatically, set this parameter to 1: Active.

4.3.2 Parameterizing the reference switch

If you are working with absolute positions and would like to determine a reference position while referencing using the reference switch, proceed as follows.

1. Select the Application wizard > Drive Based synchronous > Data sources > Application digital signals: Data source.
2. I103 Source reference switch:
 - 2.1. If a digital input (direct or inverted) acts as the signal source, select the corresponding input.
 - 2.2. If the control word I210 of the Application > Drive Based synchronous acts as the signal source, select 2: Parameter.
Bit 3 of the control word is set as the data source for the reference switch.
Note that individual bits of the control word are transmitted only if the control word is part of the process data mapping.

4.3.3 Setting the reference

If you are working with absolute positions and would like to reference without using regular referencing methods, i.e. on-the-fly, the value of the current actual position is applied directly as a parameter value from I34 using the "Set reference" signal.

1. Select the Application wizard > Drive Based synchronous > Data sources > Application digital signals: Data source.
2. I111 Source set reference:
 - 2.1. If a digital input (direct or inverted) acts as the signal source, select the corresponding input.
 - 2.2. If the control word I210 of the Application > Drive Based synchronous acts as the signal source, select 2: Parameter.
Bit 11 of the control word is set as a reference source.
Note that individual bits of the control word are transmitted only if the control word is part of the process data mapping.

4.3.4 Parameterizing the limit switch

Using the limit switch, you can limit the travel range in the positive and negative direction using sensor signals, regardless of the operating mode. Both signals are protected against wire breaks (= low active, i.e. a low level leads to a limit switch fault).

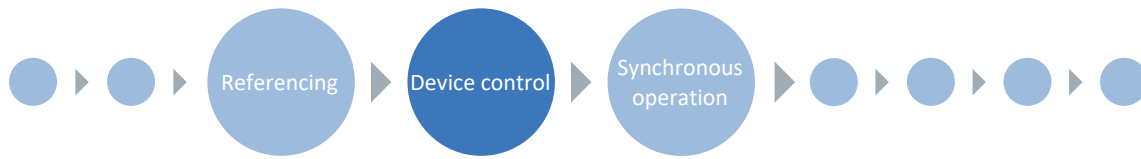
Parameterizing a positive limit switch

1. Select the Application wizard > Drive Based synchronous > Data sources > Application digital signals: Data source.
2. I101 Source positive /limit switch:
 - 2.1. If a digital input (direct or inverted) acts as the signal source, select the corresponding input.
 - 2.2. If the control word I210 of the Application > Drive Based synchronous acts as a signal source, select 2: Parameter.
Bit 1 of the control word is set as a data source for the positive limit switch.
Note that individual bits of the control word are transmitted only if the control word is part of the process data mapping.

Parameterizing a negative limit switch

1. Select the Application wizard > Drive Based synchronous > Data sources > Application digital signals: Data source.
2. I102 Source negative /limit switch:
 - 2.1. If a digital input (direct or inverted) acts as the signal source, select the corresponding input.
 - 2.2. If the control word I210 of the Application > Drive Based synchronous acts as a signal source, select 2: Parameter.
Bit 2 of the control word is set as a data source for the negative limit switch.
Note that individual bits of the control word are transmitted only if the control word is part of the process data mapping.

4.4 Parameterizing Drive Based device control



The Drive Based device control describes the control sequence of a drive controller based on a state machine, i.e. a drive controller switches its state due to an event. Some of the conditions and responses coupled with the state transitions can be influenced depending on the application; for example, it is possible to define the end of a quick stop or enable signal delays tailored to the respective application case.

Detailed information on the device control and Drive Based device state machine can be found in the chapter [Drive Based device control](#) [▶ 126].

4.4.1 Parameterizing transition conditions

✓ You have projected a drive controller with the Drive Based device control.

1. Highlight the relevant drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
2. Select the Drive Based device control wizard.
3. A34 Auto start:
If the drive controller is to switch directly to the Operation enabled state after the Switch on disabled state, activate this parameter.
4. A43 Enable switch-on/off delay
If you need to hide possible fault or test pulses on the enable signal input, enter the maximum internal activation and deactivation delay.
5. A44 Quick stop for enable off:
If the drive is to be stopped with a quick stop in the event of a deactivated enable signal, activate this parameter.
6. A60 Source additional enable:
If you are working with multiple enable signals, define the source of additional signals in this parameter.
7. A29 Quick stop in case of fault:
If the drive is to be stopped with a quick stop—if possible—in case of a fault, activate this parameter. If the parameter is deactivated, motion of the axis is no longer controlled by the drive controller.
8. A39 Maximum quick stop duration for enable off:
If you have selected 1: Active for parameter A44, define the maximum time after which the power unit is switched off in A39.
9. A45 Quick stop end:
In this parameter, define whether a quick stop is considered ended when the drive is at a standstill or when the quick stop request is canceled.
10. A62 Source /quick stop:
Define how a quick stop is triggered in this parameter.
11. E48 Device control state:
Shows the current state of the drive controller.
12. If you are using a double-axis controller, click in the project menu > Wizard area on the second projected axis and repeat the steps.

4.5 Configuring synchronous operation



The following chapters describe the different configuration options you have in DriveControlSuite for synchronous operation. Follow the handling instructions for the approach that conforms to the conditions of your system environment.

Different configuration options are available in DriveControlSuite to operate one or more axes in synchronous operation.

Synchronous operation: Model 1

- Master value source: Virtual master or master encoder
- Multiple axes move synchronously with the master value source

The master values are provided by a virtual master or a master encoder. For double-axis controllers, both axes follow the master value that is provided by an internal device master axis model. For SD6 type single-axis controllers, a master drive controller provides the master value over IGB motion bus to both its own axis as well as other slave drive controllers across devices.

Synchronous operation: Model 2

- Master value source: Master encoder
- A single axis moves synchronously with the master value source

The master values are provided by a master encoder. For double-axis controllers as well as single-axis controllers, only one individual axis follows the master value that is provided by an internal device master axis model.

Synchronous operation: Model 3

- Master value source: Motor axis model of the master axis
- Multiple axes move synchronously with the master value source

The master values are provided by the actual position of the master axis. For double-axis controllers, the second axis follows the master value that is provided by an internal device master axis model. For single-axis controllers, a master drive controller provides the master value over IGB motion bus to other slave drive controllers across devices.

4.5.1 Model 1: Virtual master or master encoder, multiple slave axes

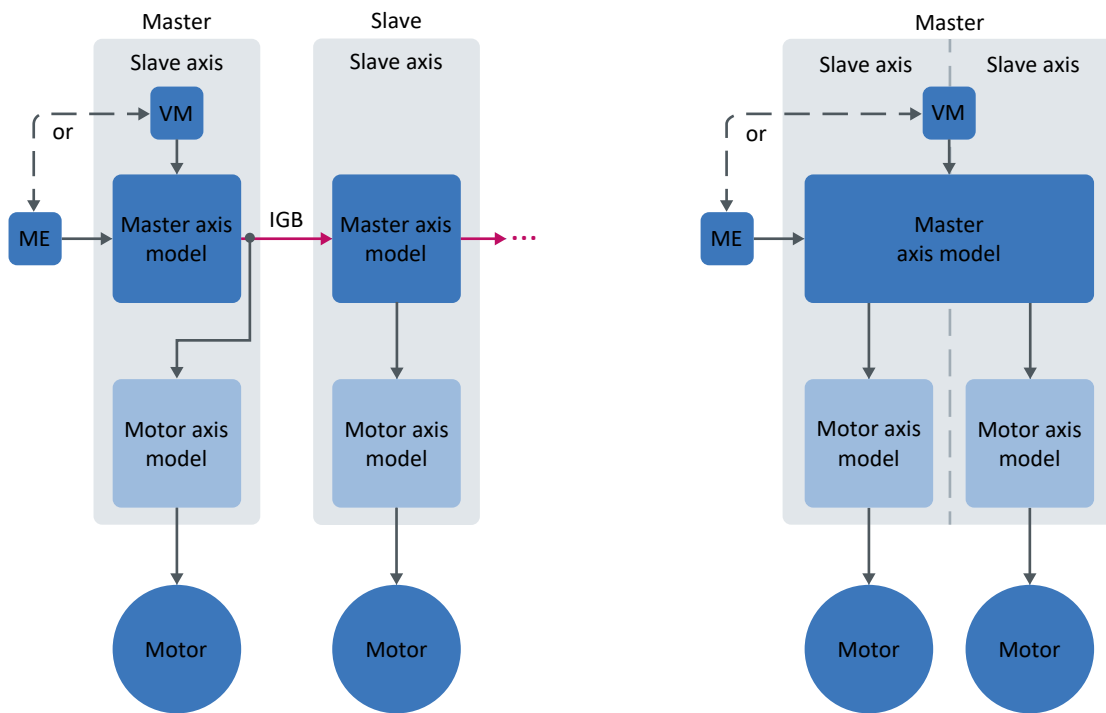


Fig. 3: Synchronous operation: Model 1

The master values are provided by a virtual master or a master encoder. For double-axis controllers, both axes follow the master value that is provided by an internal device master axis model. For single-axis controllers, a master drive controller provides the master value over IGB motion bus to both its own axis as well as other slave drive controllers across devices.

Proceed as follows:

- Parameterize the master:
Scale the master axis model and parameterize the master value source (virtual master or master encoder).
- Then, define the master value source for the slave axes.

Information

For double-axis controllers, the wizards for synchronous operation of both axes are coupled with each other. If you configure parameter values for one of the two axes, these are automatically taken over at several places for the second axis. Parameterizations that deviate from this are described in the following step-by-step instructions.

4.5.1.1 Parameterizing the master

Scale the master axis model and then parameterize the master value source as described below.

4.5.1.1.1 Scaling the master axis model

Scale the master axis model as described below.

1. Highlight the relevant drive controller in the project tree and click on the axis for which you would like to scale the master axis model in the project menu > Wizard area.
2. Select the Drive Based Synchronous application wizard > Synchronous operation > Master: Axis model, scaling.
3. G30 Position range master:
Define the travel range of the master axis.
4. G40 Circular length master:
If you have selected 1: Endless for G30, enter the master position from which counting begins again at zero.
5. I418 Gear ratio denominator and I417 Gear ratio numerator:
Enter a factor for the gear ratio of the master position and axis movement in the form of a quotient (numerator and denominator).
6. G46 Decimal places:
Enter the number of decimal places for specifying and displaying the master positions.
7. G49 Master measure unit:
Enter the desired unit for the master positions. Any unit of measure can be selected (max. 8 characters).
8. If you are using a double-axis controller, enter a factor for the gear ratio for the second axis as well: Click on the second projected axis in the project menu > Wizard area and repeat steps 1, 2 and 5.

Information

A change to parameter G46 moves the decimal separator for all master position values. Therefore, change G46 before parameterizing other master position values and then check them.

Information

For the synchronous operation of multiple SD6 drive controllers over an IGB motion bus network, the master axis model must be scaled identically for master and slave axes (decimal places, unit of measure, travel range and revolution length, if applicable). Scale the master axis model identically from the start so that you do not have to adapt position settings to changed scaling afterwards.

This always applies to parameters G30, G40, G46 and G49 for a slave axis, and to G30, G40, G46 and G49 (master value source: virtual master or master encoder) or I00, I01, I06 and I09 (master value source: motor axis model of the master) for the master axis.

4.5.1.1.2 Parameterizing the virtual master

If you are working with a virtual master, activate it first. Next, limit the position, velocity, acceleration and jerk for the virtual master and then define it as the master value source.

Activating the virtual master

1. Highlight the relevant drive controller in the project tree and click on the axis for which you would like to parameterize the virtual master in the project menu > Wizard area.
2. Select the Drive Based Synchronous application wizard > Additional functions.
3. Activate the option Virtual master.
⇒ In the list of wizards, the Virtual master wizard is displayed.

Parameterizing the virtual master

✓ You have activated the additional virtual master function.

1. Highlight the relevant drive controller in the project tree and click on the axis for which you would like to parameterize the virtual master in the project menu > Wizard area.
2. Select the Virtual master wizard > Limit: Position, velocity, acceleration, jerk for virtual master.
3. G30 Position range master:
Define the travel range of the virtual master.
4. G40 Circular length master:
If you have selected 1: Endless for G30, enter the master position from which counting begins again at zero.
5. G146 Master software limit switch +, G147 Master software limit switch -:
In order to protect the travel range, limit the position of your axis if necessary using a positive or negative software limit switch.
6. G50 virtual master maximal speed:
Enter the maximum velocity of the virtual master.
7. G51 virtual master maximal acceleration:
Enter the maximum acceleration of the virtual master.
8. G54 virtual master quickstop deceleration:
Enter the maximum quick-stop acceleration of the virtual master.
9. G52 virtual master maximal jerk:
Enter the maximum jerk of the virtual master.

Information

The limit variables for the velocity, acceleration, quick stop and jerk of the virtual master (G50, G51, G52, G54) depend on the limit variables of the slave axes.

To determine the maximum velocity for the virtual master, for example, convert the maximum velocity of each individual slave axis together with the respective gear factor I417 Gear ratio numerator and I418 Gear ratio denominator into an alternative maximum velocity. The slave axis with the lowest maximum velocity is simultaneously the velocity limit value of the virtual master.

Proceed in the same manner for the limit values for acceleration, quick stop and jerk.

Setting the virtual master as the master value source

1. Highlight the relevant drive controller in the project tree and click on the axis for which you would like to parameterize the virtual master in the project menu > Wizard area.
2. Select the Drive Based Synchronous application wizard > Synchronous operation > Master.
3. G126 Source IGB producer position:
Select 2: Virtual master.
4. Select the Drive Based Synchronous application wizard > Synchronous operation > Slave.
5. G27 Source master position:
If the master axis itself is to follow the master value source, select 2: Virtual master or 6: IGB motion bus.

4.5.1.1.3 Parameterizing the master encoder

If you are working with an external master encoder, first activate the interface you have used to connect the master encoder with the drive controller. Then, parameterize the master encoder along with the encoder interface and, finally, set it as the master value source.

Parameterizing the master encoder

1. Highlight the corresponding drive controller in the project tree and click on the axis for which you would like to parameterize the master encoder in the project menu > Wizard area.
2. Select the Encoder wizard > Master encoder: Scaling.
3. G104 Source master encoder:
Select the interface over which the master encoder is connected.
4. G47 Distance factor numerator master and G48 Distance factor denominator master:
Enter the conversion factor for the path of the master encoder in relation to the master position.
5. G297 maximum-speed master-encoder
Define the value for the permitted maximum velocity of the master encoder.
6. If you have connected the master encoder over the X4 interface, select the Encoder wizard > X4.
 - 6.1. H00 X4 function:
Select the encoder model that is connected to the interface.
⇒ The appropriate parameters are displayed based on the selected encoder.
 - 6.2. H03 Encoder version:
Select whether the encoder design is rotational or translational.
 - 6.3. Parameterize the interface according to the properties of the master encoder.
7. If you have connected the master encoder over the X120 interface, select the Encoder wizard > X120.
 - 7.1. H120 X120 function:
Select the encoder model that is connected to the interface.
⇒ The appropriate parameters are displayed based on the selected encoder.
 - 7.2. H123 Encoder version:
Select whether the encoder design is rotational or translational.
 - 7.3. Parameterize the interface according to the properties of the master encoder.
8. If you have connected the master encoder over the X140 interface, select the Encoder wizard > X140.
 - 8.1. H140 X140 function:
Select the encoder model that is connected to the interface.
⇒ The appropriate parameters are displayed based on the selected encoder.
 - 8.2. H143 Encoder version:
Select whether the encoder design is rotational or translational.
 - 8.3. Parameterize the interface according to the properties of the master encoder.

Information

Parameterize G297 maximum-speed master-encoder according to your application case: If G297 is set too low, the permitted maximum speed is exceeded even at normal operating speeds. If G297 is set too high, measuring errors of the encoder can be overlooked.

G297 depends on the following parameters: G46 Decimal places, G47 Distance factor numerator master, G48 Distance factor denominator master and G49 Master measure unit. If you have made changes to one of the parameters listed, adjust G297 accordingly as well.

Setting the master encoder as the master value source

1. Highlight the corresponding drive controller in the project tree and click on the axis for which you would like to parameterize the master encoder in the project menu > Wizard area.
2. Select the Drive Based Synchronous application wizard > Synchronous operation > Master.
3. G126 Source IGB producer position:
Select 1: Master encoder.
4. Select the Drive Based Synchronous application wizard > Synchronous operation > Slave.
5. G27 Source master position:
If the master axis itself is to follow the master value source, select 1: Master encoder or 6: IGB motion bus.

4.5.1.2 Parameterizing slave axes

Parameterize the master value source for the slave axes. If you have networked multiple drive controllers with each other over IGB motion bus, scale the master axis models for the slaves as well.

4.5.1.2.1 Setting the master value source for slave axes

Define the master value source for the slave axes as described below.

1. Highlight the relevant drive controller in the project tree and click on the axis which you would like to parameterize as a slave in the project menu > Wizard area.
2. Select the Drive Based Synchronous application wizard > Synchronous operation > Slave.
3. G27 Source master position:
Set the master value source for the slave axis.
 - 3.1. If you are using a double-axis controller, select 1: Master encoder or 2: Virtual master according to your master value source.
 - 3.2. If you are using a single-axis controller, select 6: IGB motion bus.
4. If you have networked multiple drive controllers with each other over IGB motion bus, repeat the steps for all other slaves in the IGB motion bus network.

4.5.1.2.2 Scaling master axis models

If you have networked multiple drive controllers with each other over IGB motion bus, scale the master axis model for the slaves as described below.

1. Highlight the relevant drive controller in the project tree and click on the axis for which you would like to scale the master axis model in the project menu > Wizard area.
2. Select the Drive Based Synchronous application wizard > Synchronous operation > Master: Axis model, scaling.
3. G30 Position range master:
Define the travel range of the master axis.
4. G40 Circular length master:
If you have selected 1: Endless for G30, enter the master position from which counting begins again at zero.
5. I418 Gear ratio denominator and I417 Gear ratio numerator:
Enter a factor for the gear ratio of the master position and axis movement in the form of a quotient (numerator and denominator).
6. G46 Decimal places:
Enter the number of decimal places for specifying and displaying the master positions.
7. G49 Master measure unit:
Enter the desired unit for the master positions. Any unit of measure can be selected (max. 8 characters).
8. If you have networked multiple drive controllers with each other over IGB motion bus, repeat the steps for all other slaves in the IGB motion bus network.

Information

A change to parameter G46 moves the decimal separator for all master position values. Therefore, change G46 before parameterizing other master position values and then check them.

Information

For the synchronous operation of multiple SD6 drive controllers over an IGB motion bus network, the master axis model must be scaled identically for master and slave axes (decimal places, unit of measure, travel range and revolution length, if applicable). Scale the master axis model identically from the start so that you do not have to adapt position settings to changed scaling afterwards.

This always applies to parameters G30, G40, G46 and G49 for a slave axis, and to G30, G40, G46 and G49 (master value source: virtual master or master encoder) or I00, I01, I06 and I09 (master value source: motor axis model of the master) for the master axis.

4.5.2 Model 2: Master encoder, individual slave axis

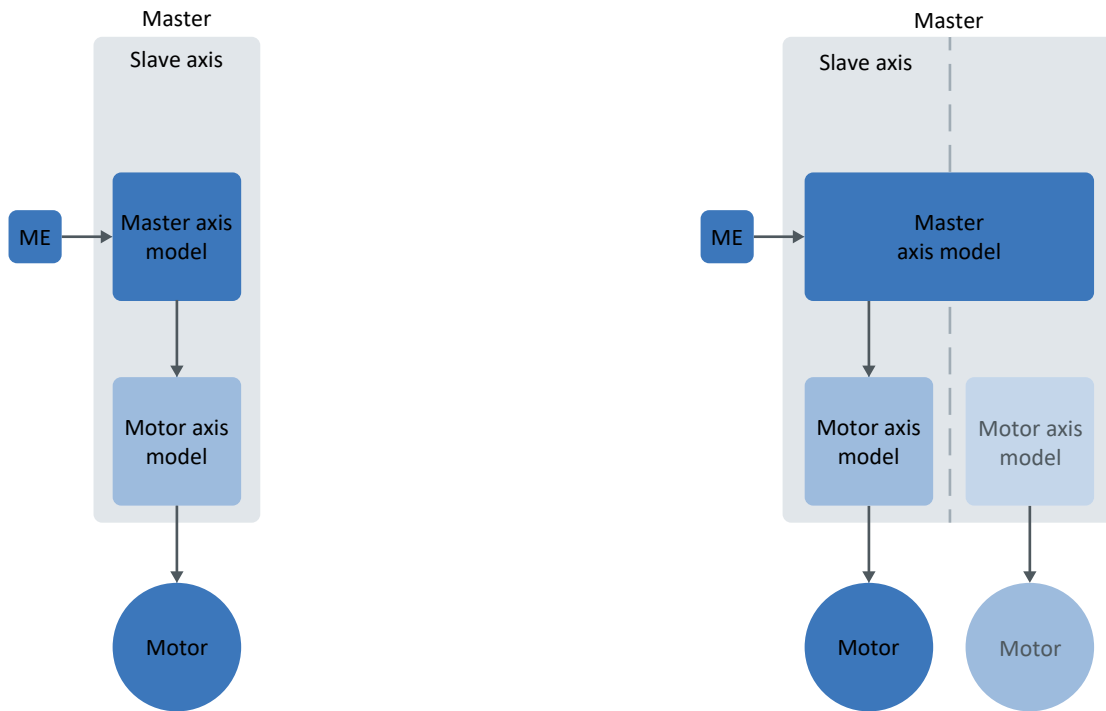


Fig. 4: Synchronous operation: Model 2

The master values are provided by a master encoder. For double-axis controllers as well as single-axis controllers, only one individual axis follows the master value that is provided by an internal device master axis model.

Proceed as follows:

- Parameterize the master:
Scale the master axis model and parameterize the master value source (master encoder).
- There is no need to parameterize additional slave axes.

Information

If you are using a double-axis controller with two projected axes, you can operate one axis in the Drive Based synchronous application and another axis in the Drive Based application. This enables you to use the axes independently of each other.

Information

There is no need to parameterize additional slave axes for this model. The master value source and the associated parameter G27 Source master position can be parameterized for the individual slave axis as described in the chapter [Parameterizing the master encoder \[▶ 40\]](#).

4.5.2.1 Parameterizing the master

Scale the master axis model and then parameterize the master value source as described below.

4.5.2.1.1 Scaling the master axis model

Scale the master axis model as described below.

1. Highlight the relevant drive controller in the project tree and click on the axis for which you would like to scale the master axis model in the project menu > Wizard area.
2. Select the Drive Based Synchronous application wizard > Synchronous operation > Master: Axis model, scaling.
3. G30 Position range master:
Define the travel range of the master axis.
4. G40 Circular length master:
If you have selected 1: Endless for G30, enter the master position from which counting begins again at zero.
5. I418 Gear ratio denominator and I417 Gear ratio numerator:
Enter a factor for the gear ratio of the master position and axis movement in the form of a quotient (numerator and denominator).
6. G46 Decimal places:
Enter the number of decimal places for specifying and displaying the master positions.
7. G49 Master measure unit:
Enter the desired unit for the master positions. Any unit of measure can be selected (max. 8 characters).
8. If you are using a double-axis controller, enter a factor for the gear ratio for the second axis as well: Click on the second projected axis in the project menu > Wizard area and repeat steps 1, 2 and 5.

Information

A change to parameter G46 moves the decimal separator for all master position values. Therefore, change G46 before parameterizing other master position values and then check them.

Information

For the synchronous operation of multiple SD6 drive controllers over an IGB motion bus network, the master axis model must be scaled identically for master and slave axes (decimal places, unit of measure, travel range and revolution length, if applicable). Scale the master axis model identically from the start so that you do not have to adapt position settings to changed scaling afterwards.

This always applies to parameters G30, G40, G46 and G49 for a slave axis, and to G30, G40, G46 and G49 (master value source: virtual master or master encoder) or I00, I01, I06 and I09 (master value source: motor axis model of the master) for the master axis.

4.5.2.1.2 Parameterizing the master encoder

If you are working with an external master encoder, first activate the interface you have used to connect the master encoder with the drive controller. Then, parameterize the master encoder along with the encoder interface and, finally, set it as the master value source.

Parameterizing the master encoder

1. Highlight the corresponding drive controller in the project tree and click on the axis for which you would like to parameterize the master encoder in the project menu > Wizard area.
2. Select the Encoder wizard > Master encoder: Scaling.
3. G104 Source master encoder:
Select the interface over which the master encoder is connected.
4. G47 Distance factor numerator master and G48 Distance factor denominator master:
Enter the conversion factor for the path of the master encoder in relation to the master position.
5. G297 maximum-speed master-encoder
Define the value for the permitted maximum velocity of the master encoder.
6. If you have connected the master encoder over the X4 interface, select the Encoder wizard > X4.
 - 6.1. H00 X4 function:
Select the encoder model that is connected to the interface.
⇒ The appropriate parameters are displayed based on the selected encoder.
 - 6.2. H03 Encoder version:
Select whether the encoder design is rotational or translational.
 - 6.3. Parameterize the interface according to the properties of the master encoder.
7. If you have connected the master encoder over the X120 interface, select the Encoder wizard > X120.
 - 7.1. H120 X120 function:
Select the encoder model that is connected to the interface.
⇒ The appropriate parameters are displayed based on the selected encoder.
 - 7.2. H123 Encoder version:
Select whether the encoder design is rotational or translational.
 - 7.3. Parameterize the interface according to the properties of the master encoder.
8. If you have connected the master encoder over the X140 interface, select the Encoder wizard > X140.
 - 8.1. H140 X140 function:
Select the encoder model that is connected to the interface.
⇒ The appropriate parameters are displayed based on the selected encoder.
 - 8.2. H143 Encoder version:
Select whether the encoder design is rotational or translational.
 - 8.3. Parameterize the interface according to the properties of the master encoder.

Information

Parameterize G297 maximum-speed master-encoder according to your application case: If G297 is set too low, the permitted maximum speed is exceeded even at normal operating speeds. If G297 is set too high, measuring errors of the encoder can be overlooked.

G297 depends on the following parameters: G46 Decimal places, G47 Distance factor numerator master, G48 Distance factor denominator master and G49 Master measure unit. If you have made changes to one of the parameters listed, adjust G297 accordingly as well.

Setting the master encoder as the master value source

1. Highlight the corresponding drive controller in the project tree and click on the axis for which you would like to parameterize the master encoder in the project menu > Wizard area.
2. Select the Drive Based Synchronous application wizard > Synchronous operation > Master.
3. G126 Source IGB producer position:
Select 1: Master encoder.
4. Select the Drive Based Synchronous application wizard > Synchronous operation > Slave.
5. G27 Source master position:
If the master axis itself is to follow the master value source, select 1: Master encoder or 6: IGB motion bus.

4.5.3 Model 3: Master axis, one or more slave axes

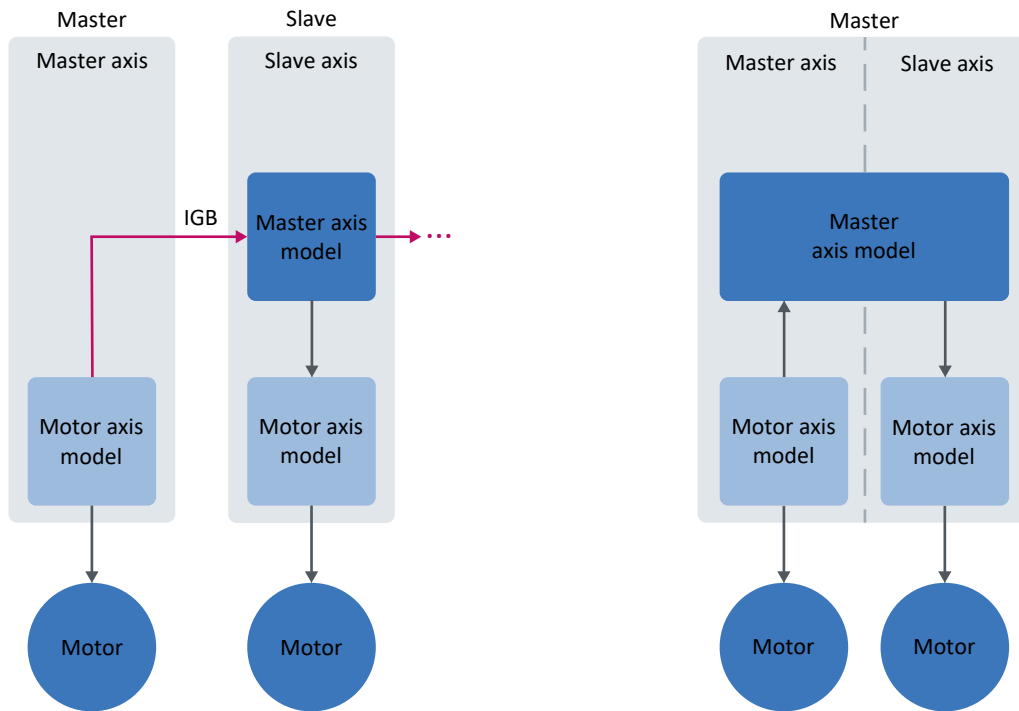


Fig. 5: Synchronous operation: Model 3

The master values are provided by the actual position of the master axis. For double-axis controllers, the second axis follows the master value that is provided by an internal device master axis model. For single-axis controllers, a master drive controller provides the master value over IGB motion bus to other slave drive controllers across devices.

Proceed as follows:

- Parameterize the master:
Scale the master axis model and parameterize the master value source (actual position of the master axis).
- Then, define the master value source for the slave axes.

Information

For double-axis controllers, the wizards for synchronous operation of both axes are coupled with each other. If you configure parameter values for one of the two axes, these are automatically taken over at several places for the second axis. Parameterizations that deviate from this are described in the following step-by-step instructions.

4.5.3.1 Parameterizing the master

Parameterize the master value source as described below. All other necessary values for specifying the actual or set position for this master/slave model are taken from the previously defined motor axis model of the master.

Setting the actual position of the master axis as the master value source

1. Highlight the corresponding drive controller in the project tree and click on the axis which you would like to parameterize as the master in the project menu > Wizard area.
 2. Select the Drive Based Synchronous application wizard > Synchronous operation > Master.
 3. G126 Source IGB producer position:
Select 4: Lageistwert I80.
- ⇒ The actual position of the master axis is projected as the master value source.

4.5.3.2 Parameterizing slave axes

Parameterize the master value source for the slave axes. If you have networked multiple drive controllers with each other over IGB motion bus, scale the master axis models for the slaves as well.

4.5.3.2.1 Setting the master value source for slave axes

Define the master value source for the slave axes as described below.

1. Highlight the relevant drive controller in the project tree and click on the axis which you would like to parameterize as a slave in the project menu > Wizard area.
2. Select the Drive Based Synchronous application wizard > Synchronous operation > Slave.
3. G27 Source master position:
Select 6: IGB motion bus.
4. If you have networked multiple drive controllers with each other over IGB motion bus, repeat the steps for all other slaves in the IGB motion bus network.

4.5.3.2.2 Scaling master axis models

If you have networked multiple drive controllers with each other over IGB motion bus, scale the master axis model for the slaves as described below.

1. Highlight the relevant drive controller in the project tree and click on the axis for which you would like to scale the master axis model in the project menu > Wizard area.
2. Select the Drive Based Synchronous application wizard > Synchronous operation > Master: Axis model, scaling.
3. G30 Position range master:
Define the travel range of the master axis.
4. G40 Circular length master:
If you have selected 1: Endless for G30, enter the master position from which counting begins again at zero.
5. I418 Gear ratio denominator and I417 Gear ratio numerator:
Enter a factor for the gear ratio of the master position and axis movement in the form of a quotient (numerator and denominator).
6. G46 Decimal places:
Enter the number of decimal places for specifying and displaying the master positions.
7. G49 Master measure unit:
Enter the desired unit for the master positions. Any unit of measure can be selected (max. 8 characters).
8. If you have networked multiple drive controllers with each other over IGB motion bus, repeat the steps for all other slaves in the IGB motion bus network.

Information

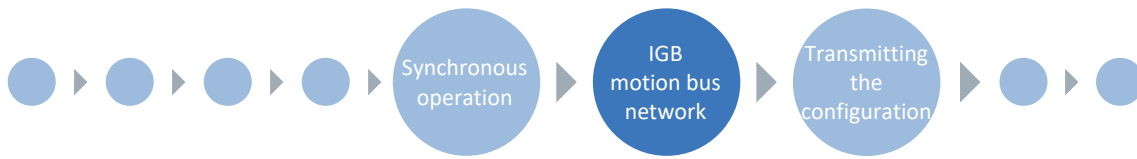
A change to parameter G46 moves the decimal separator for all master position values. Therefore, change G46 before parameterizing other master position values and then check them.

Information

For the synchronous operation of multiple SD6 drive controllers over an IGB motion bus network, the master axis model must be scaled identically for master and slave axes (decimal places, unit of measure, travel range and revolution length, if applicable). Scale the master axis model identically from the start so that you do not have to adapt position settings to changed scaling afterwards.

This always applies to parameters G30, G40, G46 and G49 for a slave axis, and to G30, G40, G46 and G49 (master value source: virtual master or master encoder) or I00, I01, I06 and I09 (master value source: motor axis model of the master) for the master axis.

4.6 Configuring the IGB motion bus network



If you are networking multiple SD6 drive controllers with each other over IGB motion bus, you must project an IGB motion bus network in DriveControlSuite so that the data for synchronous operation can be exchanged between the participating drive controllers. To do so, assign the corresponding drive controllers to the IGB motion bus network and configure the data which is exchanged between drive controllers in the network. Then, transfer the configuration to all drive controllers in the IGB motion bus network.

Further information on the IGB motion bus network can be found in the chapter [IGB motion bus network](#) [▶ 88].

Information

The following chapter is only relevant if you have networked multiple SD6 drive controllers with each other over IGB motion bus. If you are not working with an IGB motion bus network, continue with commissioning by transferring your configuration to the drive controllers as described in the chapter [Transmitting a configuration to drive controllers](#) [▶ 48].

4.6.1 Projecting the IGB motion bus network

Project the IGB motion bus network as described below.

- ✓ You have completed the configuration model for synchronous operation.
- 1. Highlight the module for which you would like to project an IGB motion bus network in the project tree and click on IGB motion bus mapping in the project menu.
 - ⇒ The IGB assignment dialog box opens.
Drive controller structure area: The IGB-capable drive controllers of the selected module are grouped in the IGB network column under Not assigned.
- 2. Click on Add IGB network.
 - ⇒ IGB network column:
A new IGB network **IGB1** is added.
- 3. IGB network column > Not assigned:
 - Drag and drop the drive controllers under the newly created IGB network.
 - ⇒ In the IGB target number column, the number of assigned drive controllers is displayed for **IGB1**.
- 4. Click on Assign IGB addresses automatically.
 - ⇒ IGB address column:
The drive controllers of the IGB network automatically receive an IGB address.
 - ⇒ IGB networks area:
The Mapping and Master/Slave buttons become active once each drive controller has a valid IGB address.
- 5. Scaling column:
 - Make sure that the scaling of the parameter values is consistent within the IGB network.

Information

You can change IGB addresses manually by double-clicking on the IGB address in question and assigning a new, unused value. Numerical values from 0 up to and including 31 are accepted.

4.6.2 Assigning master and slave axes

Assign master and slave axes in the projected IGB motion bus network to each other and check whether the scaling of the axis model of master and slave axes agrees with each other.

Selecting the master

- ✓ You are in the IGB assignment window.
- ✓ You have projected an IGB motion bus network.
- 1. IGB networks area:
Click on the **Master/Slave** button next to the IGB network in question.
 - ⇒ The **Master/slave management window** opens.
- 2. Click on **Select master**.
 - ⇒ The **Select master window** opens.
- 3. Select the master for the IGB motion bus network by activating the option for the corresponding drive controller and confirming with **OK**.
 - ⇒ **Master/slave management window, Master column:**
The selected drive controller is displayed as the master for the IGB network.
- 4. To assign the associated slaves, double-click on the master.
 - ⇒ The **Master/Slave window** opens.
- 5. **Master axis model area:**
Check the settings for the travel range, decimal places and unit of measure for the axis model of the master.

Assigning and checking slave axes

- ✓ You are in the **Master/Slave window**.
- 1. Click on **Add slave axes**.
 - ⇒ The **Add slave axes window** opens.
- 2. Assign slave axes by activating the option for the corresponding drive controllers and confirm with **OK**.
 - ⇒ **Master/Slave window, Slave axes area:**
The assigned slave axes are displayed.
- 3. **G30 Position range master, G46 Decimal places, G49 Master measure unit:**
Check whether the settings for scaling the slave axes agree with those of the master.
- 4. To apply the scaling of the master to all slave axes, click on **Apply master axis scaling for all slave axes**.
- 5. To accept the settings, close the window by clicking on **X**.

Information

For the synchronous operation of multiple SD6 drive controllers over an IGB motion bus network, the master axis model must be scaled identically for master and slave axes (decimal places, unit of measure, travel range and revolution length, if applicable). Scale the master axis model identically from the start so that you do not have to adapt position settings to changed scaling afterwards.

This always applies to parameters G30, G40, G46 and G49 for a slave axis, and to G30, G40, G46 and G49 (master value source: virtual master or master encoder) or I00, I01, I06 and I09 (master value source: motor axis model of the master) for the master axis.

Information

By assigning the slave axes to the master, the parameters G127 IGB position producer and G128 IGB position producer timestamp of the master are written to the parameters G120 Lead position consumer and G121 Lead-position consumer timestamp of the individual slave axes.

4.6.3 Monitoring synchronous operation

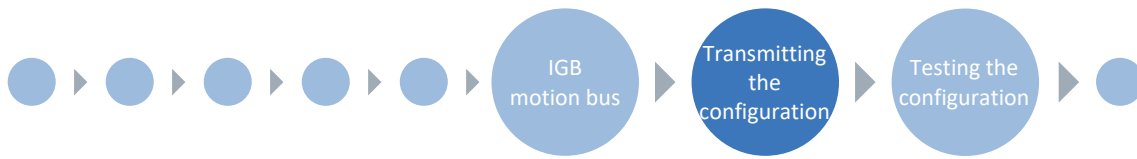
In synchronous operation, you have the option to monitor whether the slave axes are still running synchronously with the master. As soon as a monitored slave axis is no longer synchronous, a quick stop can bring the master to a standstill and thus also stop all slave axes.

- ✓ You have projected an IGB motion bus network.
 - ✓ You have carried out the mapping for synchronous operation.
 - ✓ You are located in the *Master/Slave wizard*.
1. To activate synchronous operation monitoring, select the *Monitoring* option.
 - ⇒ The function is active and the associated parameters are displayed.
 2. *Slave axes area*:
Select which slave axes the master is to monitor for synchronism.
 3. *G108 Auswahl Synchronous Monitoring Schnellhalt*:
Select how the master is to behave if one of the monitored axes is no longer synchronous.
 - 3.1. If a virtual master is acting as the master value source, select 1: *Schnellhalt Virtueller Master*.
 - 3.2. If a master encoder or the axis model of the master axis is acting as the master value source, select 2: *Schnellhalt Motion Kern*.
 4. To accept the settings, close the window by clicking on *X*.

Information

If you subsequently remove a drive controller that is part of monitoring from the module or project, you must configure the settings for synchronous operation monitoring again.

4.7 Transmitting a configuration to drive controllers



Transmit the configuration to the drive controllers and save it in non-volatile memory on the respective device. Then, restart the drive controllers to read in the configuration.

4.7.1 Transmitting and saving the configuration

In order to transmit and save the configuration to one or more drive controllers, you must connect your PC and the drive controllers over the network.

⚠ WARNING!

Injury to persons and material damage due to axis movement!

If there is an online connection between DriveControlSuite and the drive controller, changes to the configuration can lead to unexpected axis movements.

- Only change the configuration if you have visual contact with the axis.
- Make sure that no people or objects are within the travel range.

Information

During the search, all drive controllers within the [broadcast domain](#) are found via [IPv4 limited broadcast](#).

Prerequisites for finding a drive controller in the network:

- Network supports IPv4 limited broadcast
- All drive controllers are in the same subnet (broadcast domain)

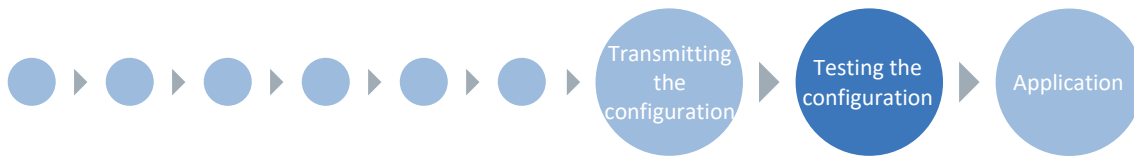
Transmitting the configuration

- ✓ The drive controllers are switched on.
- 1. In the project tree, highlight the module under which you have recorded your drive controller and click **Assignment and live firmware update** in the project menu.
 - ⇒ The **Add connection** window opens. All drive controllers found via IPv4 limited broadcast are displayed.
- 2. **Direct connection** tab > **IP address** column:
Activate the IP addresses in question and confirm your selection with **OK**.
 - ⇒ The **Assignment and live firmware update** window opens. All drive controllers connected through the selected IP addresses are displayed.
- 3. Select the drive controller to which you would like to transfer the configuration. Change the selection of transmission type from **Read** to **Send**.
- 4. Change the selection **Create new drive controller**:
Select the configuration that you would like to transfer to the drive controller.
- 5. Repeat steps 3 and 4 for all other drive controllers to which you would like to transfer your configuration.
- 6. **Online** tab:
Click on **Establish online connections**.
 - ⇒ The configurations are transferred to the drive controllers.

Saving the configuration

- ✓ You have successfully transferred the configurations.
- 1. **Assignment and live firmware update** window:
Click on **Save values (A00)**.
 - ⇒ The **Save values (A00)** window opens.
- 2. Click on **Start action**.
 - ⇒ The configurations are saved.
- 3. Close the **Save values (A00)** window.
- 4. **Assignment and live firmware update** window:
Click on **Restart (A09)**.
 - ⇒ The **Restart (A09)** window opens.
- 5. Click on **Start action**.
- 6. Confirm the safety instruction with **OK**.
 - ⇒ The **Restart (A09)** window closes.
 - ⇒ The fieldbus communication and connection to DriveControlSuite are interrupted.
 - ⇒ The drive controllers restart.

4.8 Testing the configuration



Check the configuration of your projected axis model before parameterizing your application. Use Jog control panel to test the electronic and mechanical data of your project configuration and parameterization for plausibility. Control panel motion gives you the option to test the basic movements of your axis using PLCopen-compliant and manufacturer-specific motion commands.

If you are working with a virtual master or master encoder as the master value source, first reference its absolute position and test its functionality. Then, check the functionality of the master/slave coupling (GearIn).

4.8.1 Testing jog mode

Jog control panel provides various commands for jog mode which you can use to test the configuration of your projected axis model for plausibility.

⚠ WARNING!

Injury to persons and material damage due to axis movement!

By activating the control panel, you have sole control over the movements of the axis using DriveControlSuite. If you are using a controller, it no longer monitors the axis movements after the control panel is activated. The controller cannot intervene to prevent collisions. The controller takes over control again when the control panel is deactivated, which can cause unexpected axis movements.

- Only use the control panel if you have visual contact with the axis.
- Make sure that no people or objects are within the travel range.

Information

If you are using an SD6 drive controller, you can also use jog mode via the operating unit of the SD6.

Information

Make sure that the values of the control panel are compatible with your projected axis model and adjust the default values if necessary. This is the only way to get suitable test results which you can use to optimize your configuration for the respective axis.

The scaling calculator is available under the Axis model wizard >Axis: Scaling to recalculate the values for the control panel according to your projected axis model.

Testing the configuration using the jog control panel

- ✓ The drive controller is switched on.
- ✓ You have established an online connection to the drive controller.
- 1. Highlight the relevant drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
- 2. Select the Jog control panel wizard.
- 3. Click Control panel on and then Enable.
 - ⇒ The axis is monitored using the active control panel.
- 4. Check the default values of the control panel and adjust them to your projected axis model if necessary.
- 5. To test the configuration of your projected axis for direction of motion, velocity, etc., move the axis gradually using the Jog+, Jog-, Jog step+ and Jog step- buttons.
- 6. Use your test results to optimize your configuration as necessary.
- 7. To deactivate the control panel, click on Control panel off.

Information

Jog+ and Jog- cause a continual manual movement in the positive or negative direction. Jog step+ and Jog step- move the axis relative to the current actual position by the increment specified in I14.

Jog+ and Jog- have a higher priority than Jog step+ and Jog step-.

4.8.2 Testing the motion commands

Control panel motion provides a standard set of motion commands which you can use to test the basic movements of your axis. The command set is PLCopen-compliant, supplemented by manufacturer-specific motion commands.

If you are working with absolute position measuring systems and need the MC_MoveAbsolute motion command for testing, you must reference an absolute position in advance (see [Referencing absolute position \[► 27\]](#)).

WARNING!

Injury to persons and material damage due to axis movement!

By activating the control panel, you have sole control over the movements of the axis using DriveControlSuite. If you are using a controller, it no longer monitors the axis movements after the control panel is activated. The controller cannot intervene to prevent collisions. The controller takes over control again when the control panel is deactivated, which can cause unexpected axis movements.

- Only use the control panel if you have visual contact with the axis.
- Make sure that no people or objects are within the travel range.

Information

Make sure that the values of the control panel are compatible with your projected axis model and adjust the default values if necessary. This is the only way to get suitable test results which you can use to optimize your configuration for the respective axis.

The scaling calculator is available under the *Axis model wizard* > *Axis: Scaling* to recalculate the values for the control panel according to your projected axis model.

Testing the configuration using the motion control panel

- ✓ The drive controller is switched on.
 - ✓ You have established an online connection to the drive controller.
1. Highlight the relevant drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
 2. Select the Control panel motion wizard.
 3. Click Control panel on and then Enable.
 - ⇒ The axis is monitored using the active control panel.
 4. Check the default values of the control panel and adjust them to your projected axis model if necessary.
 5. K402 Command:
 - Select the desired command and define the position, velocity, acceleration, deceleration and jerk.
 6. Execute the command by clicking Start.
 7. To test the configuration of your projected axis for direction of motion, velocity, etc., move the axis precisely using the individual commands.
 8. Use your test results to optimize your configuration as necessary.
 9. To deactivate the control panel, click on Control panel off.

4.8.3 Testing synchronous operation

If you are working with a virtual master or master encoder as the master value source, first reference its absolute position and test its functionality. Then, check the functionality of the master/slave coupling (GearIn).

4.8.3.1 Referencing and testing the virtual master

If you are working with a virtual master as the master value source, reference it and test its functionality using the Control panel motion: Virtual master wizard.

Referencing the virtual master

- ✓ The drive controller is switched on.
 - ✓ You have established an online connection to the drive controller.
 - ✓ You have completed and read in the configuration for synchronous operation.
1. Mark the master drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
 2. Select the Drive Based Synchronous application wizard > Additional functions > Virtual master > Control panel motion: Virtual master.
 3. Activate the control panel by clicking Control panel on.
 - ⇒ The control panel is active: The parameters and buttons which you can use to control the virtual master are displayed.
 4. In order to reference the actual position of the virtual master manually, click Set pos.
 - ⇒ The value of parameter G58 Virtual master preset position is transmitted to parameter G190 Actual position virtual master.

Testing the virtual master

- ✓ The drive controller is switched on.
 - ✓ You have established an online connection to the drive controller.
 - ✓ You have completed and read in the configuration for synchronous operation.
1. Mark the master drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
 2. Select the Drive Based Synchronous application wizard > Additional functions > Virtual master > Control panel motion: Virtual master.
 3. Activate the control panel by clicking Control panel on.
 - ⇒ The control panel is active: The parameters and buttons which you can use to control the virtual master are displayed.
 4. To test the virtual master, assign values to the following parameters.
 5. G483 Virtual master command:
Select the motion command you would like to use to move the virtual master.
 6. G484 Virtual master target position:
Define the set position to which the virtual master is to travel.
 7. G485 Virtual master velocity:
Define the set velocity at which the virtual master is to travel.
 8. G486 Virtual master acceleration:
Define the acceleration at which the virtual master is to travel.
 9. G487 Virtual master deceleration:
Define the deceleration at which the virtual master is to travel.
 10. G488 Virtueller Master Ruck:
Define the jerk at which the virtual master is to travel.
 11. To execute the motion command, click on Start.
 - ⇒ The values of parameters G190 Actual position virtual master, G191 Actual velocity virtual master and G159 Virtual master PLCopen-state change according to the parameterization.

Information

The values of Control panel motion: Virtual master for the set velocity, acceleration and jerk of the virtual master are limited using the Limit: Position, velocity, acceleration, jerk for virtual master wizard (parameters G485, G486 and G488 or parameters G50, G51 and G52).

4.8.3.2 Referencing and testing the master encoder

If you are working with an external master encoder as the master value source, reference it and test its function by moving it manually.

- ✓ The drive controller is switched on.
 - ✓ You have established an online connection to the drive controller.
 - ✓ You have completed and read in the configuration for synchronous operation.
1. Mark the master drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
 2. Select the Drive Based Synchronous application wizard > Synchronous operation > Master digital signals: Data source.
 3. G453 Master-Referenz setzen Quelle:
Enter the signal source for triggering the referencing.
 4. To reference the master encoder, trigger the referencing signal.
 5. Select the Drive Based Synchronous application wizard > Synchronous operation > Master.
 6. Move the master encoder mechanically.
- ⇒ The value of parameter G122 Lead-position producer changes according to the movements of the master encoder.

4.8.3.3 Testing the master/slave coupling

Test the master/slave coupling using Control panel motion and the 12: MC_GearIn motion command.

WARNING!

Injury to persons and material damage due to axis movement!

By activating the control panel, you have sole control over the movements of the axis using DriveControlSuite. If you are using a controller, it no longer monitors the axis movements after the control panel is activated. The controller cannot intervene to prevent collisions. The controller takes over control again when the control panel is deactivated, which can cause unexpected axis movements.

- Only use the control panel if you have visual contact with the axis.
- Make sure that no people or objects are within the travel range.

Information

To prevent an enabled axis from moving unnoticed in the background, active control panels are deactivated as soon as you open a new wizard.

To test the master/slave coupling (Gear-In), you need multiple control panels at the same time. Therefore, open all necessary wizards in the first step. In the second step, arrange them next to each other in window mode and only activate the required control panels in the third step.

Opening wizards

1. Highlight the corresponding drive controller in the project tree and click on the master axis in the project menu > Wizard area.
 2. Open the wizard according to the master value source.
 - 2.1. Virtual master:
Select the Drive Based Synchronous application wizard > Additional functions > Virtual master > Jog control panel: Virtual master.
 - 2.2. Master encoder:
Select the Drive Based Synchronous application wizard > Synchronous operation > Master.
 - 2.3. Motor axis model of the master axis:
Select the Drive Based Synchronous application wizard > Jog control panel.
 3. Highlight the corresponding drive controller in the project tree, hold down [CTRL] and click on the first slave axis in the project menu > Wizard area.
 4. Select the Control panel motion wizard.
 5. If you are using a double-axis controller and both axes are running in synchronous operation, repeat the steps for the second axis.
 6. Select Windows in the menu bar > Show windows side by side.
 - ⇒ The wizards are arranged next to each other.
- ⇒ You can now move the master and check the behavior of the slaves at the same time without the control panels being automatically deactivated.

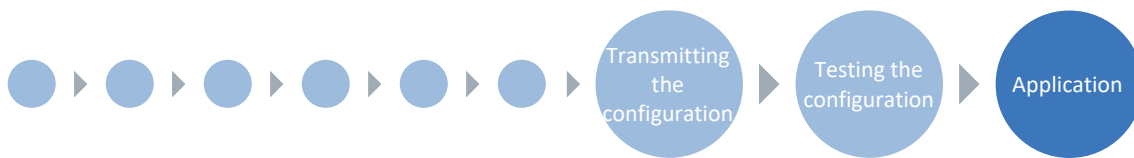
Coupling the slave axis

- ✓ The drive controller is switched on.
 - ✓ You have established an online connection to the drive controller.
 - ✓ You have completed and read in the configuration for synchronous operation.
 - ✓ You have opened all necessary wizards and arranged them next to each other in window mode.
1. Select the previously opened Control panel motion wizard of the slave axis.
 2. Activate the control panel by clicking Control panel on.
 - ⇒ The control panel is active: The parameters and buttons which you can use to move the axis are displayed.
 3. K402 Command:
Select 12: MC_GearIn.
 4. Click on Start.
 - ⇒ Parameter E80 Operating condition indicates the state 29: Synchronous motion.
 - ⇒ The slave axis is active and coupled to the master.
 5. If you are using a double-axis controller and both axes are running in synchronous operation, repeat the steps for the second axis.

Moving the master

- ✓ The drive controller is switched on.
 - ✓ You have established an online connection to the drive controller.
 - ✓ You have completed and read in the configuration for synchronous operation.
 - ✓ You have opened all necessary wizards and arranged them next to each other in window mode.
1. Open the wizard according to the master value source.
 2. Virtual master:
 - 2.1. Select the Jog control panel: Virtual master wizard.
 - 2.2. Activate the control panel by clicking Control panel on.
 - 2.3. Move the virtual master using jog.
 - ⇒ Parameter G190 Actual position virtual master:
The value of the parameter changes according to the position of the virtual master.
 3. Master encoder:
 - 3.1. Select the Master wizard.
 - 3.2. Move the master encoder mechanically.
 - ⇒ Parameter G122 Lead-position producer:
The value of the parameter changes according to the position of the master encoder.
 4. Motor axis model of the master:
 - 4.1. Select the previously opened Jog control panel wizard of the master axis.
 - 4.2. Activate the control panel by clicking Control panel on.
 - 4.3. Move the master axis using jog.
 - ⇒ Parameter I80 Current position:
The value of the parameter changes according to the position of the master axis.
- ⇒ Slave axis, Control panel motion wizard, parameter I80 Current position:
If you move the master, the actual position of the slave axis changes accordingly.

4.9 Parameterizing the Drive Based synchronous application



The Drive Based synchronous application provides two operating modes: Command operating mode and motion block operating mode

The operating modes provide you with a standard, PLCopen-compliant set of motion commands, supplemented with unique motion commands. The operating mode that is best suited to your application depends on your application case. If you are coordinating motion sequences using a controller, use the command operating mode. If you are not using a controller and are instead storing the motion profiles on the respective drive controller, use the motion block operating mode.

The general motion variables that apply to the operating modes in common are grouped in the Data sources wizard. Depending on the operating mode, unique wizards are also available which you can use to parameterize operating mode-specific motion variables. The general or operating mode-specific motion variables you need to parameterize depend on your drive project.

If you would like to use set values from external sources for your project, parameterize them in the first step. Then, continue with the operating mode-specific motion variables in the respective wizards.

Information

A drive controller can only process the motion commands of one operating mode for each projected axis. If you are using a double-axis controller with two projected axes, you must configure the operating mode for each axis individually.

To configure the Drive Based synchronous application, follow the handling instructions for the operating mode in which you want to operate the corresponding axis.

4.9.1 General motion variables and signal sources

First, parameterize the general motion variables and signal sources before you begin parameterization of the operating mode-specific motion variables. The motion variables that you need to parameterize depend on your drive project.

Information

If you are using a double-axis controller with two projected axes, you must configure the shared motion variables and signal sources for each axis individually.

4.9.1.1 Velocity – Parameterizing sources

If you are obtaining the set value for velocity from external sources for applications of the Drive Based synchronous type, enter this as described below.

Either an analog input or a parameter can be used as the source for the set value. If you transmit data via fieldbus, the set value can either be obtained directly from an internal parameter or indirectly by reading out a parameter (such as with [PID controllers](#)).

Set velocity - Parameterizing the source

If you are obtaining the set value for velocity from an external source, enter the source for the main set value.

✓ You use the motion command MC_MoveSpeed or MC_MoveVelocity.

1. Select the Drive Based synchronous application wizard > Data sources > External velocity: Data source.
2. G461 Source external velocity:
 - 2.1. If an analog input acts as the source, select the corresponding input.
 - 2.2. If you are transmitting data via fieldbus and would like to configure the value as the general data source, select 4: Parameter G460.
 - 2.3. If the set value is read out indirectly via a parameter, select 5: Indirect read parameter G811.
3. G460 External velocity:

If you have selected 4: Parameter G460 for G461, enter the set velocity here.
4. G811 Indirect read external velocity:

If you have selected 5: Indirect read parameter G811 for G461, enter the associated parameter coordinate here.

Additional velocity – Parameterizing the source

If you also want to regulate the set velocity, enter the source for the additional set value.

- ✓ You use the motion command MC_MoveSpeed or MC_MoveVelocity.
 - ✓ You obtain the set velocity from an external source.
1. Select the Drive Based synchronous application wizard > Data sources > External additional velocity: Data source.
 2. G464 Source external additional velocity:
 - 2.1. If an analog input acts as the source, select the corresponding input.
 - 2.2. If you are transmitting data via fieldbus and would like to configure the value as the general data source, select 4: Parameter G463.
 - 2.3. If the set value is read out indirectly via a parameter, select 5: Indirect read parameter G812.
 3. G463 External additional velocity:

If you have selected 4: Parameter G463 for G464, enter the set velocity here.
 4. G812 Indirect read additional external velocity:

If you have selected 5: Indirect read parameter G812 for G463, enter the parameter coordinate here.

4.9.1.2 Velocity override – Parameterizing the source

If you would like to use a velocity override to scale the velocity profile of your motion commands, enter the source for the velocity override.

Either an analog input or a parameter can be used as the source for the set value. If you transmit data via fieldbus, the set value can either be obtained directly from an internal parameter or indirectly by reading out a parameter (such as with PID controllers).

1. Select the Drive Based synchronous application wizard > Data sources > Velocity override: Data source.
2. G467 Quelle velocity override:
 - 2.1. If an analog input acts as the source, select the corresponding input.
 - 2.2. If you are transmitting data via fieldbus and would like to configure the value as the general data source, select 4: Parameter G466.
 - 2.3. If the set value is read out indirectly via a parameter, select 5: Indirect read parameter G813.
3. G466 Velocity override:

If you have selected 4: Parameter G466 for G467, enter the set velocity here.
4. G813 Indirect read velocity override:

If you have selected 5: Indirect read parameter G813 for G467, enter the parameter coordinate here.

4.9.1.3 Torque/force – Parameterizing sources

If you have selected the motion command MC_TorqueControl for Drive Based synchronous type applications and obtain set values from external sources, enter the sources for the set torque/force and velocity bracketing.

Either an analog input or a parameter can be used as the source for the set value. If you transmit data via fieldbus, the set value can either be obtained directly from an internal parameter or indirectly by reading out a parameter (such as with PID controllers).

Set torque/force - Parameterizing the source

If you obtain the set value for torque/force from an external source, enter the source for the set torque/force.

1. Select the Drive Based synchronous application wizard > Data sources > Set torque/force, velocity bracketing: Data source.
2. G470 Source torque/force reference:
 - 2.1. If an analog input acts as the source, select the corresponding input.
 - 2.2. If you are transmitting data via fieldbus and would like to configure the value as the general data source, select 4: Parameter G469.
 - 2.3. If the set value is read out indirectly via a parameter, select 5: Indirect read parameter G814.
3. G469 Torque/Force reference:
If you have selected 4: Parameter G469 for G470, enter the value for set torque/force here. The value relates to C03 Maximum positive M/F and C05 Maximum negative M/F.
4. G814 Indirect read torque/force reference:
If you have selected 5: Indirect read parameter G814 for G470, enter the associated parameter coordinate here.

Velocity bracketing – Parameterizing the source

If you obtain values for limiting the velocity from an external source, enter the source for the velocity bracketing.

1. Select the Drive Based synchronous application wizard > Data sources > Set torque/force, velocity bracketing: Data source.
2. G473 Source velocity bracketing positiv:
 - 2.1. If an analog input acts as the source, select the corresponding input.
 - 2.2. If you are transmitting data via fieldbus, select 4: Parameter G472.
3. G472 Velocity bracketing positiv:
If you have selected 4: Parameter G472 for G473, enter the maximum limit that you want to permit for your mechanical system.
4. G476 Source velocity bracketing negative:
 - 4.1. If an analog input acts as the source, select the corresponding input.
 - 4.2. If you are transmitting data via fieldbus, select 4: Parameter G475.
5. G475 Velocity bracketing negative:
If you have selected 4: Parameter G475 for G476, enter the maximum limit that you want to permit for your mechanical system here.

Information

If you define the velocity bracketing using G472 or G475, you should allow the control enough room. Therefore, select a value for the velocity bracketing that is greater than 1%. G472 and G475 refer to I10 Maximal speed.

4.9.2 Parameterizing motion block operating mode

The following graphic shows the signal flows of motion block operating mode. The elements shown in a lighter shade are optional.

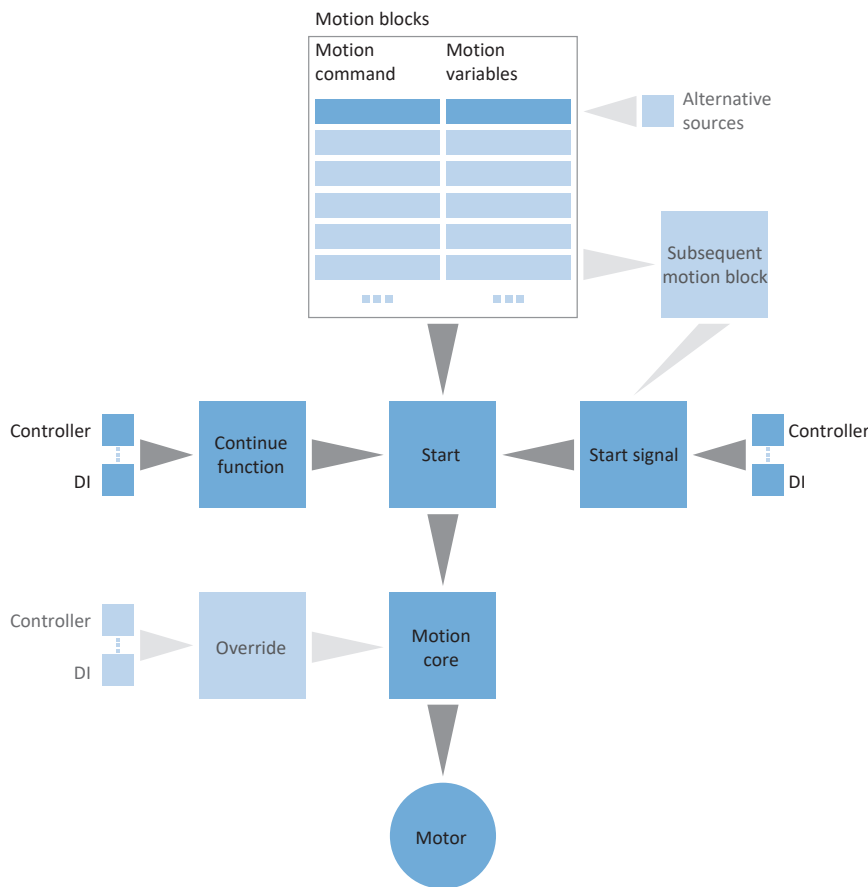


Fig. 6: Motion block operating mode: Signal flows

In the motion block operating mode, motion profiles are permanently stored on the drive controller in the form of motion blocks. A motion block includes a PLCopen-compliant motion command which defines the basic motion method of the axis. The associated motion variables supply specifications on velocity, acceleration, deceleration and jerk, which the motion core of the drive controller uses to calculate the motion profile.

Motion blocks can be selected individually using motion block numbers and they can be linked with each other, i.e. they are automatically processed in succession. In motion block linking, a linear, defined sequence as well as branches based on conditions are both possible for the further motion block action. You can parameterize specifications on velocity override and brake for each motion block individually.

In motion block operating mode, drive controllers are either activated using a fieldbus or by a mixed operation of a fieldbus and terminals. The number of axes and motion blocks used determines the memory you need for your project.

Proceed as follows ...

- Activate motion block operating mode.
- Parameterize one or more motion blocks.
- Test the individual motion blocks using the motion block control panel.
- If you want to link motion blocks with each other, define subsequent motion blocks and triggers for continuation.
- Test the motion block links using the motion block control panel.
- Define how the motion blocks are started.
- Parameterize the source for the continue, reset and, if necessary, stop signals.

Information

The motion variables available for this operating mode that you configure in the software depend on your application case as well as on additional project-specific factors such as the use of a controller or the type of data transmission (fieldbus, terminals).

4.9.2.1 Activating motion block operating mode

1. Highlight the relevant drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
 2. Select the Drive Based synchronous application wizard.
 3. Operating modes area:
Activate the option for motion block operating mode.
- ⇒ The associated wizards are shown and are available for parameterizing the operating mode.

4.9.2.2 Limiting the torque/force using the operating mode

Define whether you also want to limit torque/force using the operating mode. Limiting using the mechanical system takes place in the project configuration of the axis model. You have the option of specifying an additional limit for all motion blocks together or for each motion block individually.

1. Select the Drive Based synchronous application wizard > Motion block operating mode > Motion block > Torque/force limit: Data source.
2. J08 Source maximum M/F:
 - 2.1. If you would like to specify the maximum value from the axis model for all motion blocks centrally, select 3: 100%.
 - 2.2. If you would like to specify the maximum value using the operating mode for all motion blocks centrally, select 1: Parameter J09.
 - 2.3. If you would like to define an individual maximum value for each motion block, select 2: Parameter J25.
3. J09 Maximum torque/force:
If you have selected 1: Parameter J09 for J08, enter the maximum permitted value for torque/force.
4. If you have selected 2: Parameter J25 for J08, enter the maximum permitted value for torque/force during the parameterization of individual motion blocks in J25 Maximum torque/force.
To do so, proceed as described in the chapter [Parameterizing the motion block](#) [▶ 64].

Information

The mechanical limit for torque/force is executed by C03 Maximum positive M/F and C05 Maximum negative M/F using the axis model. The additional limit using the operating mode is optional and refers to the values that you defined in the axis model.

4.9.2.3 Parameterizing motion blocks

Before you start parameterizing the individual motion blocks, determine whether you also want to limit torque/force using the operating mode. You have already defined the maximum value for torque/force in the axis model, but you have the option of specifying an additional limit for all motion blocks together or for each motion block individually.

For each motion block, select an appropriate motion command and parameterize the associated motion variables. You can copy parameterized motion blocks completely to create other motion blocks.

Then, test the parameterization of the individual motion blocks using [Motion block control panel](#).

Information

Before you start parameterizing the operating mode-specific motion variables, parameterize the general motion variables and signal sources. If you obtain set values for velocity, velocity override or torque/force for your drive project from external sources, proceed as described in the chapter [General motion variables and signal sources](#) [► 59].

4.9.2.3.1 Parameterizing the motion block

First, select the motion command that is to be executed. Depending on the motion command, various motion variables are available for parameterization. Parameterize the motion variables according to your drive project.

- ✓ You have activated motion block operating mode.
- 1. Select the Drive Based synchronous application wizard > Motion block operating mode.
 - ⇒ By default, 16 motion blocks are available to you which you can parameterize individually.
- 2. To parameterize the first motion block, click its motion block number in the Number column.
 - ⇒ The detail view for the first motion block is displayed.
- 3. J10 Name:
 - Give the motion block a unique name.
- 4. J11 Command:
 - Select the motion command that is to be executed.
 - ⇒ The associated parameters for the selected motion command are displayed.
- 5. J25 Maximum torque/force:
 - If you have also limited torque/force using the operating mode and have selected 2: Parameter J25 for J08, enter the maximum permitted value for torque/force for each motion block.
- 6. Parameterize the selected motion command based on its associated parameter.

Information

Use J28 Velocity override enable to scale the velocity profile of a motion block. For motion blocks with final velocity, the velocity override influences J14 Velocity and J18 End velocity equally, i.e. the relationship between J14 and J18 remains the same.

Information

Use J21 Command Mode to define how the motion block, as the subsequent motion block, is executed for motion block linking. If you select 0: Abort current command, the ongoing motion block is interrupted when the trigger is activated and the subsequent motion block is executed immediately. If you select 1: Start when current command has finished, the ongoing motion block is completed first before the subsequent motion block is started.

Moving the axis with a set velocity

- ✓ You have selected the motion command MC_MoveSpeed or MC_MoveVelocity.
- 1. J30 Source velocity:
Select the source from which you will obtain the set value for the velocity.
 - 1.1. If you obtain the set value via fieldbus, select 0: Parameter J14.
 - 1.2. If you obtain the set value from an external source, select 1: Parameter G462.
- 2. J14 Velocity:
If you have selected 0: Parameter J14 for J30, enter the set velocity.
- 3. J15 Acceleration, J16 Deceleration, J17 Jerk:
Enter the set values for acceleration, deceleration and jerk.
- 4. J28 Velocity override enable:
If you would like to apply a velocity override to the motion block, select 1: Active.
- 5. J21 Command Mode:
Select the mode in which the selected motion block, as the subsequent motion block, is to be executed.

Moving the axis with a set torque/force

- ✓ You have selected the motion command MC_TorqueControl.
- 1. J32 Source reference torque/force:
Select the source from which you will obtain the set value for torque/force.
 - 1.1. If you obtain the set value via fieldbus, select 0: Parameter J26.
 - 1.2. If you obtain the value from external sources, select 1: Parameter G471.
- 2. J26 Torque/Force reference:
If you have selected 0: Parameter J26 for J32, enter the set torque or set force.
- 3. J16 Deceleration, J17 Jerk:
Enter the set values for deceleration and jerk.
- 4. J28 Velocity override enable:
If you would like to apply a velocity override to the motion block, select 1: Active.
- 5. J21 Command Mode:
Select the mode for the motion block in which it, as the subsequent motion block, is to be executed.

Moving the axis to the set position

- ✓ You have selected the motion command MC_MoveAbsolute, MC_MoveAdditive or MC_MoveRelative.
1. J13 Position:
Enter the absolute or relative position to which the axis is to be moved.
 2. J14 Velocity:
Enter the set value for the velocity.
 3. J15 Acceleration, J16 Deceleration, J17 Jerk:
Enter the set values for acceleration, deceleration and jerk.
 4. J18 End velocity:
Enter the final velocity with which you want to reach the set position J13.
 5. J24 Waiting time at the end:
If a subsequent motion block is not to be started immediately, enter the wait time at the end of the motion block.
 6. J27 Engage brake at the end:
If you have parameterized a brake and it is to engage at the end of the motion block, select 1: Active.
 7. J28 Velocity override enable:
If you would like to apply a velocity override to the motion block, select 1: Active.
 8. J21 Command Mode:
Select the mode for the motion block in which it, as the subsequent motion block, is to be executed.

Information

For motion blocks with a final velocity that is not 0, a change of travel direction by the subsequent motion block is not permitted.

Referencing the axis

- ✓ You have selected the motion command MC_Home.
1. I30 Referencing type:
Indicates what referencing type you have parameterized for the referencing process.
 2. J16 Deceleration, J17 Jerk:
Enter the set values for deceleration and jerk with which the axis is stopped if referencing is interrupted by a control panel.
 3. J24 Waiting time at the end:
If a subsequent motion block is not to be started immediately, enter the wait time at the end of the motion block.
 4. J27 Engage brake at the end:
If you have parameterized a brake and it is to engage at the end of the motion block, activate J27.
 5. J28 Velocity override enable:
If you have parameterized a velocity override and would like to apply it to the motion block, select 1: Active.

Information

To parameterize the referencing process, click the [Referencing](#) link and proceed as described in the chapter [Referencing absolute position](#) [▶ 27].

Stopping the axis

- ✓ You have selected the motion command MC_Halt or MC_Stop.
- 1. J16 Deceleration, J17 Jerk:
Enter the set values for deceleration and jerk with which the axis is stopped.
- 2. J24 Waiting time at the end:
If a subsequent motion block is not to be started immediately, enter the wait time at the end of the motion block.
- 3. J27 Engage brake at the end:
If you have parameterized a brake and it is to engage at the end of the motion block, activate J27.

Move the axis synchronously

- ✓ You have selected the motion command MC_GearIn or MC_GearInPosition.
- 1. J15 Acceleration, J16 Deceleration, J17 Jerk:
Enter the set values for acceleration, deceleration and jerk.
- 2. J13 Position:
If you have selected MC_GearInPosition, enter the position of the axis from which the axis is to move synchronously with the master.
- 3. J20 Position D (Master Sync Position):
If you have selected MC_GearInPosition, enter the position of the master from which the axis is to move synchronously with the master.
- 4. J28 Velocity override enable:
If you would like to apply a velocity override to the motion block, select 1: Active.

Information

To decouple a slave axis from synchronous operation, you can use the motion commands MC_GearOut or MC_Halt, for example. MC_GearOut decouples the axis and retains the last velocity before decoupling; MC_Halt brings the axis to a standstill after decoupling.

4.9.2.3.2 Parameterizing the motion block duration

If you are using a continuous motion command to move the axis endlessly, for example, with a set velocity or set torque/force, you can switch to a subsequent motion block after a defined time using the additional motion block duration function and an indirect trigger.

- ✓ You are using an endless motion command.
1. Select the Drive Based synchronous application wizard > Motion block operating mode > Subsequent motion block > Indirect trigger: Data source.
 2. Activate the Motion block duration option.
 - ⇒ The additional function is activated and the associated parameter J452 FSM Loop Trigger Delay is shown in the Motion block wizard.
 3. Click on the Motion block link.
 - ⇒ The Motion block wizard opens.
 4. Number:

Use the arrow keys to select the number of the motion block for which you would like to define a motion block duration.
 5. J452 FSM Loop Trigger Delay:

Enter the duration for which the motion block is to be executed.

Information

Once the motion block duration is reached, the information is stored in bit 0 of parameter J453 FSM Loop Trigger Output. To continue on to a subsequent motion block once the motion block duration is reached, parameterize a subsequent motion block using the interrupt signal with an indirect trigger as described in [Parameterizing the subsequent motion block with a trigger \[► 71\]](#).

4.9.2.3.3 Copying a motion block

To make parameterization of similar motion blocks easier, you can apply the values from existing motion blocks.

- ✓ You are in the Motion block wizard.
 - ✓ You have parameterized at least one motion block.
1. Number:

Select the number of the motion block to which you want to copy the values from an existing motion block.
 2. Apply from:

Click on **Apply from** and select the existing motion block whose values you want to copy.
 3. Confirm by clicking on **OK**.
 - ⇒ The values of the existing motion block are applied to the current motion block.
 4. Adjust the copied values, such as J10 Name, if necessary.

4.9.2.3.4 Testing the motion block parameterization

Motion block control panel provides all functions you need to test the parameterization of individual motion blocks.

WARNING!

Injury to persons and material damage due to axis movement!

By activating the control panel, you have sole control over the movements of the axis using DriveControlSuite. If you are using a controller, it no longer monitors the axis movements after the control panel is activated. The controller cannot intervene to prevent collisions. The controller takes over control again when the control panel is deactivated, which can cause unexpected axis movements.

- Only use the control panel if you have visual contact with the axis.
- Make sure that no people or objects are within the travel range.

-
- ✓ The drive controller is switched on.
 - ✓ You have established an online connection to the drive controller.
 - ✓ You have parameterized at least one motion block.
1. Select the Drive Based synchronous application wizard > Motion block operating mode > Motion block control panel.
 2. Click Control panel on and then Enable.
 - ⇒ The drive is controlled using the active control panel.
 3. J311 Motion block selection:
Select the motion block whose parameterization you would like to test.
 4. Test the positions, velocities, accelerations and brakes and optimize your parameterization based on your test results if necessary.
 5. If you want to link motion blocks with each other as subsequent motion blocks, first test the parameterization for each motion block individually to be able to rule out errors in motion block linking.
 6. In order to deactivate the control panel, click Control panel off.

4.9.2.4 Parameterizing motion block linking

In order to configure rolling programs automatically, you can connect multiple motion blocks with each other in sequence. To do so, link the motion blocks in question with each other as subsequent motion blocks and enter the associated signal sources for continuation. The subsequent motion blocks that are assigned to a motion block are displayed in the motion block overview.

Then, test the linking of the individual motion blocks using [Motion block control panel](#).

4.9.2.4.1 Parameterizing subsequent motion blocks

You can define one or more possible subsequent motion blocks for each motion block. Continuation to the subsequent motion block either happens automatically at the end of the motion block or is caused by a trigger. The trigger can either be a direct trigger signal or a trigger condition that is configurable in the software (indirect trigger).

For subsequent motion blocks with triggers, a distinction is made between subsequent motion blocks that are triggered at the end of the previous motion block and subsequent motion blocks that interrupt the previous motion block when an interrupt signal is triggered. The continuation can be executed by a direct or indirect trigger in either case.

Subsequent motion blocks at the end of a motion block are triggered as soon as the set position of a finite motion command is reached and the trigger is active. For subsequent motion blocks due to interruption, a finite or endless motion is interrupted and continued in the subsequent motion block once the trigger is active. For example, the ongoing motion can be interrupted when a sensor or light barrier is triggered and the axis stopped.

Buffered subsequent motion blocks are parameterized without triggers and are intended for the case where the axis is moved to a set position in the previous motion block that is reached with the final velocity. In order to continue the motion fluidly and start the next motion command without a delay or standstill, the subsequent motion block is loaded in advance (buffered) while the previous motion block is still being executed. The continuation happens automatically at the end of the previous motions block.

4.9.2.4.1.1 Parameterizing the subsequent motion block with a trigger

You can define up to five subsequent motion blocks that are each activated by a trigger signal or trigger condition (indirect trigger) for each motion block in the *Subsequent motion block wizard*. Depending on the configuration, the subsequent motion block is only executed at the end of the previous motion block, or the ongoing motion block is interrupted when an interrupt signal is triggered or an interrupt condition occurs and the subsequent motion block is started.

For each subsequent motion block, define the signal sources for continuation by parameterizing either a direct trigger signal or an indirect trigger condition. You can configure multiple triggers for each motion block. Each trigger can either activate a subsequent motion block through interruption or a subsequent motion block at the end of the previous motion block. If multiple triggers are active simultaneously, the sequence continues with the subsequent motion block with the highest priority (A – E).

Information

Depending on the application case, you can either parameterize a regular subsequent motion block or a buffered subsequent motion block at the end of the previous motion block. The use of a regular subsequent motion block at the end rules out the simultaneous use of a buffered subsequent motion block for the same motion block, and vice versa. You can combine subsequent motion blocks through interruption with regular subsequent motion blocks at the end and with buffered subsequent motion blocks.

Information

J450 Source trigger and J439 Source indirect trigger are array parameters. The first element (J450[0], J439[0]) corresponds to trigger 1, the second element (J450[1], J439[1]) corresponds to trigger 2, etc.

Parameterizing the subsequent motion block at the end of motion block

Use subsequent motion blocks at the end of the motion block if you would like to move the axis with MC_MoveAbsolute, MC_MoveRelative or MC_MoveAdditive and continue to the subsequent motion block upon reaching the specified set position.

The following section explains the process as an example, based on the parameterization of subsequent motion block A. For the parameterization of subsequent motion block B – E, proceed in the same manner.

- ✓ You have parameterized at least two motion blocks.
 - ✓ You are moving the axis with MC_MoveAbsolute, MC_MoveRelative or MC_MoveAdditive in the motion block to which you would like to add a subsequent motion block.
1. Select the Drive Based synchronous application wizard > Motion block operating mode > Subsequent motion block.
 2. Number:
Use the arrow keys to select the number of the motion block to which you would like to add a subsequent motion block.
 3. J281 Next motion block A at end:
If the subsequent motion block is to be executed once the motion block selected in Number has been completed, select the subsequent motion block here.
 4. J251 Source trigger next motion block A:
Select the trigger that activates continuation to the subsequent motion block.
 5. Click on the Trigger: Source link.
⇒ The Trigger: Data source wizard opens.
 6. J450[0] Source trigger:
 - 6.1. If you would like the subsequent motion block to be activated directly at the end of the previous motion block, select 1: High.
 - 6.2. If a digital input (direct or inverted) acts as the signal source for the start of the subsequent motion block, select the corresponding input.
 - 6.3. If the subsequent motion block is to be executed when a configurable condition is fulfilled (indirect trigger), select 2: Parameter and proceed as described in the section **Parameterizing the indirect trigger**.

Information

For subsequent motion blocks at the end of the motion block, the trigger condition is only evaluated at the end of the ongoing motion block by using J281 – J285. Since the evaluation only takes place at the motion block end, J21 Command Mode does not influence when the subsequent motion block is started. The subsequent motion block always starts at the motion block end if the trigger condition is active.

Parameterizing the subsequent motion block due to interruption

Use subsequent motion blocks due to interruption if you would like to interrupt the ongoing motion block when the trigger is activated and continue to the subsequent motion block.

The following section explains the process as an example, based on the parameterization of subsequent motion block A. For the parameterization of subsequent motion block B – E, proceed in the same manner.

- ✓ You have parameterized at least two motion blocks.
 - ✓ You have selected 0: Abort current command for the subsequent motion block in J21 Command Mode.
1. Select the Drive Based synchronous application wizard > Motion block operating mode > Subsequent motion block.
 2. Number:
Use the arrow keys to select the number of the motion block to which you would like to add a subsequent motion block.
 3. J271 Next motion block A by interrupt:
If the subsequent motion block is to be executed as soon as the motion selected in Number is interrupted by an interrupt signal, select the subsequent motion block here.
 4. J251 Source trigger next motion block A:
Select the trigger that activates continuation to the subsequent motion block.
 5. Click on the Trigger: Source link.
⇒ The Trigger: Data source wizard opens.
 6. J450[0] Source trigger:
 - 6.1. If a digital input (direct or inverted) acts as the signal source for the start of the subsequent motion block, select the corresponding input.
 - 6.2. If the subsequent motion block is to be executed when a configurable condition is fulfilled (indirect trigger), select 2: Parameter and proceed as described in **Parameterizing the indirect trigger**.

Information

For subsequent motion blocks due to interruption, the trigger condition is evaluated while the previous motion is still running by using J271 – J275. J21 Command Mode controls when the triggered subsequent motion block is started. When 0: Abort current command is selected, the subsequent motion block is executed immediately once the trigger condition is active. When 1: Start when current command has finished is selected, the subsequent motion block is pre-loaded (buffered) upon activation of the trigger condition, but only executed at the motion block end if the previous motion block involves a finite motion.

Information

For motion blocks with a final velocity, you can parameterize a special case: a buffered subsequent motion block with trigger condition. To do so, use a subsequent motion block due to interruption (J271 – J275) with J21 Command Mode = 1: Start when current command has finished.

When 1: Start when current command has finished is selected, the subsequent motion block is pre-loaded (buffered) upon activation of the trigger condition, but only executed at the motion block end if the previous motion block involves a finite motion. This enables a fluid transition to the subsequent motion block, depending on a trigger condition.

Parameterizing the indirect trigger

Parameterize the trigger condition for motion block continuation. If you parameterize a subsequent motion block that is to be triggered at the end of the previous motion block, the condition for continuation must be fulfilled before the set position is reached.

✓ You have selected 2: Parameter for J450[0] Source trigger.

1. Select the Drive Based synchronous application wizard > Motion block operating mode > Subsequent motion block > Indirect trigger: Data source.

2. J439[0] Source indirect trigger:

Activate the desired indirect trigger and enter the condition for continuation to subsequent motion block A.

2.1. If a simple parameter is used as the condition for continuation, select the BOOL notation: `Coordinate`

2.2. If a certain bit of a simple parameter is used as the condition for continuation, select the bit notation: `Coordinate.Bit`

2.3. If an array or structure parameter is used as the condition for continuation, select the array notation: `Coordinate[Element]`

2.4. If a certain bit of an array or structure parameter is used as the condition for continuation, select the following notation: `Coordinate[Element].Bit`

Information

Possible values for J439 Source indirect trigger are the entry of parameters, individual bits of the status or control word or signals of additional functions such as motion block duration, comparator or cams.

If you are using a double-axis controller, make sure you specify the correct axis when entering the trigger condition: `1stCoordinate` or `2ndCoordinate`. They can only access the parameters of their own axis.

Notation	Input format	Example
BOOL notation	<code>Coordinate</code>	J439[0] = I86 The subsequent motion block is only triggered if I86 In reference is fulfilled, i.e. if the axis is referenced.
Bit notation	<code>Coordinate.Bit</code>	J439[0] = J453.0 You use J452 FSM Loop Trigger Delay and the triggering of the subsequent motion block is to be time-controlled. The information "Motion block duration is reached" is stored in bit 0 of J453 FSM Loop Trigger Output.
Array notation	<code>Coordinate[Element]</code>	J439[0] = I64[0] The subsequent motion block is to be triggered once the first cam position is reached. The information for the first cam is stored in array element 0 of I64 Cam result.
Combined notation	<code>Coordinate[Element].Bit</code>	–

Tab. 4: Indirect trigger: Notations

4.9.2.4.1.2 Parameterizing the buffered subsequent motion block

In the *Subsequent motion block wizard*, you can define a buffered subsequent motion block for each motion block. This means that the subsequent motion block is already loaded while the previous motion block is still being executed. Continuation therefore takes place automatically at the end of the previous motion block, without a delay and without a trigger signal.

Information

Depending on the application case, you can either parameterize a regular subsequent motion block or a buffered subsequent motion block at the end of the previous motion block. The use of a regular subsequent motion block at the end rules out the simultaneous use of a buffered subsequent motion block for the same motion block, and vice versa. You can combine subsequent motion blocks through interruption with regular subsequent motion blocks at the end and with buffered subsequent motion blocks.

Parameterizing the buffered subsequent motion block

Use a buffered subsequent motion block if you are moving the axis in the previous motion block with a set position (MC_MoveAbsolute, MC_MoveRelative, MC_MoveAdditive) and would like to transition fluidly into the next motion upon reaching the set position with the final velocity.

- ✓ You have parameterized at least two motion blocks.
 - ✓ You are moving the axis in the previous motion block with MC_MoveAbsolute, MC_MoveRelative or MC_MoveAdditive.
 - ✓ You have selected 1: Start when current command has finished for the motion block intended as the subsequent motion block in J21 Command Mode.
1. Select the Drive Based synchronous application wizard > Motion block operating mode > Subsequent motion block.
 2. Number:
Use the arrow keys to select the number of the motion block to which you would like to add a buffered subsequent motion block.
 3. J286 Buffered motion block:
Select the motion block that is to follow the motion block selected in *Number* without a delay or trigger signal.

Information

If you add a buffered subsequent motion block using J286 Buffered motion block, make sure that J281 Next motion block A at end up to J285 Next motion block E at end is set to -1: Inactive.

Information

For motion blocks with a final velocity, you can parameterize a special case: a buffered subsequent motion block with trigger condition. To do so, use a subsequent motion block due to interruption (J271 – J275) with J21 Command Mode = 1: Start when current command has finished.

When 1: Start when current command has finished is selected, the subsequent motion block is pre-loaded (buffered) upon activation of the trigger condition, but only executed at the motion block end if the previous motion block involves a finite motion. This enables a fluid transition to the subsequent motion block, depending on a trigger condition.

4.9.2.4.2 Testing the motion block linking

Motion block control panel provides all functions that you need to test the motion block linking.

WARNING!

Injury to persons and material damage due to axis movement!

By activating the control panel, you have sole control over the movements of the axis using DriveControlSuite. If you are using a controller, it no longer monitors the axis movements after the control panel is activated. The controller cannot intervene to prevent collisions. The controller takes over control again when the control panel is deactivated, which can cause unexpected axis movements.

- Only use the control panel if you have visual contact with the axis.
- Make sure that no people or objects are within the travel range.

-
- ✓ The drive controller is switched on.
 - ✓ You have established an online connection to the drive controller.
 - ✓ You have parameterized at least one motion block.
1. Select the Drive Based synchronous application wizard > Motion block operating mode > Motion block control panel.
 2. Click Control panel on and then Enable.
 - ⇒ The drive is controlled using the active control panel.
 3. J311 Motion block selection:
Select the motion block whose linking with the subsequent motion block you would like to check.
 4. Test the positions, velocities, accelerations and brakes and optimize your parameterization based on your test results if necessary.
 5. Repeat the tests for each additional motion block for which you have defined subsequent motion blocks.
 6. In order to deactivate the control panel, click Control panel off.

4.9.2.5 Parameterizing the type of motion block

Define how the motion blocks are to be started. Motion block operating mode offers the option to parameterize the direct start of up to three motion blocks without a controller. If you would like to use a controller, parameterize the start signal by defining the source for the execute signal and the source for the selection of the motion block to be started.

4.9.2.5.1 Parameterizing the direct motion block type

Motion block operating mode offers the option to parameterize the direct start of up to three motion blocks of your choice without a controller.

1. Select the Drive Based synchronous application wizard > Motion block operating mode > Motion block > Direct motion block start: Data source.
2. J80 Activation direct motion block start:
In order to activate the direct motion block start, select 1: Active.
3. J87 Selected direct motion block start 1:
Enter the number of the motion block that you would like to start directly.
4. J81 Source direct motion block start 1:
 - 4.1. If a digital input (direct or inverted) acts as the signal source for the start of the motion block, select the corresponding input.
 - 4.2. If you would like to activate the motion block directly, select 2: Parameter J84.
5. J84 Direct motion block start 1:
If you have selected 2: Parameter J84 for J81 and want to start the motion block directly from DriveControlSuite, select High.
6. Configure the direct start of a second and third motion block in the same way as the first motion block, if necessary.
 - 6.1. If you want to parameterize the direct start for a second motion block, repeat the steps in the same way for J88, J82 and J85.
 - 6.2. If you want to parameterize the direct start for a third motion block, repeat the steps in the same way for J89, J83 and J86.

Information

If you have configured the direct start of multiple motion blocks, they are activated with a simultaneous start signal according to your priority: J83 Source direct motion block start 3 > J82 Source direct motion block start 2 > J81 Source direct motion block start 1.

4.9.2.5.2 Parameterizing the start signal

If you want to use a controller and start the motion block using an execute signal, i.e. a start signal, define the source for the start signal and the source for the selection of the motion block that is to be started by the start signal.

Defining the source for the start signal

1. Select the Drive Based synchronous application wizard > Data sources > Application digital signals: Data source.
2. I100 Source execute:
 - 2.1. If a digital input (direct or inverted) acts as the signal source, select the corresponding input.
 - 2.2. If the control word I210 of the application acts as the signal source, select 2: Parameter.
Bit 0 of the control word is set as the data source.
Note that individual bits of the control word are transmitted only if the control word is part of the process data mapping.

Defining the source for the motion block selection

1. Select the Drive Based synchronous application wizard > Motion block operating mode > Motion block > Motion block selection: Data source.
2. J03 Source motion block selection:
 - 2.1. If the control word J02 of motion block operating mode acts as the data source, select 1: Parameter J02 and enter the number of the motion block to be started in the event of an execute signal in J02Reference motion block.
 - 2.2. If digital inputs act as the value source, select 2: Digital inputs and enter the digital inputs (direct or inverted) that activate a controller or binary switch.

4.9.2.6 Parameterizing the continue, reset and halt signal

A motion block can be interrupted during its run, such as by an Enable-off or due to the triggering of the safety technology. In this case, the software checks whether a resumption of the motion block (continue) is possible afterward. The continue function makes it possible to continue and complete the interrupted motion block. If a resumption of the motion block is not possible or you do not want to use the continue function, the interrupted motion block must always be deleted by a reset.

Parameterize the sources for the reset and continue signals. If you would like to use the motion command MC_Halt to stop a motion block during its run, also add the associated signal source here.

1. Select the Drive Based synchronous application wizard > Motion block operating mode > Motion block > Operating mode digital signals: Data source.
2. J917 Source continue:
In order to use the continue function, enter the associated signal source.
 - 2.1. If a digital input (direct or inverted) acts as the signal source, select the corresponding input.
 - 2.2. If the control byte J01 of the application acts as the signal source, select 2: Parameter.
⇒ Bit 2 of the control byte is set as the data source.
3. J07 Source reset motion block:
To delete a canceled motion block from the continue memory and enable a different motion block start, enter the data source for the motion block reset.
 - 3.1. If a digital input (direct or inverted) acts as the signal source, select the corresponding input.
 - 3.2. If the control word J01 of the application acts as the signal source, select 2: Parameter J01.
⇒ Bit 1 of the control word is set as the data source.
4. J06 Source MC_Halt motion block:
To be able to stop a motion block in general circumstances, enter the associated signal source.
 - 4.1. If a digital input (direct or inverted) acts as the signal source, select the corresponding input.
 - 4.2. If the control word J01 of the application acts as the signal source, select 2: Parameter J01.
⇒ Bit 0 of the control word is set as the data source.

Information

The operating mode configuration is completed.

If you are working with a fieldbus system, next configure the settings specific to CANopen, EtherCAT or PROFINET. The procedure for this can be found in the accompanying STOBER fieldbus manuals (see the chapter [Detailed information](#) [▶ 143]).

4.9.3 Parameterizing command operating mode

The following graphic shows the signal flows of the command operating mode. The elements shown in a lighter shade are optional.

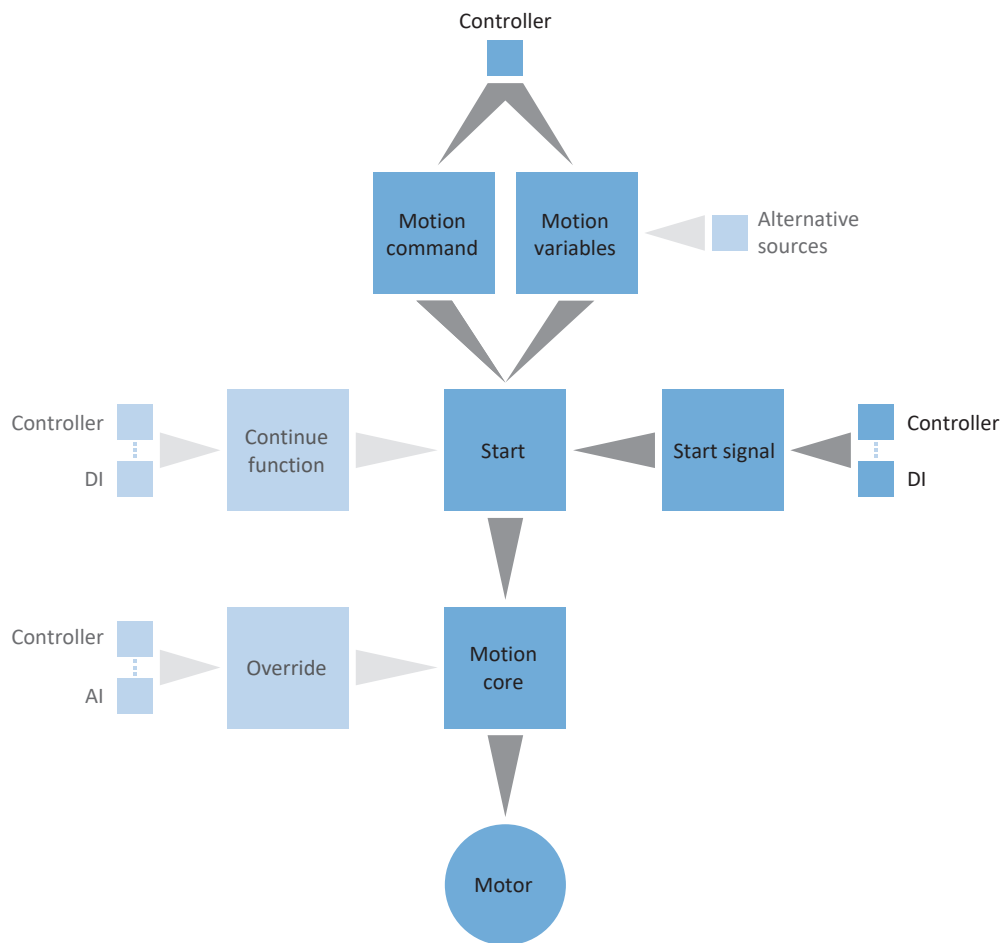


Fig. 7: Command operating mode: Signal flows

In command operating mode, motion profiles are sent from a controller to the drive controller in the form of motion tasks. A motion task includes a PLCopen-compliant motion command which defines the basic motion method of the axis. The associated motion variables supply specifications on velocity, acceleration, deceleration and jerk, from which the motion core of the drive controller calculates the motion profile.

The controller coordinates the time progression and selects the motion tasks for the drive controller that the positioning, velocity and torque/force commands process and run. You define the motion tasks in the controller and parameterize the signal sources from which the drive controller receives the motion tasks from the controller in the DS6.

In command operating mode, drive controllers are either activated using a fieldbus or by a mixed operation of a fieldbus and terminals.

Proceed as follows ...

- Activate command operating mode.
- Optional: Limit torque/force using the operating mode.
- Parameterize the command-specific motion variables.
- Define the source for the start signal.
- Optional: define the continue signal.

Information

The motion variables available for this operating mode that you configure in the software depend on your application case as well as on additional project-specific factors such as the use of a controller or the type of data transmission (fieldbus, terminals).

4.9.3.1 Activating command operating mode

1. Highlight the relevant drive controller in the project tree and click on the first projected axis in the project menu > Wizard area.
 2. Select the Drive Based synchronous application wizard.
 3. Operating modes area:
Activate the option for command operating mode.
- ⇒ The associated wizards are shown and are available for parameterizing the operating mode.

4.9.3.2 Limiting the torque/force using the operating mode

Define whether you also want to limit torque/force using the operating mode. Limiting using the mechanical system takes place in the project configuration of the axis model. An additional limit using the operating mode can be obtained via the analog inputs or a fieldbus.

1. Select the Drive Based synchronous application > Data sources > Torque/force limit: Data source.
2. C132 Source maximum positiv M/F, C133 Source maximum negative M/F:
Define whether the limiting values are transmitted via fieldbus or via the analog inputs.
 - 2.1. If analog inputs are used as sources for the limiting values, select the corresponding inputs.
 - 2.2. If the limiting values are transmitted over a fieldbus, select 4: Parameter C232 and 4: Parameter C233.
3. If the limiting values are transmitted over a fieldbus, select Drive Based synchronous application > Command operating mode.
4. J47 Maximum positive M/F, J48 Maximum negative M/F:
Enter the maximum permitted positive and negative values for torque/force.

Information

The mechanical limit for torque/force is executed by C03 Maximum positive M/F and C05 Maximum negative M/F using the axis model. The additional limit using the operating mode is optional and refers to the values that you defined in the axis model.

4.9.3.3 Parameterizing command-specific motion variables

The parameters for the motion variables acceleration, deceleration and jerk are not included as standard in the fieldbus mapping. You can either store the values for the motion variables permanently on the drive controller or add the corresponding parameters to the fieldbus mapping in order to receive the values from the controller.

Information

Before you start parameterizing the operating mode-specific motion variables, parameterize the general motion variables and signal sources. If you obtain set values for velocity, velocity override or torque/force for your drive project from external sources, proceed as described in the chapter [General motion variables and signal sources](#) [► 59].

4.9.3.3.1 Storing motion variables on the drive controller

If you want to store the values for acceleration, deceleration and jerk permanently on the drive controller, check the default values in the **Command operating mode wizard** and, if necessary, adjust them to your drive project.

- ✓ You have activated command operating mode.
 - ✓ Parameters J44, J45 and J46 are not part of the fieldbus mapping.
1. Select the **Drive Based synchronous application wizard** > **Command operating mode**.
 2. **J44 Acceleration, J45 Deceleration, J46 Jerk:**
Specify the values for acceleration, deceleration and jerk.
- ⇒ The specifications for acceleration, deceleration and jerk are permanently stored on the drive controller.

Information

The parameters in the **Command operating mode wizard** are written by the controller once there is an online connection between the drive controller and controller. In this state, you can read off the values in the parameters that the drive controller receives from the controller. Using parameter J41 Motion-ID, you can identify the ongoing motion task in order to assign the status bit correctly.

4.9.3.3.2 Storing motion variables in the fieldbus mapping

You have the option of obtaining the specifications for acceleration, deceleration and jerk via the controller instead of storing them permanently on the drive controller. To do so, adjust the fieldbus mapping by supplementing the desired parameters both in the receive PDO of the drive controller and in the transmit PDO of the controller.

Supplementing motion variables in EtherCAT fieldbus mapping

If you want to specify values for motion variables using a controller, supplement the desired parameters in the receive PDO of the drive controller.

- ✓ You have activated command operating mode.
- 1. Select the EtherCAT wizard > Received process data RxPDO.
- 2. Enter the coordinate of the parameter that you want to apply to the fieldbus mapping in the Coordinate column.
 - 2.1. If you want to obtain the values for acceleration from the controller, enter J44.
 - 2.2. If you want to obtain the values for deceleration from the controller, enter J45.
 - 2.3. If you want to obtain the values for jerk from the controller, enter J46.

⇒ The parameter is applied to the receive PDO of the drive controller.
In the Name, Data type and Length columns, your information about the parameters is displayed.
- 3. Add your changes to the fieldbus mapping also to the transmit PDO of the controller.

Information

For adjusting the [receive process data](#) (RxPDO), parameter A225 1. rec. PDO Mapping Rx – A228 4. rec. PDO Mapping Rx is available for the SD6 drive controller.

For SI6 and SC6 drive controllers, parameter A225 1. rec. PDO Mapping Rx is available for single-axis controllers and, for double-axis controllers, parameter A226 2. rec. PDO Mapping Rx is also available for the second axis. For double-axis controllers, make sure that the correct axis is specified and the correct parameter A225 for axis A and A226 for axis B is used.

Supplementing motion variables in PROFINET fieldbus mapping

If you want to specify values for motion variables using a controller, supplement the desired parameters in the receive PDO of the drive controller.

- ✓ You have activated command operating mode.
- 1. Select the PROFINET wizard > Received process data RxPZD.
- 2. Enter the coordinate of the parameter that you want to apply to the fieldbus mapping in the Coordinate column.
 - 2.1. If you want to obtain the values for acceleration from the controller, enter J44.
 - 2.2. If you want to obtain the values for deceleration from the controller, enter J45.
 - 2.3. If you want to obtain the values for jerk from the controller, enter J46.

⇒ The parameter is applied to the receive PDO of the drive controller.
In the Name, Data type and Length columns, your information about the parameters is displayed.
- 3. Add your changes to the fieldbus mapping also to the transmit PDO of the controller.

Information

For adjusting the receive PZD, parameter A90 PZD Setpoint Mapping Rx – A92 PZD Setpoint Mapping 3Rx is available for the SD6 drive controller.

For SI6 and SC6 drive controllers, parameter A90 PZD Setpoint Mapping Rx is available for single-axis controllers and, for double-axis controllers, parameter A91 PZD Setpoint Mapping 2Rx is also available for the second axis. For double-axis controllers, make sure that the correct axis is specified and the correct parameter A90 for axis A and A91 for axis B is used.

4.9.3.4 Parameterizing the start signal

In command operating mode, a command selected in J40 can be initiated only using an execute (i.e. start) signal. Define the associated signal source.

1. Select the Drive Based synchronous application > Data sources > Application digital signals: Data source.
2. I100 Source execute:
 - 2.1. If a digital input (direct or inverted) acts as the signal source, select the corresponding input.
 - 2.2. If the control word I210 for the command operating mode acts as the signal source, select 2: Parameter.
⇒ Bit 0 of the control word is set as the data source.

4.9.3.5 Parameterizing the continue signal

If a motion command is canceled during its execution, such as due to an Enable-off or due to the triggering of the safety technology, the software checks whether the motion command can be continued afterward. If this is the case, you can continue and end the interrupted motion command using the continue function.

If you would like to use this function, configure the data source for the continue signal.

If you do not use the continue function, select a new command using J40 and start it using the start signal (execute).

1. Select the Drive Based synchronous application > Command operating mode > Operating mode digital signals: Data source.
2. J38 Source continue:
 - 2.1. If a digital input (direct or inverted) acts as the signal source, select the corresponding input.
 - 2.2. If the control byte J37 for the command operating mode acts as the signal source, select 2: Parameter.
⇒ Bit 0 of the control byte is set as the data source.

Information

The operating mode configuration is completed.

If you are working with a fieldbus system, next configure the settings specific to CANopen, EtherCAT or PROFINET. The procedure for this can be found in the accompanying STOBER fieldbus manuals (see the chapter [Detailed information \[▶ 143\]](#)).

5 More on Drive Based synchronous?

The following chapters summarize the important terms, modules and relationships concerning Drive Based synchronous.

5.1 Drive Based synchronous – Concept

Applications like Drive Based synchronous calculate motions synchronously based on drives. As a rule, the master specifies reference values; the connected slaves move according to a specified proportional relationship to the master. They are either networked over a fieldbus or obtain signals and set values using analog and digital hardware inputs.

The Drive Based synchronous software includes a motion command set based on PLCopen motion control and consequently offers flexible drive-based motion control and technology functions for synchronous operation (multi-axis motions). The PLCopen-compliant standard commands are collected in Drive Based synchronous for different application cases for corresponding operating modes and are extended by additional functions like jerk limit, motion block linking and cams. All motion variables that are necessary for parameterizing individual motion profiles are also collected centrally using operating modes, like relevant set or limiting values.

Based on all parameterized data, the motion core finally calculates an individual motion profile. Specified set values are transmitted to the control cascade, which in turn activates the motor.

Additional wizards such as control panels or jog mode help with commissioning and the configuration test or are designed for emergency operation.

The following graphic shows the Drive Based synchronous-associated components and configuration steps. The elements shown in a lighter shade are optional.

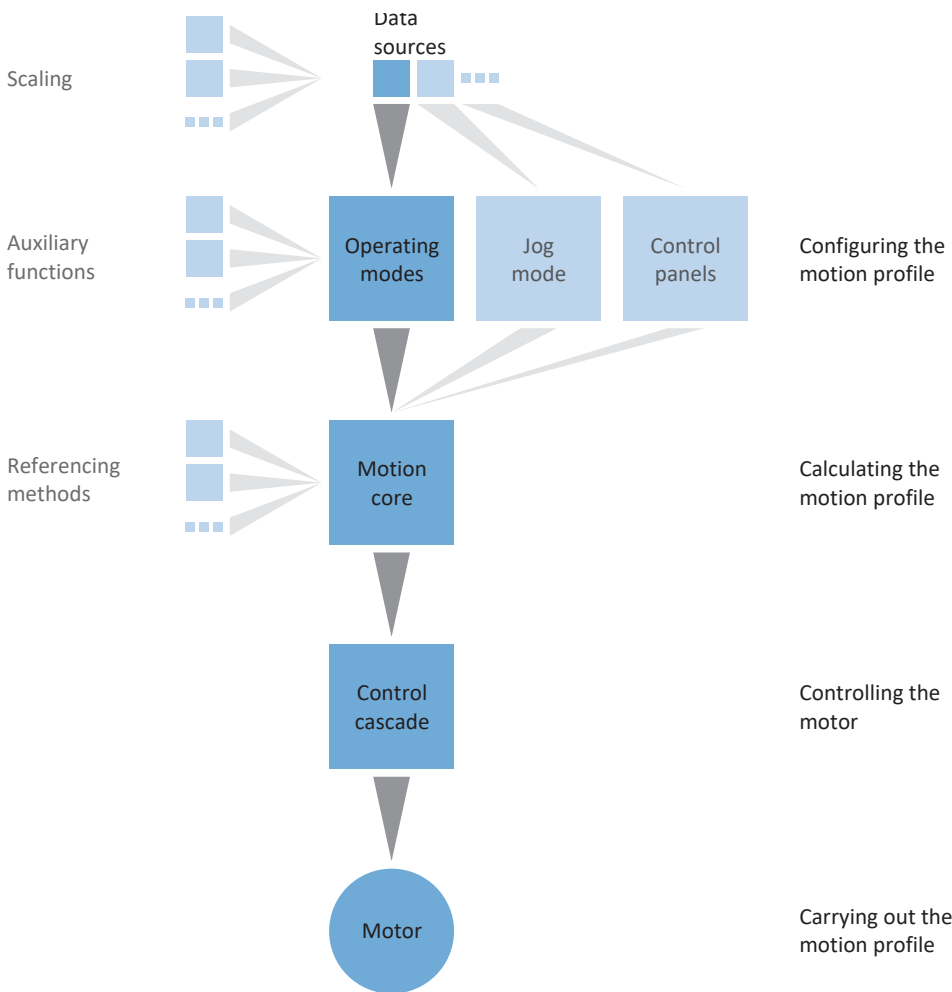


Fig. 8: Components and configuration steps

5.1.1 Operating modes

The operating modes of the Drive Based synchronous application involve command sets bundled by application for configuring individual motion profiles for production operation.

The drive functions of these operating modes span from simple functions like position, velocity and torque/force control types to technology functions for individual axis applications to a synchronous operation of multi-axis systems.

Thanks to flexible, user-defined adjustment options for the operating modes, the Drive Based synchronous application offers a compact – yet extremely variable – scope of functions that is suitable for a wide range of application cases in drive technology. After selecting the suitable operating mode for the respective application case, Drive Based synchronous users concentrate just on parameterizing the most important sequences for their application.

Note that a drive controller can only process one operating mode per axis.

5.1.1.1 Motion block

For the Motion block operation mode, movement properties are predefined in the form of motion blocks. Using special linking options, complete motion sequences can be set up that allow a fast execution of motion sequences – irrespective of whether an external controller is connected or the motion sequences are executed via binary signals.

Typical application examples are small machine automations or machine modules with permanently assigned motion sequences.

5.1.1.2 Command

Command operating mode enables a drive controller to run parameterizable motions. In this process, a range of motion commands are processed that correspond to the behavior of the motion control blocks of the PLCopen standard.

A controller coordinates the time sequences by selecting PLCopen-compliant motion commands, such as MC_MoveAbsolute (move to absolute set position) or MC_MoveRelative (move relative to actual position).

Parameters such as set position, velocity or torque limit can be defined individually.

Typical application examples are individual axis movements that are communicated by a controller (PLC) to the drive controller.

5.1.2 Jog mode

For commissioning, emergency operation and maintenance or repair work, the manufacturer-specific jog mode (manual movement) is available. Using jog mode, you can move the drive independently of the controller, for example.

You can either use jog mode through the display of the SD6 drive controller, through the jog control panel or through a controller that applies the manual movement.

5.1.3 Control panels

Using control panels, you can test the wiring and project configuration of your physical axis model as well as the parameterization of individual operating modes before switching to normal operation.

Drive Based synchronous differentiates between three control panels:

- **Jog control panel** for checking the projected axis model.
The functions of the control panel are identical to those of jog mode.
- **Motion control panel** provides a standard set of motion commands based on PLCopen.
This gives you the option to review the functionality of the parameterized synchronous operation and parameterized command operating mode in advance.
- **Motion block control panel** for reviewing the functioning of the parameterized motion block operating mode.

Because control panels override normal operation, they can be activated only with a deactivated enable signal and exclusively operated by experienced users.

5.1.4 Data sources

The signals that activate a drive controller can be obtained from different data sources.

Each of the possible operating modes has a set of set values that is permanently stored in the software. Furthermore, signals exist for the motion start or motion limits or velocity specifications that generally come from external data sources.

A fieldbus commonly acts as the signal source, though analog or digital hardware inputs or even a mixed operation made up of the above-listed data sources are also possible.

Values that are obtained from external data sources are usually adapted to the stored reference values automatically, i.e. calibrated and calculated with scaling.

5.1.5 Additional functions

Each of the available operating modes can be expanded by drive-based additional functions. For example, these offer convenient monitoring of process variables such as position, velocity or torque/force (cams, comparators) or control of external process variables (PID controller).

5.1.6 Motion core

Based on the data you have projected and parameterized, the motion core calculates a motion profile and associated detail movements as a basis for the drive controller as well as mandatory set values for the control cascade.

5.2 IGB motion bus network

The IGB motion bus transmits signals between SD6 series drive controllers. In an IGB motion bus network, data is synchronized in millisecond cycles and transmitted cyclically, in real time, according to the producer/consumer network model.

Application cases

For synchronous operation, an IGB motion bus network transmits actual and set values between network nodes, such as the master value for the position or information about velocities or torque/force of the individual axes.

System requirements

- At least 2 and up to 32 drive controllers can be networked with each other
- All network nodes must be connected with each other directly, without any intermediary hubs or switches
- The IGB motion bus network must follow a line topology
- The X3A interfaces may be connected only to the X3B interfaces of other drive controllers and vice versa
- The use of a suitable Ethernet cable is required for a functioning network
- The total length of the network may be up to 100 m

Features

In the IGB motion bus network, you can network up to 32 drive controllers with each other, eliminating complicated fieldbus configuration.

In the role of a producer, any drive controller in an IGB motion bus network can place 32 bytes on the bus. Of those 32 bytes, 6 bytes are preassigned with system data that can be used at any time. The remaining 26 bytes can be assigned freely with PDO-capable parameters.

In the role of a consumer, any drive controller can receive up to 64 parameters from the IGB motion bus network from any other drive controller.

The IGB motion bus function of an IGB network is automatically active once a drive controller is dragged and dropped into the newly created network when configuring an IGB motion bus network. Correct operation of an IGB motion bus is required for the device state machine to switch from the Switch on disabled state to the Ready to switch on state.

Each node of an IGB motion bus network requires a unique IGB motion bus address, which can be assigned automatically during network project configuration. The address can be edited manually. In addition, the number of drive controllers in the entire IGB motion bus group must be known to each drive controller of an IGB motion bus network. The associated IGB target number is automatically parameterized during IGB motion bus mapping and transmitted to the individual drive controllers along with the overall configuration.

5.3 Mechanical axis model

Smooth operation of a drive train in combination with one or more drive controllers requires mapping the associated real mechanical environment in DriveControlSuite. There, you define the mechanical axis model as part of commissioning.

The following chapters provide explanations of the configuration options for rotational and translational drives in combination with different position or motor encoders. Position encoders are generally optional, whereas the use of a motor encoder depends on the control mode and motor type.

5.3.1 Rotational drives

If you are working with rotational drives, the following configuration options are available.

5.3.1.1 Endless rotational movements

The schematic diagram shows an endless rotational drive using an example with a motor, gear unit and rotary table.

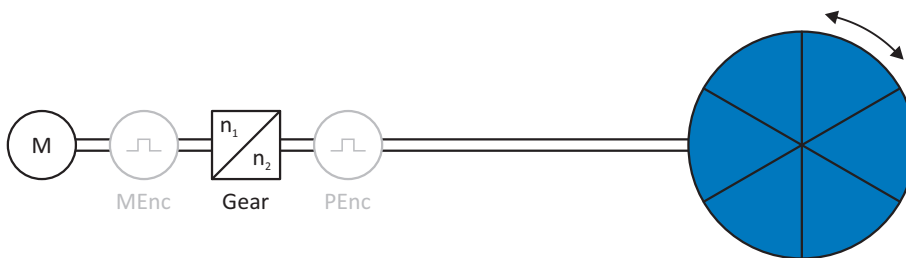


Fig. 9: Endless rotational drive

For endless rotational drives, the following encoder arrangements can be configured:

- Rotational motor encoder = Position encoder
- Rotational motor encoder and external rotational position encoder
- External rotational position encoder without a motor encoder

5.3.1.2 Limited rotational movements

The schematic diagram shows a limited rotational drive using an example with a motor, gear unit and pointer.

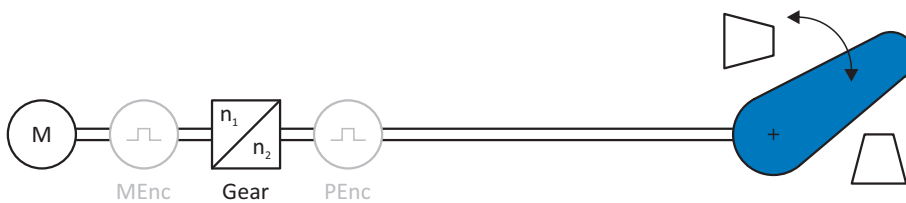


Fig. 10: Limited rotational drive

For limited rotational drives, the following encoder arrangements can be configured:

- Rotational motor encoder = Position encoder
- Rotational motor encoder and external rotational position encoder
- External rotational position encoder without a motor encoder

5.3.2 Translational drives

If you are working with translational drives, the following configuration options are available.

5.3.2.1 Endless translational movements

The schematic diagram shows an endless translational drive using an example with a motor, gear unit, feed and conveyor belt.

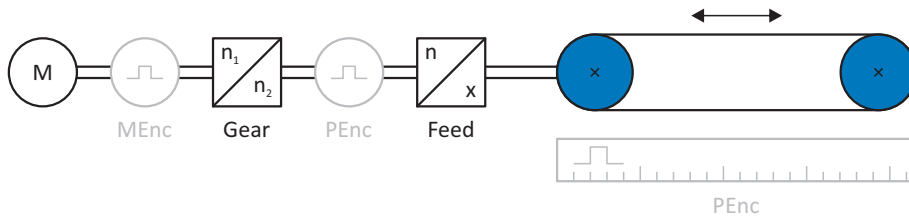


Fig. 11: Endless translational drive

For endless translational drives, the following encoder arrangements can be configured:

- Rotational motor encoder = Position encoder
- Rotational motor encoder and external rotational position encoder
- Rotational motor encoder and external translational position encoder
- External rotational position encoder without a motor encoder
- External translational position encoder without a motor encoder

5.3.2.2 Limited translational movements

The schematic diagram shows a limited translational drive using an example with a motor, gear unit, feed and tool slide.

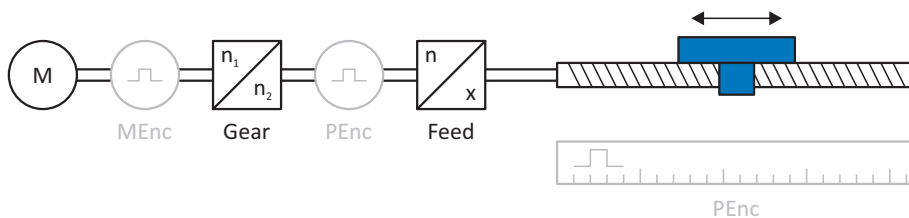


Fig. 12: Limited translational drive

For limited translational drives, the following encoder arrangements can be configured:

- Rotational motor encoder = Position encoder
- Rotational motor encoder and external rotational position encoder
- Rotational motor encoder and external translational position encoder
- External rotational position encoder without a motor encoder
- External translational position encoder without a motor encoder

5.3.2.3 Linear motors

The schematic diagram shows a limited translational drive using the example of a linear motor with an encoder.

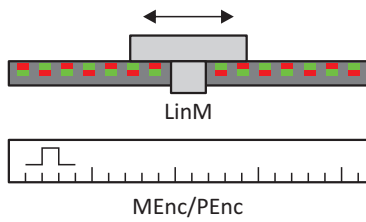


Fig. 13: Limited translational drive – Linear motor

For linear motors, the following encoder arrangements can be configured:

- Translational motor encoder = Position encoder
- Translational motor encoder and external translational position encoder

5.4 Limit switches

Limit switches are sensors that detect when a certain position has been reached.

A distinction is made here between hardware limit switches and software limit switches. While a hardware limit switch is a real switch (hardware), a software limit switch refers to the position limiting implemented in the software.

There are hardware and software limit switches on real axes, while virtual masters only have software limit switches.

Behavior in jog mode, reaching the +/- 31 bit computing limit and simultaneous triggering of positive and negative limit switches are special cases.

5.4.1 Real axes

There are hardware and software limit switches on real axes.

5.4.1.1 Faults

If a hardware or software limit switch is overshoot, a fault is triggered.

Fault

53: Limit switch

Causes

- 1: Hardware-Limit-Switch positive
- 2: Hardware-Limit-Switch negative
- 3: SW-limit switch positive
- 4: SW-limit switch negative
- 5: +/- 31 bit computing limit reached
- 6: Linear motor range
- 7: Both limit switches not connected

The fault can be acknowledged. Note that movement is only possible in the opposite direction of the limit switch, since the direction of the triggered hardware or software limit switch is locked. A direction lock is displayed in parameter I196 Direction block.

5.4.1.2 Denial

If the axis is at a hardware limit switch (I441 Signal /HW limit switch positive or I442 Signal /HW limit switch negative = High) or if it is beyond of the software limit switch, movement is denied in the locked direction.

In this case, parameter I91 Error = 1: Active.

The cause is output in parameter I90 PLCOpen ErrorStop cause. Possible causes are:

- 1: Illegal direction
- 2: SW-limit switch positive
- 3: SW-limit switch negative
- 10: Denied because position is not in circular length
- 11: HW-limit switch positive
- 12: HW-limit switch negative

Information

There are no software limit switches for an infinite axis, since this would contradict the nature of an infinite axis.

A movement limitation is also displayed in parameter E80 Operating condition:

- If E80 = 20: Limit switch, check the parameterization and the connection of the limit switches.
- If E80 = 15: Illegal direction, check the set values and a possible direction lock in parameter I196.

Also check I196 if the permitted direction was limited with I04 Move direction.

Information

Parameter I04 is only available for infinite axes.

5.4.1.3 Hardware limit switches

Hardware limit switches apply if your sources are defined in parameters I101 and I102.

Relevant parameters

- I101 Source positive /limit switch
- I102 Source negative /limit switch
- I441 Signal /HW limit switch positive
- I442 Signal /HW limit switch negative
- I805 Actual value HW-Limit switch positive
- I806 Actual value HW-Limit switch negative
- I52 Delete limit switch memory
- I196 Direction block

If I441 and I442 = 0: Inactive, fault 53 is triggered with cause 7: Both limit switches not connected. Therefore, after parameterizing I101 and I102, check whether the hardware limit switches are also connected physically.

If I101 and I102 = 2: Parameter, then the fault is triggered by I210 Control word application. In this case, check the controller connection.

Passable hardware limit switches

The hardware limit switches can be passed over. It is therefore possible to use an infinite cam as a hardware limit switch. The limit switch is detected if the signal goes to 0: Inactive (parameters I441 and I442).

If the hardware limit switch is overshot, the position at which the limit switch was detected is saved. When resetting limit switches, the axis must have reached or passed back over this saved position before the signal is valid again.

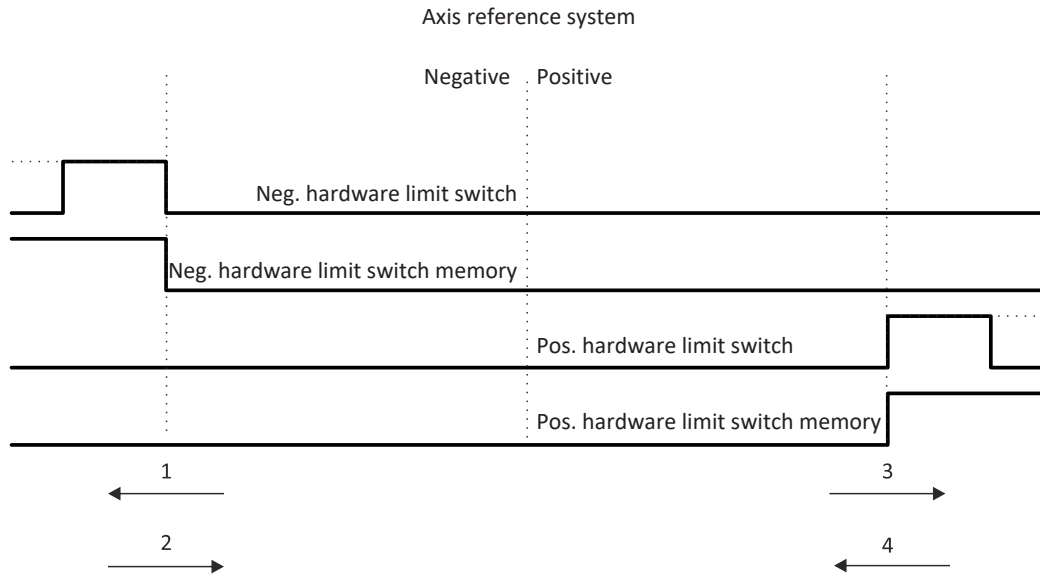


Fig. 14: Hardware limit switch memory

- 1 Setting the memory (negative hardware limit switch) with rising edge
- 2 Resetting the memory (negative hardware limit switch) with falling edge
- 3 Setting the memory (positive hardware limit switch) with rising edge
- 4 Resetting the memory (positive hardware limit switch) with falling edge

Information
 The passable hardware limit switch is no longer effective once the position at which the edge of the limit switch was detected has been passed over again.

Note that the positions are not saved in non-volatile memory. This means that if the axis is beyond a passable limit switch after activation, it must first be reset to the regular travel range.

The entire system is simpler if you do not use any passable limit switches.

During commissioning or in case of defective hardware limit switch connections, problems can arise with the saved positions. These can be deleted with parameter I52. However, deletion only takes place if the corresponding limit switch signal is inactive. Saved positions can also be deleted by referencing, setting a reference using I452 Signal set reference or restarting the drive controller.

During referencing, the hardware limit switches are not evaluated as limit switches. There are referencing methods in which the hardware limit switches are used for referencing.

Example

A positive limit switch ranges from position 100 to 120.
 When moving in the positive direction, it is detected at 100.
 On the return journey, due to hysteresis, it can be the case that the limit switch has already been left at 101.
 Despite this, position 100 must be undershot in order to leave the effective limit switch.

5.4.1.4 Hardware limit switches when referencing

Hardware limit switches assume a special position when referencing.

Limit switch as reference switch

The hardware limit switches can be used in place of a reference switch (I30 Referencing type = 2: Limit switch).

Change of direction

In other referencing types, a hardware limit switch signal leads to a change of direction. However, the signal only causes a change if it matches the direction of travel.

If the detected hardware limit switch does not match the direction of travel (e.g. negative limit switch with positive direction of travel), a fault is then triggered.

Example

At the start of referencing, the axis is between the reference switch and positive limit switch. The referencing direction is positive. The axis moves in the positive direction and first encounters the positive limit switch, not the reference switch. The axis reverses and searches for the reference switch in the other direction.

5.4.1.5 Software limit switches

Software limit switches only apply if the axis has been referenced.

Only limited axes have software limit switches (I00 Position range = 0: Limited).

Relevant parameters

- I50 Software stop positive
- I51 Software stop negative
- I196 Direction block

If the values defined in parameter I50 Software stop positive and parameter I51 Software stop negative are the same, the software limit switches are deactivated.

5.4.1.6 Software limit switches when referencing

Software limit switches are not evaluated when referencing.

5.4.1.7 Shutting down by the limit switch

If a limit switch is approached from the permitted travel range, a fault is immediately triggered after detecting the limit switch edge.

If the current position is on or beyond the limit switch, a fault is only triggered again if the limit switch area has been left for at least 1 s.

5.4.2 Virtual master

Virtual masters only have software limit switches.

Relevant parameters

- G146 Master software limit switch +
- G147 Master software limit switch -

Faults

If a software limit switch is overshot, fault 51: Virtual master limit switch is triggered.

Causes

- 1: SW-limit switch positive
- 2: SW-limit switch negative
- 3: +/- 31 bit computing limit reached

The fault can be acknowledged. It can only be shut down in the opposite direction of the software limit switch. The direction of the triggered software limit switch is locked.

Denial

If the axis is beyond the software limit switch, movement in the wrong direction is denied.

In this case, parameter G161 Virtual master error = 1: Active.

The cause is output in parameter G163 Virtual master error cause:

- 2: SW-limit switch positive
- 3: SW-limit switch negative

Information

There are no software limit switches for an infinite axis, since this would contradict the nature of an infinite axis.

5.4.3 Special cases

A few special cases are described below.

5.4.3.1 Behavior in jog mode

As soon as the axis moves to a software limit switch, a quick stop is triggered instead of a fault. As a result, the axis does not stop precisely on the software limit switch, but shortly thereafter depending on the quick stop ramp. It can immediately be shut down again by the software limit switch. The "incorrect" direction is locked.

5.4.3.2 Reaching the +/- 31 bit computing limit

Fault:

53: Limit switch

Cause:

5: +/- 31 bit computing limit reached

The computing limit can be reached if, for example, so many commands follow after each other for the moving axis that the total distance to be traveled (including decimal places) adds up to over 2^{31} .

5.4.3.3 Simultaneous detection of positive and negative hardware limit switches

Fault:

53: Limit switch

Cause:

7: Both limit switches not connected

This fault is triggered if I441 Signal /HW limit switch positive and I442 Signal /HW limit switch negative are both 0: Inactive. Check whether the hardware limit switches are also connected physically.

If I101 and I102 = 2: Parameter, then the fault is triggered by I210 Control word application. In this case, check the controller connection.

The fault can be acknowledged with I52 Delete limit switch memory.

5.5 Referencing

When commissioning a system with a position measurement system, the relation of a measured axis to an actual axis position must be determined.

When first activating the 24 V_{DC} supply voltage or after changes to the axis model, the actual position of the axis is unknown; a defined starting position is required.

As a rule, this is either identified through a reference search or referencing setting. The associated process is referred to as referencing.

Absolute movements can be performed only in a referenced state.

5.5.1 Referencing methods

The following table shows the possible referencing methods.

	Method	Initial movement	Zero pulse	Feature
	A	Negative	Yes	Negative limit switch
	B	Negative	Yes	Negatively positioned reference switch
	C	Negative	–	Negative limit switch
	D	Negative	–	Negatively positioned reference switch
	E	Positive	Yes	Positive limit switch
	F	Positive	Yes	Positively positioned reference switch
	G	Positive	–	Positive limit switch
	H	Positive	–	Positively positioned reference switch
	I	Positive	Yes	Centrally positioned reference switch
	J	Positive	–	Centrally positioned reference switch
	K	Negative	Yes	Centrally positioned reference switch
	L	Negative	–	Centrally positioned reference switch
	M	Negative	Yes	Zero pulse
	N	Positive	Yes	Zero pulse
	O	–	–	Setting the reference
	P	Positive	–	Torque/force stop
	Q	Positive	Yes	Torque/force stop
	R	Negative	–	Torque/force stop
	S	Negative	Yes	Torque/force stop

Tab. 5: Referencing methods

5.5.1.1 Referencing methods in detail

The following chapters show the individual referencing methods in detail. The following abbreviations are used in the graphics for each referencing method.

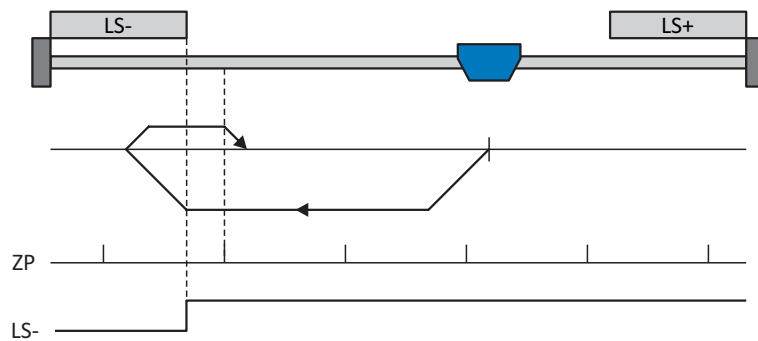
Abbreviation	Meaning
ALT	Alternative
LS	Limit Switch
RS	Reference Switch
ZP	Zero Pulse

Information

For graphically representing axes in this chapter, the left is the smallest and the right is the largest position value. This means a positive movement goes to the right and a negative to the left.

5.5.1.1.1 Referencing method A

Referencing method A determines a reference using a run to the negative limit switch and zero pulse; the run starts with a negative initial movement.



Preparation

1. Activate referencing method A by setting the following parameters to the specified values:
 I30 to 2: Limit switch,
 I31 to 1: Negative,
 I35 to 1: Active.
2. I102:
 Enter the data source for the negative limit switch.
3. I32, I33, I39, I44, I34:
 Define the set values necessary for referencing.
4. I53:
 Define the start of the search for the zero pulse search

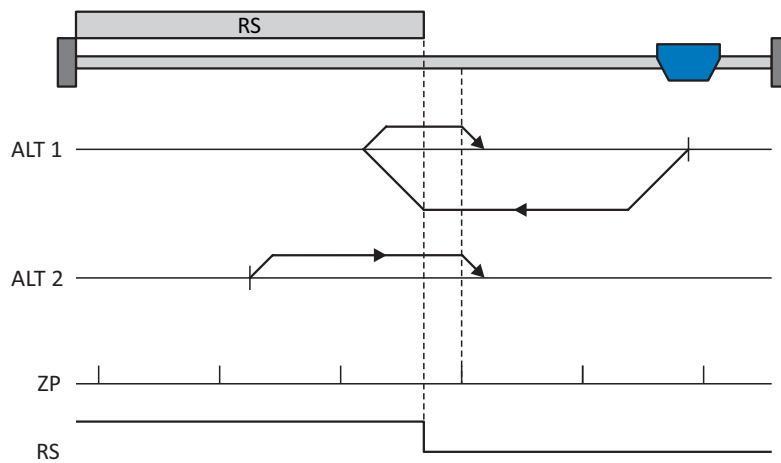
Referencing

If the PLCopen command MC_Home is active, referencing takes places as follows:

1. The drive starts in the negative direction with acceleration I39 and velocity I32.
2. It changes its direction upon reaching the negative limit switch and continues its run with velocity I33 until it reaches the next zero pulse after leaving the limit switch.
3. The current actual position is set to the value of reference position I34 upon reaching the zero pulse.
4. The drive comes to a standstill with deceleration I39.
5. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.2 Referencing method B

Referencing method B determines the reference using a run to the negatively positioned reference switch and zero pulse; the run starts with a negative initial movement.



Preparation

1. Activate referencing method B by setting the following parameters to the specified values:
I30 to 1: Reference input,
I31 to 1: Negative,
I35 to 1: Active.
2. I103:
Enter the data source for the reference switch.
3. I32, I33, I39, I44, I34:
Define the set values necessary for referencing.
4. I53:
Define the start of the search for the zero pulse search

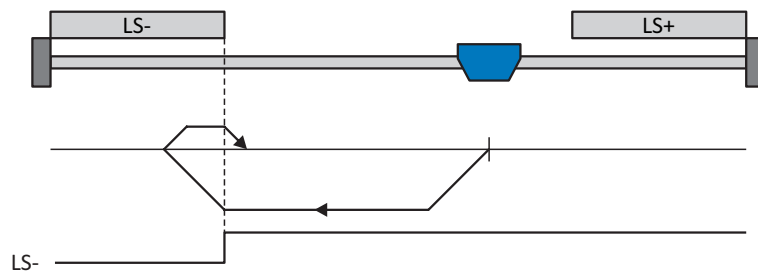
Referencing

If the PLCopen command MC_Home is active, a distinction is made between two referencing variants.

- ✓ Alternative 1: Drive is positioned in front of the reference switch
 1. The drive starts in the negative direction with acceleration I39 and velocity I32.
 2. It changes its direction upon reaching the reference switch and continues its run with velocity I33 until it reaches the next zero pulse after leaving the limit switch.
 3. The current actual position is set to the value of reference position I34 upon reaching the zero pulse.
 4. The drive comes to a standstill with deceleration I39.
 5. If I43 is set to 1: Active, the drive is positioned at reference position I34.
- ✓ Alternative 2: Drive stopped at reference switch
 1. The drive starts in the positive direction with acceleration I39 and velocity I33 until it reaches the next zero pulse after leaving the limit switch.
 2. The current actual position is set to the value of reference position I34 upon reaching the zero pulse.
 3. The drive comes to a standstill with deceleration I39.
 4. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.3 Referencing method C

Referencing method C determines the reference using a run to the negative limit switch; the run starts with a negative initial movement.



Preparation

1. Activate referencing method C by setting the following parameters to the specified values:
 I30 to 2: Limit switch,
 I31 to 1: Negative,
 I35 to 0: Inactive.
2. I102:
 Enter the data source for the negative limit switch.
3. I32, I33, I39, I44, I34:
 Define the set values necessary for referencing.

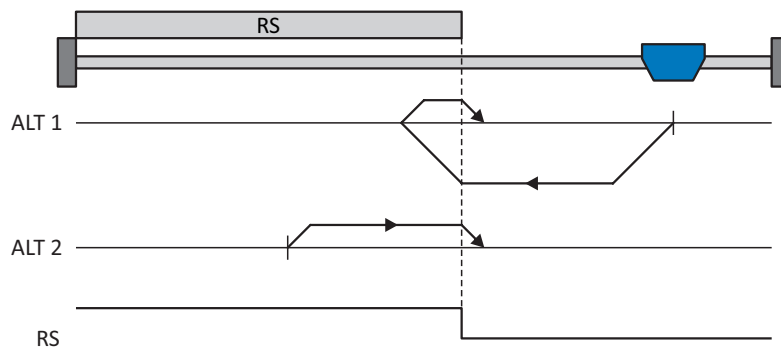
Referencing

If the PLCopen command MC_Home is active, referencing takes places as follows:

1. The drive starts in the negative direction with acceleration I39 and velocity I32.
2. It changes its direction upon reaching the negative limit switch and continues its run with velocity I33 until it leaves the limit switch again.
3. The current actual position is set to the value of the reference position I34 upon leaving the limit switch.
4. The drive comes to a standstill with deceleration I39.
5. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.4 Referencing method D

Referencing method D determines the reference using a run to the negatively positioned reference switch; the run starts with a negative initial movement.



Preparation

1. Activate referencing method D by setting the following parameters to the specified values:
I30 to 1: Reference input,
I31 to 1: Negative,
I35 to 0: Inactive.
2. I103:
Enter the data source for the reference switch.
3. I32, I33, I39, I44, I34:
Define the set values necessary for referencing.

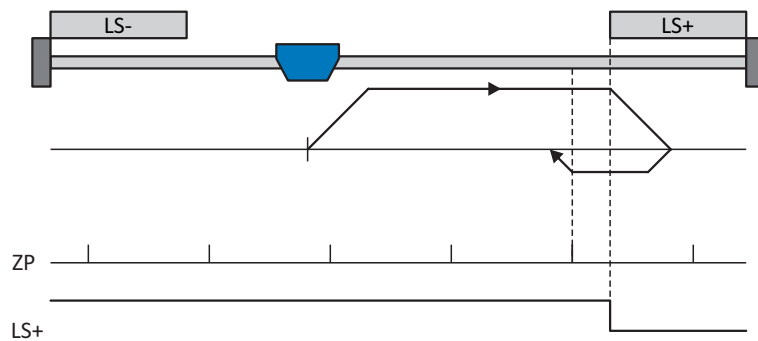
Referencing

If the PLCopen command MC_Home is active, a distinction is made between two referencing variants.

- ✓ Alternative 1: Drive is positioned in front of the reference switch
 1. The drive starts in the negative direction with acceleration I39 and velocity I32.
 2. It changes its direction upon reaching the reference switch and continues its run with velocity I33.
 3. The current actual position is set to the value of the reference position I34 upon leaving the reference switch.
 4. The drive comes to a standstill with deceleration I39.
 5. If I43 is set to 1: Active, the drive is positioned at reference position I34.
- ✓ Alternative 2: Drive stops at reference switch
 1. The drive starts in the positive direction with acceleration I39 and velocity I33.
 2. The current actual position is set to the value of the reference position I34 upon leaving the reference switch.
 3. The drive comes to a standstill with deceleration I39.
 4. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.5 Referencing method E

Referencing method E determines a reference using a run to the positive limit switch and zero pulse; the run starts with a positive initial movement.



Preparation

1. Activate referencing method E by setting the following parameters to the specified values:
I30 to 2: Limit switch,
I31 to 0: Positive,
I35 to 1: Active.
2. I101:
Enter the data source for the positive limit switch.
3. I32, I33, I39, I44, I34:
Define the set values necessary for referencing.
4. I53:
Define the start of the search for the zero pulse search

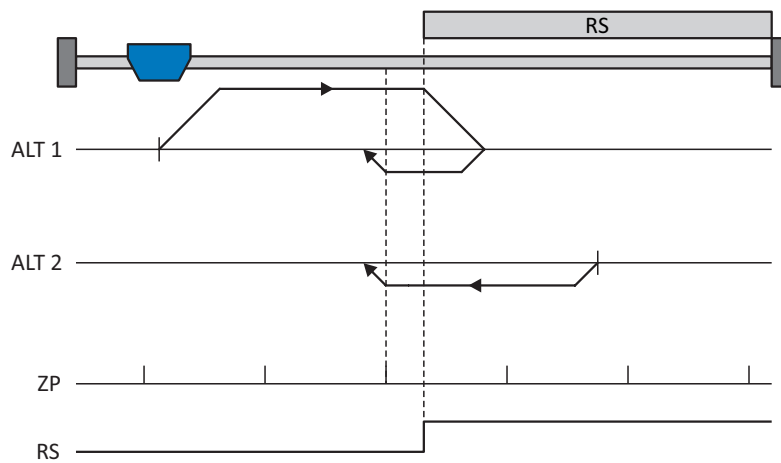
Referencing

If the PLCopen command MC_Home is active, referencing takes places as follows:

1. The drive starts in the positive direction with acceleration I39 and velocity I32.
2. It changes its direction upon reaching the positive limit switch and continues its run with velocity I33 until it reaches the next zero pulse after leaving the limit switch.
3. The current actual position is set to the value of reference position I34 upon reaching the zero pulse.
4. The drive comes to a standstill with deceleration I39.
5. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.6 Referencing method F

Referencing method F determines the reference using a run to the positively positioned reference switch and zero pulse; the run starts with a positive initial movement.



Preparation

1. Activate referencing method F by setting the following parameters to the specified values:
I30 to 1: Reference input,
I31 to 0: Positive,
I35 to 1: Active.
2. I103:
Enter the data source for the reference switch.
3. I32, I33, I39, I44, I34:
Define the set values necessary for referencing.
4. I53:
Define the start of the search for the zero pulse search

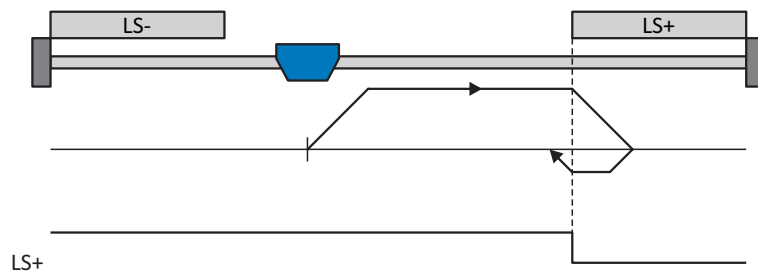
Referencing

If the PLCopen command MC_Home is active, a distinction is made between two referencing variants.

- ✓ Alternative 1: Drive is positioned in front of the reference switch
 1. The drive starts in the positive direction with acceleration I39 and velocity I32.
 2. It changes its direction upon reaching the reference switch and continues its run with velocity I33 until it reaches the next zero pulse after leaving the reference switch.
 3. The current actual position is set to the value of reference position I34 upon reaching the zero pulse.
 4. The drive comes to a standstill with deceleration I39.
 5. If I43 is set to 1: Active, the drive is positioned at reference position I34.
- ✓ Alternative 2: Drive stopped at reference switch
 1. The drive starts in the negative direction with acceleration I39 and velocity I33.
 2. After leaving the reference switch, the current actual position is set to the value of the reference position I34 upon reaching the next zero pulse.
 3. The drive comes to a standstill with deceleration I39.
 4. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.7 Referencing method G

Referencing method G determines the reference using a run to the positive limit switch; the run starts with a positive initial movement.



Preparation

1. Activate referencing method G by setting the following parameters to the specified values:
 I30 to 2: Limit switch,
 I31 to 0: Positive,
 I35 to 0: Inactive.
2. I101:
 Enter the data source for the positive limit switch.
3. I32, I33, I39, I44, I34:
 Define the set values necessary for referencing.

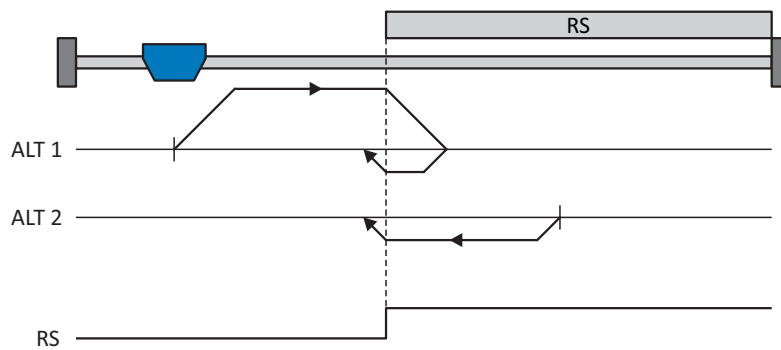
Referencing

If the PLCopen command MC_Home is active, referencing takes place as follows:

1. The drive starts in the positive direction with acceleration I39 and velocity I32.
2. It changes its direction upon reaching the positive limit switch and continues its run with velocity I33 until it leaves the limit switch again.
3. The current actual position is set to the value of the reference position I34 upon leaving the limit switch.
4. The drive comes to a standstill with deceleration I39.
5. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.8 Referencing method H

Referencing method H determines the reference using a run to the positively positioned reference switch; the run starts with a positive initial movement.



Preparation

1. Activate referencing method H by setting the following parameters to the specified values:
 - I30 to 1: Reference input,
 - I31 to 0: Positive,
 - I35 to 0: Inactive.
2. I103:
 - Enter the data source for the reference switch.
3. I32, I33, I39, I44, I34:
 - Define the set values necessary for referencing.

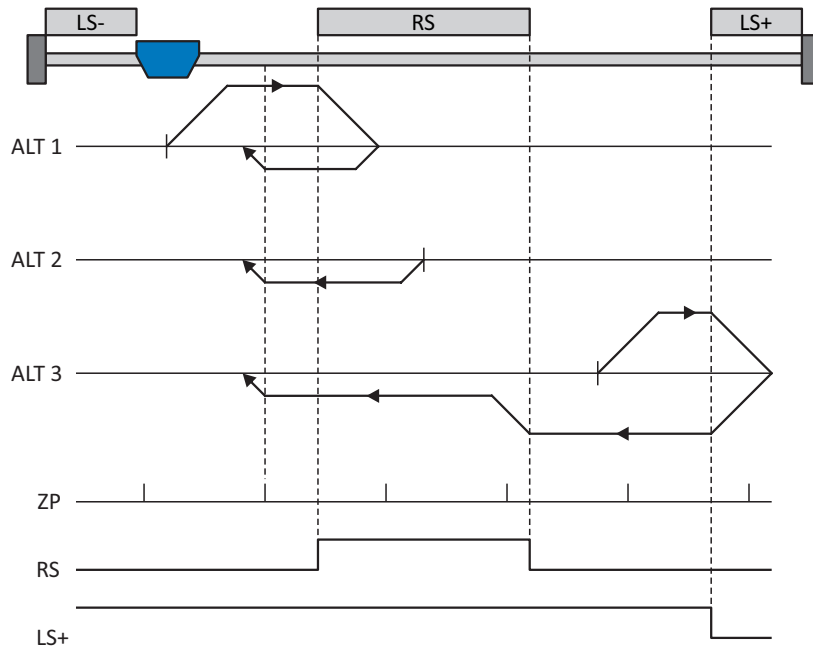
Referencing

If the PLCopen command MC_Home is active, a distinction is made between two referencing variants.

- ✓ Alternative 1: Drive is positioned in front of the reference switch
 1. The drive starts in the positive direction with acceleration I39 and velocity I32.
 2. It changes its direction upon reaching the reference switch and continues its run with velocity I33.
 3. The current actual position is set to the value of the reference position I34 upon leaving the reference switch.
 4. The drive comes to a standstill with deceleration I39.
 5. If I43 is set to 1: Active, the drive is positioned at reference position I34.
- ✓ Alternative 2: Drive stopped at reference switch
 1. The drive starts in the negative direction with acceleration I39 and velocity I33.
 2. The current actual position is set to the value of the reference position I34 upon leaving the reference switch.
 3. The drive comes to a standstill with deceleration I39. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.9 Referencing method I

Referencing method I determines the reference using a run to the centrally positioned reference switch and zero pulse; the run starts with a positive initial movement.



Preparation

1. Activate referencing method I by setting the following parameters to the specified values:
 I30 to 1: Reference input,
 I31 to 0: Positive,
 I35 to 1: Active.
2. I103:
 Enter the data source for the reference switch.
3. I32, I33, I39, I44, I34:
 Define the set values necessary for referencing.
4. I53:
 Define the start of the search for the zero pulse search

Referencing

If the PLCopen command MC_Home is active, a distinction is made between three referencing sequences.

✓ Alternative 1: Drive is positioned between the negative limit and reference switches

1. The drive starts in the positive direction with acceleration I39 and velocity I32.
2. It changes its direction upon reaching the reference switch and continues its run with velocity I33 until it reaches the next zero pulse after leaving the reference switch.
3. The current actual position is set to the value of reference position I34 upon reaching the zero pulse.
4. The drive comes to a standstill with deceleration I39.
5. If I43 is set to 1: Active, the drive is positioned at reference position I34.

✓ Alternative 2: Drive stopped at reference switch

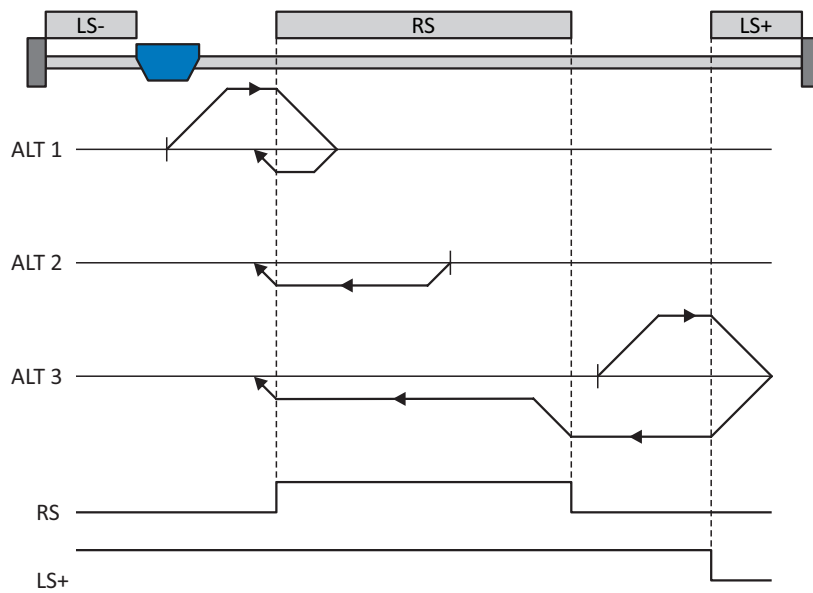
1. The drive starts in the negative direction with acceleration I39 and velocity I33.
2. The current actual position is set to the value of the reference position I34 after leaving the reference switch and upon reaching the next zero pulse.
3. The drive comes to a standstill with deceleration I39.
4. If I43 is set to 1: Active, the drive is positioned at reference position I34.

✓ Alternative 3: Drive is positioned between the reference and positive limit switches

1. The drive starts in the positive direction with acceleration I39 and velocity I32.
2. It changes its direction upon reaching the positive limit switch until the reference switch is reached.
3. Upon reaching the reference switch, the drive changes its velocity to I33 until it leaves the reference switch.
4. The current actual position is set to the value of the reference position I34 after leaving the reference switch and upon reaching the next zero pulse.
5. The drive comes to a standstill with deceleration I39.
6. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.10 Referencing method J

Referencing method J determines the reference using a run to the centrally positioned reference switch; the run starts with a positive initial movement.



Preparation

1. Activate referencing method J by setting the following parameters to the specified values:
 I30 to 1: Reference input,
 I31 to 0: Positive,
 I35 to 0: Inactive.
2. I103:
 Enter the data source for the reference switch.
3. I32, I33, I39, I44, I34:
 Define the set values necessary for referencing.

Referencing

If the PLCopen command MC_Home is active, a distinction is made between three referencing sequences.

- ✓ Alternative 1: Drive is positioned between the negative limit and reference switches
1. The drive starts in the positive direction with acceleration I39 and velocity I32.
 2. It changes its direction upon reaching the reference switch and continues its run with velocity I33 until it leaves the reference switch.
 3. The current actual position is set to the value of the reference position I34 upon leaving the reference switch.
 4. The drive comes to a standstill with deceleration I39.
 5. If I43 is set to 1: Active, the drive is positioned at reference position I34.

✓ Alternative 2: Drive stops at reference switch

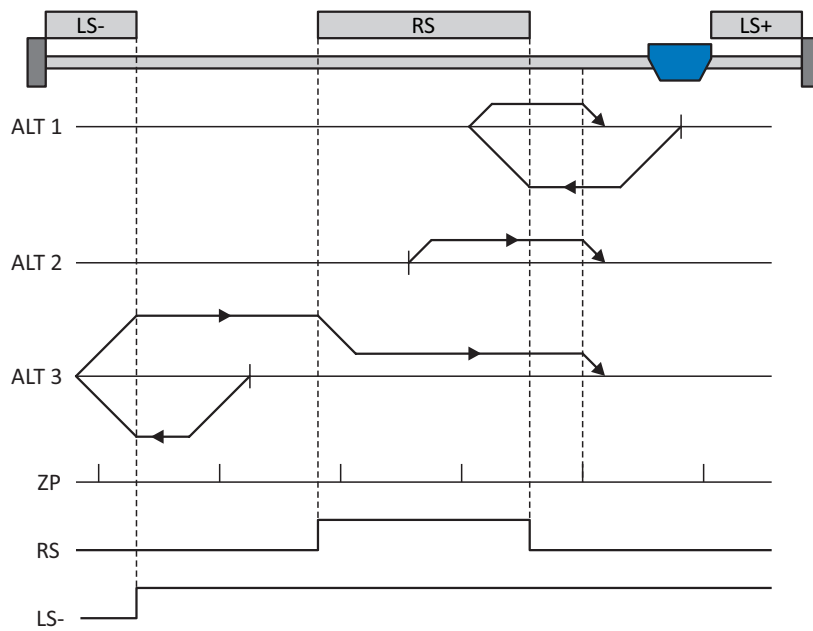
1. The drive starts in the negative direction with acceleration I39 and velocity I33.
2. The current actual position is set to the value of the reference position I34 upon leaving the reference switch.
3. The drive comes to a standstill with deceleration I39.
4. If I43 is set to 1: Active, the drive is positioned at reference position I34.

✓ Alternative 3: Drive is positioned between the reference and positive limit switches

1. The drive starts in the positive direction with acceleration I39 and velocity I32.
2. Upon reaching the positive limit switch, the drive changes its direction until it reaches the reference switch.
3. Upon reaching the reference switch, the drive changes its velocity to I33 until it leaves the reference switch.
4. The current actual position is set to the value of the reference position I34 upon leaving the reference switch.
5. The drive comes to a standstill with deceleration I39.
6. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.11 Referencing method K

Referencing method K determines the reference using a run to the centrally positioned reference switch and zero pulse; the run starts with a negative initial movement.



Preparation

1. Activate referencing method K by setting the following parameters to the specified values:
 - I30 to 1: Reference input,
 - I31 to 1: Negative,
 - I35 to 1: Active.
2. I103:
 - Enter the data source for the reference switch.
3. I32, I33, I39, I44, I34:
 - Define the set values necessary for referencing.
4. I53:
 - Define the start of the search for the zero pulse search

Referencing

If the PLCopen command MC_Home is active, a distinction is made between three referencing variants.

✓ Alternative 1: Drive is positioned between the reference and positive limit switches

1. The drive starts in the negative direction with acceleration I39 and velocity I32.
2. It changes its direction upon reaching the reference switch and continues its run with velocity I33 until it reaches the next zero pulse after leaving the limit switch.
3. The current actual position is set to the value of reference position I34 upon reaching the zero pulse.
4. The drive comes to a standstill with deceleration I39.
5. If I43 is set to 1: Active, the drive is positioned at reference position I34.

✓ Alternative 2: Drive stopped at reference switch

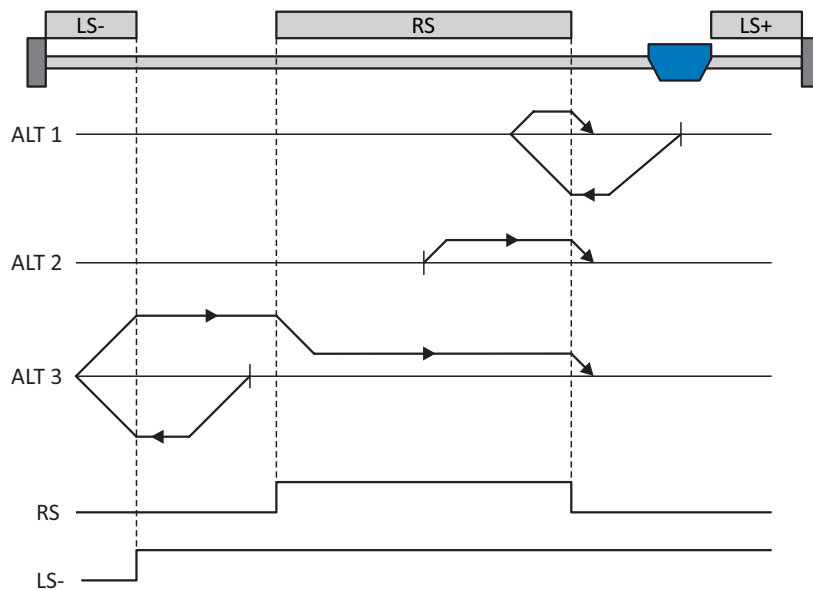
1. The drive starts in the positive direction with acceleration I39 and velocity I33.
2. The current actual position is set to the value of the reference position I34 after leaving the reference switch and upon reaching the next zero pulse.
3. The drive comes to a standstill with deceleration I39.
4. If I43 is set to 1: Active, the drive is positioned at reference position I34.

✓ Alternative 3: Drive is positioned between the negative limit and reference switches

1. The drive starts in the negative direction with acceleration I39 and velocity I32.
2. It changes its direction upon reaching the negative limit switch until the reference switch is reached.
3. Upon reaching the reference switch, the drive changes its velocity to I33 until it leaves the reference switch.
4. The current actual position is set to the value of the reference position I34 after leaving the reference switch and upon reaching the next zero pulse.
5. The drive comes to a standstill with deceleration I39.
6. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.12 Referencing method L

Referencing method L determines the reference using a run to the centrally positioned reference switch; the run starts with a negative initial movement.



Preparation

1. Activate referencing method L by setting the following parameters to the specified values:
 I30 to 1: Reference input,
 I31 to 1: Negative,
 I35 to 0: Inactive.
2. I103:
 Enter the data source for the reference switch.
3. I32, I33, I39, I44, I34:
 Define the set values necessary for referencing.

Referencing

If the PLCopen command MC_Home is active, a distinction is made between three referencing variants.

- ✓ Alternative 1: Drive is positioned between the reference and positive limit switches
1. The drive starts in the negative direction with acceleration I39 and velocity I32.
 2. It changes its direction upon reaching the reference switch and continues its run with velocity I33 until it leaves the reference switch.
 3. The current actual position is set to the value of the reference position I34 upon leaving the reference switch.
 4. The drive comes to a standstill with deceleration I39.
 5. If I43 is set to 1: Active, the drive is positioned at reference position I34.

✓ Alternative 2: Drive stops at reference switch

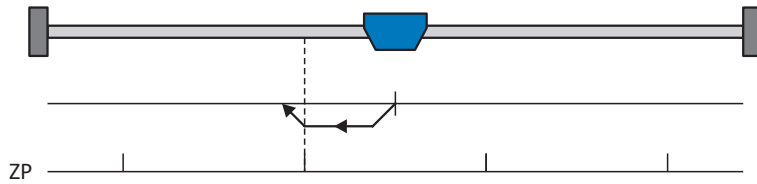
1. The drive starts in the positive direction with acceleration I39 and velocity I33.
2. The current actual position is set to the value of the reference position I34 upon leaving the reference switch.
3. The drive comes to a standstill with deceleration I39.
4. If I43 is set to 1: Active, the drive is positioned at reference position I34.

✓ Alternative 3: Drive is positioned between the negative limit and reference switches

1. The drive starts in the negative direction with acceleration I39 and velocity I32.
2. It changes its direction upon reaching the negative limit switch until the reference switch is reached.
3. Upon reaching the reference switch, the drive changes its velocity to I33 until it leaves the reference switch.
4. The current actual position is set to the value of the reference position I34 upon leaving the reference switch.
5. The drive comes to a standstill with deceleration I39.
6. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.13 Referencing method M

This method determines the reference using a run to the zero pulse that starts with a negative initial movement.



Preparation

1. Activate referencing method M by setting the following parameters to the specified values:
 I30 to 3: Zero pulse,
 I31 to 1: Negative,
2. I32, I39, I44, I34:
 Define the set values necessary for the referencing.

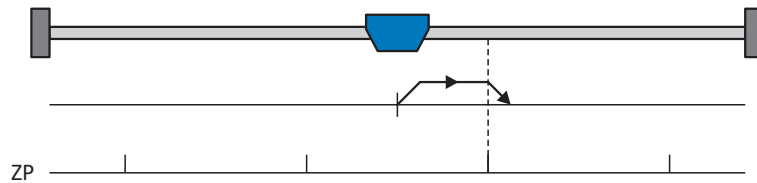
Referencing

If the PLCopen command MC_Home is active, referencing takes places as follows:

1. The drive starts in the negative direction with acceleration I39 and velocity I32.
2. The current actual position is set to the value of reference position I34 upon reaching the zero pulse.
3. The drive comes to a standstill with deceleration I39.
4. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.14 Referencing method N

Referencing method N determines the reference using a run to the zero pulse; the run starts with a positive initial movement.



Preparation

1. Activate referencing method N by setting the following parameters to the specified values:
I30 to 3: Zero pulse,
I31 to 0: Positive,
2. I32, I39, I44, I34:
Define the set values necessary for the referencing.

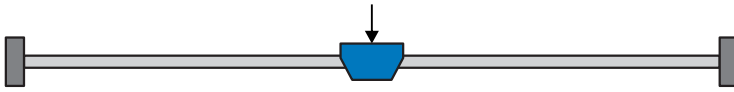
Referencing

If the PLCopen command MC_Home is active, referencing takes places as follows:

1. The drive starts in the positive direction with acceleration I39 and velocity I32.
2. The current actual position is set to the value of reference position I34 upon reaching the zero pulse.
3. The drive comes to a standstill with deceleration I39.
4. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.15 Referencing method 0

Referencing method 0 determines the reference by setting the reference to any position.



Preparation

1. I30:
Activate referencing method 0 by setting this parameter to 5: Define home.
2. I34:
Define the reference position.

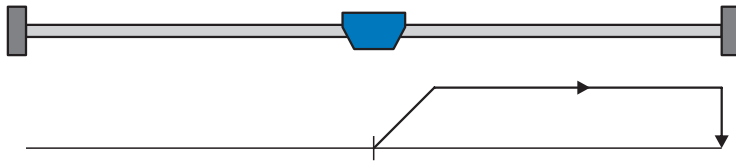
Referencing

If the PLCopen command MC_Home is active, referencing takes places as follows:

The current actual position is set to the value of the reference position I34.

5.5.1.1.16 Referencing method P

Referencing method P determines the reference using a run with a torque/force stop; the run starts with a positive initial movement.



Preparation

1. Activate referencing method P by setting the following parameters to the specified values:
 I30 to 4: Torque/force stopper,
 I31 to 0: Positive,
 I35 to 0: Inactive.
2. I32, I39, I44, I34:
 Define the set values necessary for the referencing.
3. I28, I29:
 Define the torque/force limit. If the actual torque for the time stored in I29 is above the limit defined in I28 for an ongoing period, the torque/force limit is reached.

Information

In case of a selected value for the force variable that is too big, the machine can be damaged; in case of a selected value that is too small, an incorrect reference position may be applied.

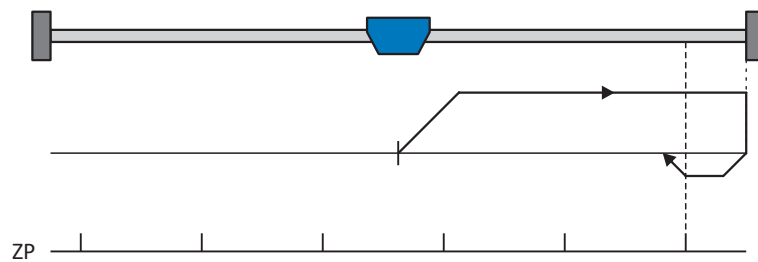
Referencing

If the PLCopen command MC_Home is active, referencing takes places as follows:

1. The drive starts in the positive direction with acceleration I39 and velocity I32.
2. The current actual position is set to the value of reference position I34 upon reaching the torque/force limit.
3. The set values are set to the value 0 with deceleration I39.

5.5.1.1.17 Referencing method Q

Referencing method Q determines the reference using a run with a torque/force stop and zero pulse; the run starts with a positive initial movement.



Preparation

1. Activate referencing method Q by setting the following parameters to the specified values:
 I30 to 4: Torque/force stopper,
 I31 to 0: Positive,
 I35 to 1: Active.
2. I32, I33, I39, I44, I34:
 Define the set values necessary for referencing.
3. I28, I29:
 Define the torque/force limit. If the actual torque for the time stored in I29 is above the limit defined in I28 for an ongoing period, the torque/force limit is reached.
4. I53:
 Define the start of the search for the zero pulse search.

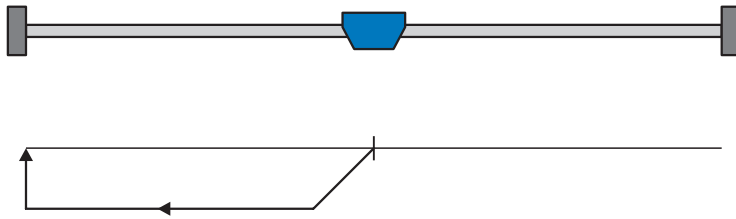
Referencing

If the PLCopen command MC_Home is active, referencing takes places as follows:

1. The drive starts in the positive direction with acceleration I39 and velocity I32.
2. The drive changes its direction upon reaching the torque/force stop and when the time stored in I29 has passed, and continues its run with velocity I33 until it reaches the next zero pulse.
3. Upon reaching the zero pulse, the current actual position is set to the value of the reference position I34.
4. The set values are set to the value 0 with deceleration I39.
5. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.18 Referencing method R

Referencing method R determines the reference using a run with a torque/force stop; the run starts with a negative initial movement.



Preparation

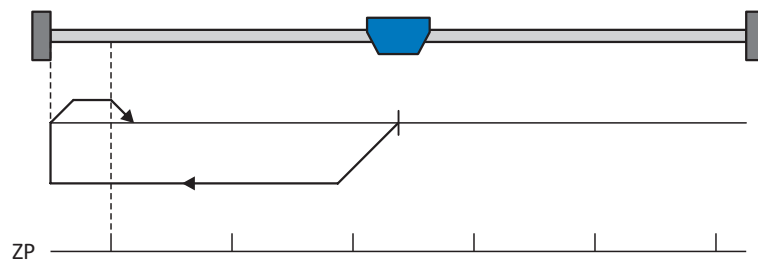
1. Activate referencing method R by setting the following parameters to the specified values:
I30 to 4: Torque/force stopper,
I31 to 1: Negative,
I35 to 0: Inactive.
2. I32, I33, I39, I44, I34:
Define the set values necessary for referencing.
3. I28, I29:
Define the torque/force limit. If the actual torque for the time stored in I29 is above the limit defined in I28 for an ongoing period, the torque/force limit is reached.

Referencing

1. The drive starts in the positive direction with acceleration I39 and velocity I32.
2. Upon reaching the torque/force stop, the current actual position is set to the value of the reference position I34.
3. The set values are set to the value 0 with deceleration I39.
4. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.1.1.19 Referencing method S

Referencing method S determines the reference using a run with a torque/force stop and zero pulse; the run starts with a negative initial movement.



Preparation

1. Activate referencing method S by setting the following parameters to the specified values:
 I30 to 4: Torque/force stopper,
 I31 to 1: Negative,
 I35 to 1: Active.
2. I32, I33, I39, I44, I34:
 Define the set values necessary for referencing.
3. I28, I29:
 Define the torque/force limit. If the actual torque for the time stored in I29 is above the limit defined in I28 for an ongoing period, the torque/force limit is reached.
4. I53:
 Define the start of the search for the zero pulse search

Referencing

If the PLCopen command MC_Home is active, referencing takes places as follows:

1. The drive starts in the negative direction with acceleration I39 and velocity I32.
2. The drive changes its direction upon reaching the torque/force stop and once the time stored in I29 has passed, and continues its run with velocity I33 until it reaches the next zero pulse.
3. Upon reaching the zero pulse, the current actual position is set to the value of the reference position I34.
4. The set values are set to the value 0 with deceleration I39.
5. If I43 is set to 1: Active, the drive is positioned at reference position I34.

5.5.2 Reference position

The reference position is the value set in the reference point. The reference point is determined by the referencing method.

Reference position I34, e.g. 2.50 m, is assigned to the reference point through referencing.

If the reference has been set, the drive comes to a standstill just behind the reference position.

5.6 Drive Based device control

Device control is the central component of a drive. It defines the control channel from which set values are processed and where actual values are output.

Drive Based device control is based on DRIVECOM profile drive technology 21; this profile describes the control sequence of a drive controller based on a device state machine. Each operating state here represents a certain behavior that can be changed only through defined events. These events are assigned to individual state transitions.

For example, device control regulates the device start-up when turning on the supply voltage, the enable state of the drive controller, the process of a quick stop, the behavior in case of fault and the restart after a fault.

Some of the conditions and responses coupled with the state transitions can be influenced specific to the user. For example, it is possible to define the end of a quick stop or enable delays in a way that is tailored to the respective application case (see device control wizard Drive Based).

The following chapters describe the operating states of a drive controller provided by Drive Based device control and the possible state transitions associated with it. You will also learn whether certain actions are necessary on your part to reach the individual operating states, what those actions are and which user-specific factors you can parameterize yourself.

5.6.1 Drive Based device state machine

A device state machine describes the different operating states of a drive controller along with the possible state change.

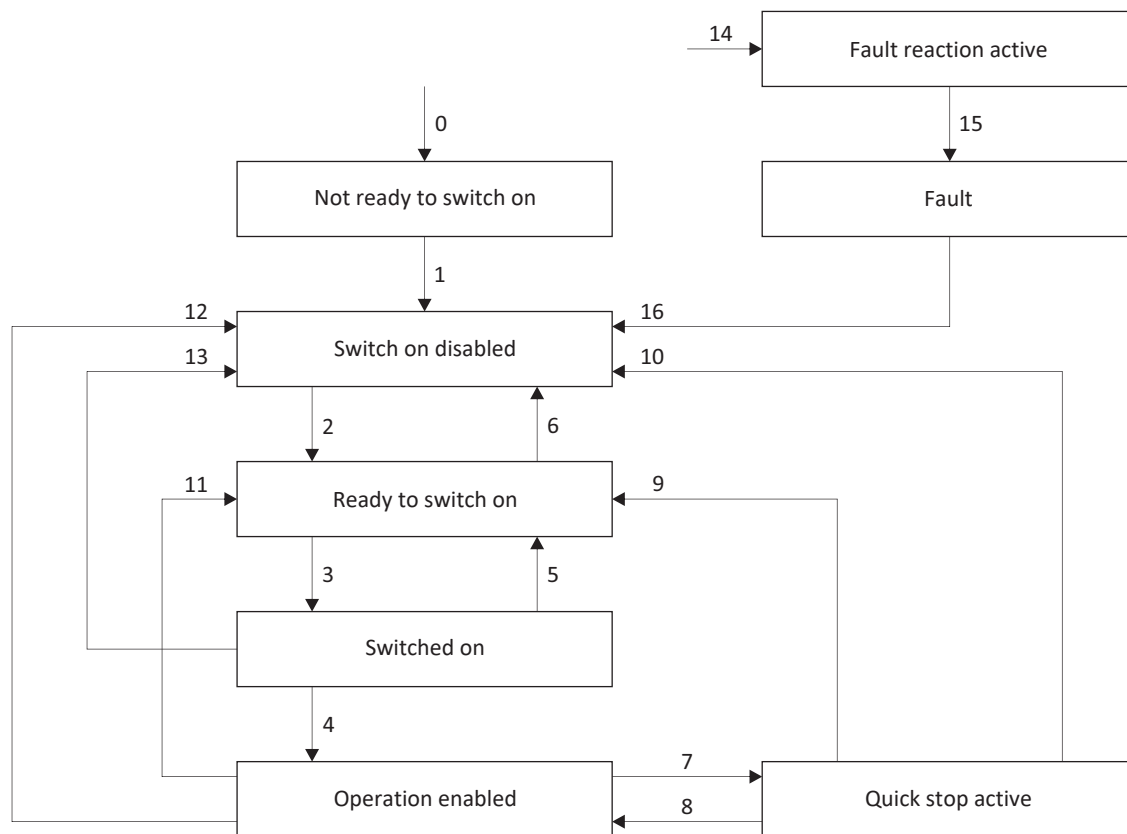


Fig. 15: Drive Based device state machine: Possible operating states and state change

5.6.2 States, transitions and conditions

The states provided by Drive Based device control all have specific features. A state either transitions into another state automatically or requires certain actions from the user. There are also transition conditions that you can parameterize to be tailored to your application case using the Drive Based device control wizard.

Parameter E48 shows the current device state of a drive controller.

5.6.2.1 Not ready to switch on

Features

- The drive is being initialized, tested and associated parameters are being loaded
- The power unit as well as the drive and activation functions are locked
- Brakes remain engaged

Transition to Switch on disabled (1)

The drive controller switches to Switch on disabled (1) automatically after successfully completing the self-test.

5.6.2.2 Switch on disabled

Features

- Self-test was successfully completed
- The power unit as well as the drive and activation functions are locked
- Brakes are or remain engaged in the event of an inactive release override

Possible causes of a switch-on lockout include:

1. Insufficient or no supply voltage or deactivated DC link supply
2. No axis active or active axis switching
3. Projected IGB motion bus not active
4. Safety technology active

Transition to Ready to switch on (2)

The exact cause of a switch-on lockout can be found in parameters E47 and E49.

The transition to Ready to switch on happens automatically upon remedying the relevant cause of the switch-on lockout.

Application-specific parameterization

A34 Auto start:

If this parameter is activated, the drive controller switches directly to the state `Operation enabled` if an enable signal is present.



Injury to persons and material damage due to unexpected motor startup!

Only activate autostart if the standards and regulations applicable to the system or machine in question permit a direct switch to the `Operation enabled` device state.

5.6.2.3 Ready to switch on

Features

- Power unit and drive function are locked, activation function is enabled
- Brakes are or remain engaged in the event of an inactive release override

Transitions to Switched on (3) or Switch on disabled (6)

Activate the enable signal (3) or one of the possible causes of a switch-on lockout is present (6).

Application-specific parameterization

In addition to the enable signal at terminal X1, the Drive Based device state machine also supports the use of an additional enable signal, for example over a fieldbus or using digital inputs. The enable signal and additional enable signals are coupled by an AND link. In this case, the drive controller is enabled only if the value of both signals is [HIGH](#).

A60 Source additional enable:

If you are working with an additional enable signal, define the data source of the additional signal in this parameter. In the case of a fieldbus connection, A180 bit 0 supplies the additional enable signal.

5.6.2.4 Switched on

Features

- Power unit is switched on
- Drive function is locked; set values are not processed
- Drive remains free of torque and force
- Brakes remain engaged in the event of an inactive release override

Transitions to Operation enabled (4), Ready to switch on (5) or Switch on disabled (13)

Leave the enable signal active (4) or deactivate it (5). If one of the possible causes of a switch-on lockout is present, the drive controller automatically switches to Switch on disabled (13).

5.6.2.5 Operation enabled

Features

- Power unit is switched on.
- Drive function is enabled, set values are processed.
- Brakes are released with the first active motion command

Transitions to Quick stop active (7), Ready to switch on (11) or Switch on disabled (12)

Request a quick stop (7) or deactivate the enable signal (11). If one of the possible causes of switch-on lockout is present, the drive controller automatically switches to Switch on disabled (12).

Application-specific parameterization

- A43 Switch-off delay X1 enable:
If the enable signal at terminal X1 of a safety relay is affected by OSSD test pulses, the drive controller cannot differentiate these signals from a functional enable-off. An incorrect interpretation of the test pulses can be avoided by defining a delay time in this parameter that is greater than the duration of the test pulses. This then eliminates unexpected switching of the drive controller to the states Ready to switch on or Quick stop active (see parameter A44).
- A44 Quick stop for enable off:
This parameter results in a drive being stopped with a quick stop in case of an inactive enable. In this case, the drive controller switches to the state quick stop active. If parameter A44 is not activated, the drive switches to the Ready to switch on state in case of Enable-off.

5.6.2.6 Quick stop active

Features

- Power unit is switched on; drive function is enabled
- A quick stop is executed
- Brakes remain released

Transition to Operation enabled (8), Ready to switch on (9) or Switch on disabled (10)

Cancel the quick stop request (8). If the quick stop was carried out fully while the enable signal is inactive, the drive controller automatically switches to Ready to switch on (9); if one of the possible causes of a switch-on lockout is present, the drive controller automatically switches to Switch on disabled (10).

Application-specific parameterization

- A39 Maximum quick stop duration for enable off:
If the drive is stopped with a quick stop in the case of an inactive enable signal (see A44), define the maximum time after which the power unit is switched off in this parameter.
- A45 Quick stop end:
Define whether a quick stop is ended with a standstill of the drive or by canceling the quick stop request in this parameter.
- A62 Source /quick stop:
Define how a requested quick stop is triggered in this parameter.

5.6.2.7 Fault reaction active

- A drive error has occurred
- Fault response is executed depending on the respective fault event
- Brakes are activated depending on the respective fault response

Transition to fault (15)

After completed fault response, the drive controller switches to Fault (15) automatically.

Application-specific parameterization

A29 Quick stop in case of fault:

If this parameter is activated, the drive is stopped with a quick stop, if possible, in case of fault; if the parameter is inactive, the movement of the axis is no longer controlled by the drive controller.

5.6.2.8 Fault

- A drive error has occurred
- The fault response is finished
- The power unit as well as drive and activation functions are locked
- Brakes are engaged in the event of an inactive release override

Transition to Switch on disabled (16)

After remedying the fault cause and then acknowledging the fault message, the drive controller automatically switches to Switch on disabled (16).

5.7 Motion commands

The individual operating modes have special motion commands that are based on the PLCopen standard and are supplemented by three unique STOBER commands (MC_DoNothing, MC_MoveSpeed and MC_TorqueControl). Each of these motion commands can be overwritten during execution, with the exception of MC_Stop. To be able to execute a motion command, the following requirements must be met:

- Local and jog mode must not be activated.
- A drive controller must not be in the Switch on disabled or Fault device state.

Command	Description	Control type	Necessary motion variables
MC_DoNothing	–	–	–
MC_MoveSpeed	Axis moves without end with set velocity (without position control)	Velocity	<ul style="list-style-type: none"> ▪ Velocity, override ▪ Acceleration ▪ Deceleration ▪ Jerk
MC_MoveVelocity	Axis moves without end with set velocity (with position control)	Position	<ul style="list-style-type: none"> ▪ Velocity, override ▪ Acceleration ▪ Deceleration ▪ Jerk
MC_TorqueControl	Axis moves without end with set torque/force	Torque/force	<ul style="list-style-type: none"> ▪ Torque/force
MC_MoveRelative	Axis moves a relative distance; set position is relative to the actual position at the start of the command	Position	<ul style="list-style-type: none"> ▪ Position ▪ Velocity, override ▪ Acceleration ▪ Deceleration ▪ Jerk
MC_MoveAdditive	Axis moves a relative distance; set position is relative to the set position of the previous motion command	Position	<ul style="list-style-type: none"> ▪ Position ▪ Velocity, override ▪ Acceleration ▪ Deceleration ▪ Jerk
MC_MoveAbsolute	Axis moves to an absolute set position (referencing necessary)	Position	<ul style="list-style-type: none"> ▪ Position ▪ Velocity, override ▪ Acceleration ▪ Deceleration ▪ Jerk
MC_Home	Axis is referenced	–	<ul style="list-style-type: none"> ▪ Velocity, override ▪ Acceleration ▪ Deceleration ▪ Jerk ▪ Torque/force

Command	Description	Control type	Necessary motion variables
MC_Halt	Axis is stopped; next command can be run before standstill	Velocity	<ul style="list-style-type: none"> ▪ Deceleration ▪ Jerk
MC_Stop	Axis is brought to a standstill; next command can be executed after standstill	Velocity	<ul style="list-style-type: none"> ▪ Deceleration ▪ Jerk
MC_GearIn	Axis moves synchronously with a master; velocity- based coupling (electronic gear unit)	Position	<ul style="list-style-type: none"> ▪ Acceleration ▪ Deceleration ▪ Jerk
MC_GearInPosition	Axis moves synchronously with a master; position- based coupling (electronic gear unit)	Position	<ul style="list-style-type: none"> ▪ Position, master position
MC_GearOut	Axis ends the ongoing synchronous motion and continues with the last synchronous velocity (electronic gear unit)	Position	<ul style="list-style-type: none"> ▪ Velocity override ▪ Acceleration ▪ Deceleration ▪ Jerk

Tab. 6: Drive Based synchronous: Available motion commands

5.8 Electronic nameplate

STOBER synchronous servo motors are generally equipped with EnDat encoders that provide a special memory. This memory includes the electronic nameplate, i.e. all type-relevant master data as well as special mechanical and electronic values of a motor.

When you operate a drive controller with a STOBER synchronous servo motor and an EnDat encoder, the electronic nameplate is read and all data transferred if the drive controller is connected online. This specifically includes job data, current controller values, motor parameters, data on motor temperature sensors, brakes, the commutation offset and the measured electromotive force. These can be manually optimized later and resaved in the drive controller.

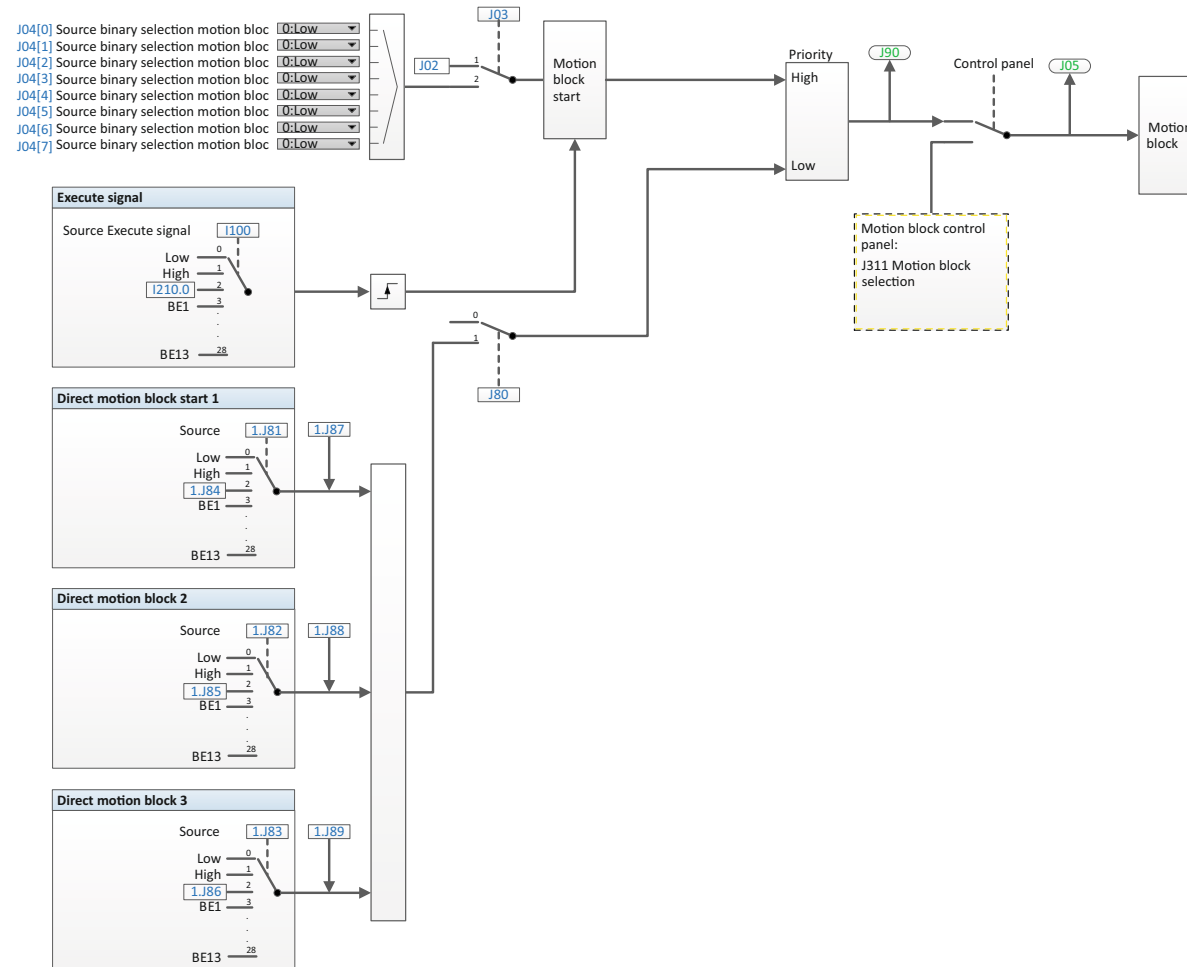
Every time a drive controller is restarted, it checks whether the projected motor, the brake (if applicable), the motor temperature sensor or commutation have been changed. If so, the changed data is read out. The optimizations saved in the drive controller are maintained whenever possible.

The automatic read-out of the electronic nameplate is activated at the factory (B04 Electronic nameplate = 64).

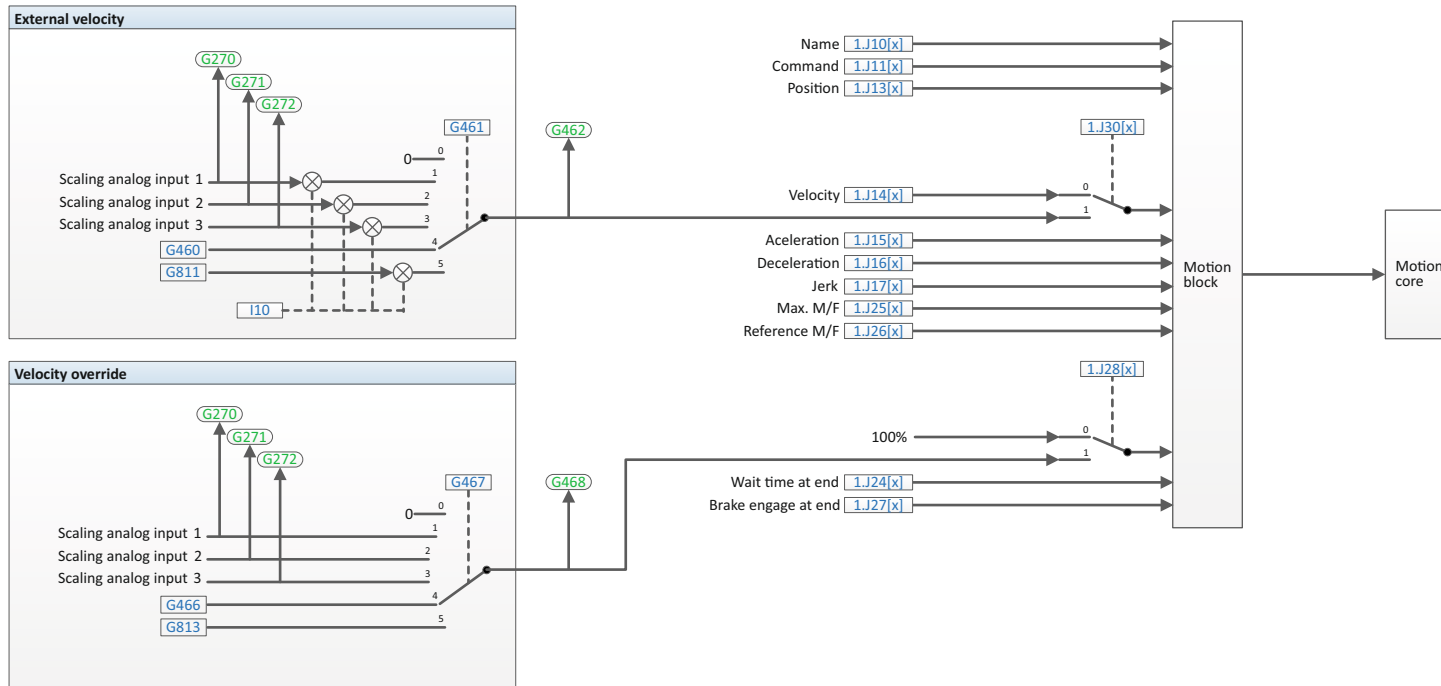
5.9 Operating modes – Block diagrams

5.9.1 Motion block

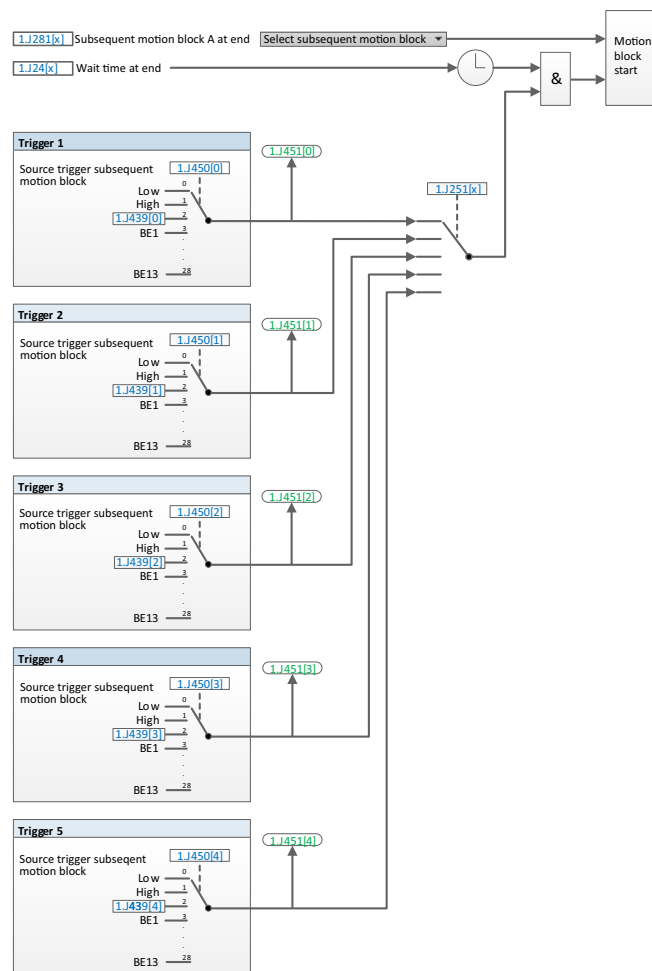
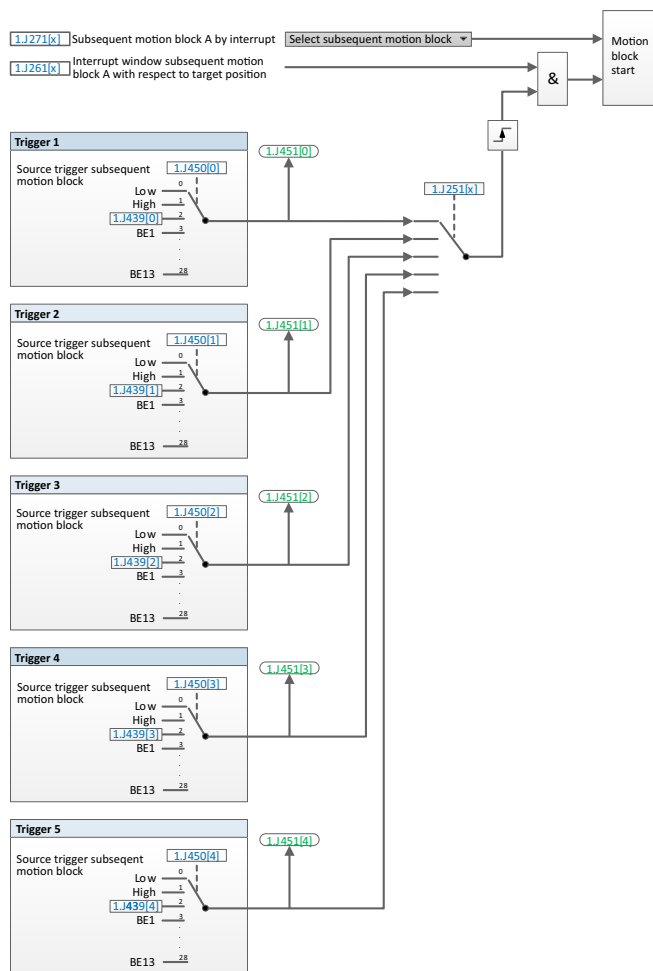
Motion block selection



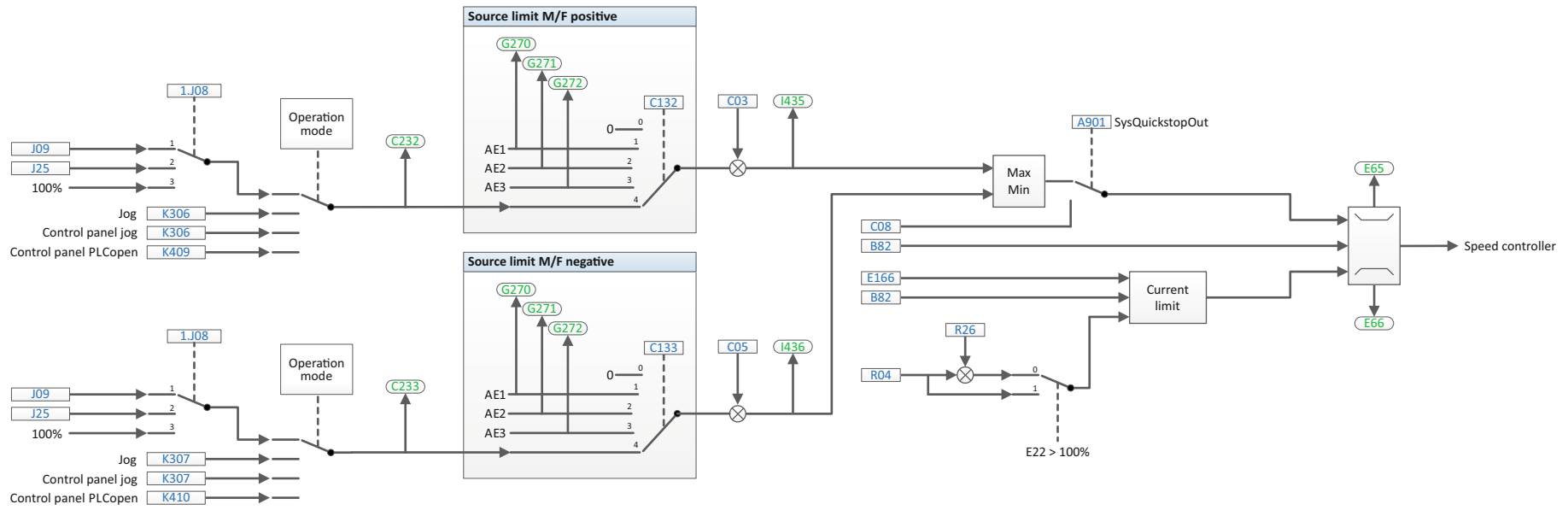
External velocity, velocity override



Motion block continuation

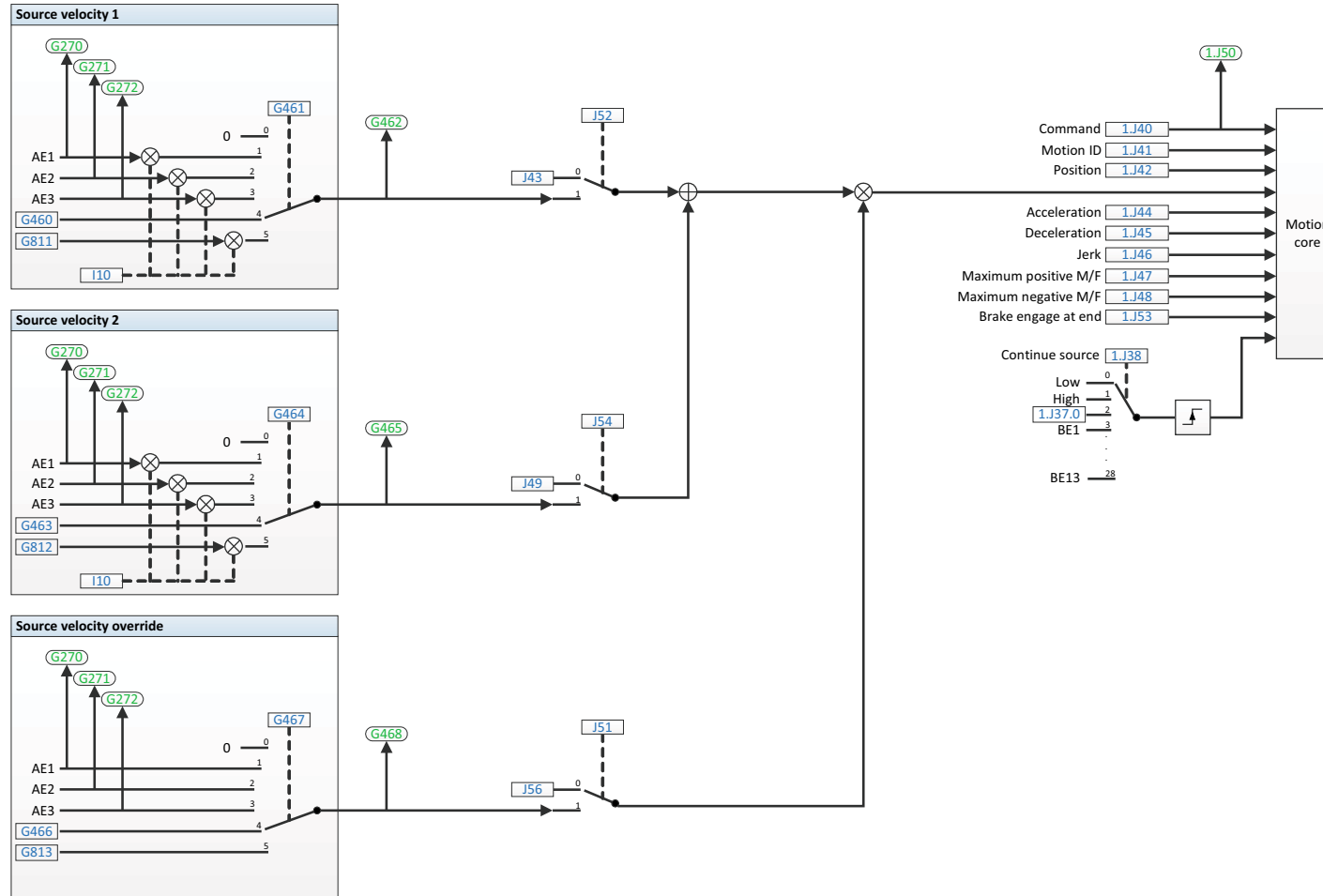


Torque/force limit

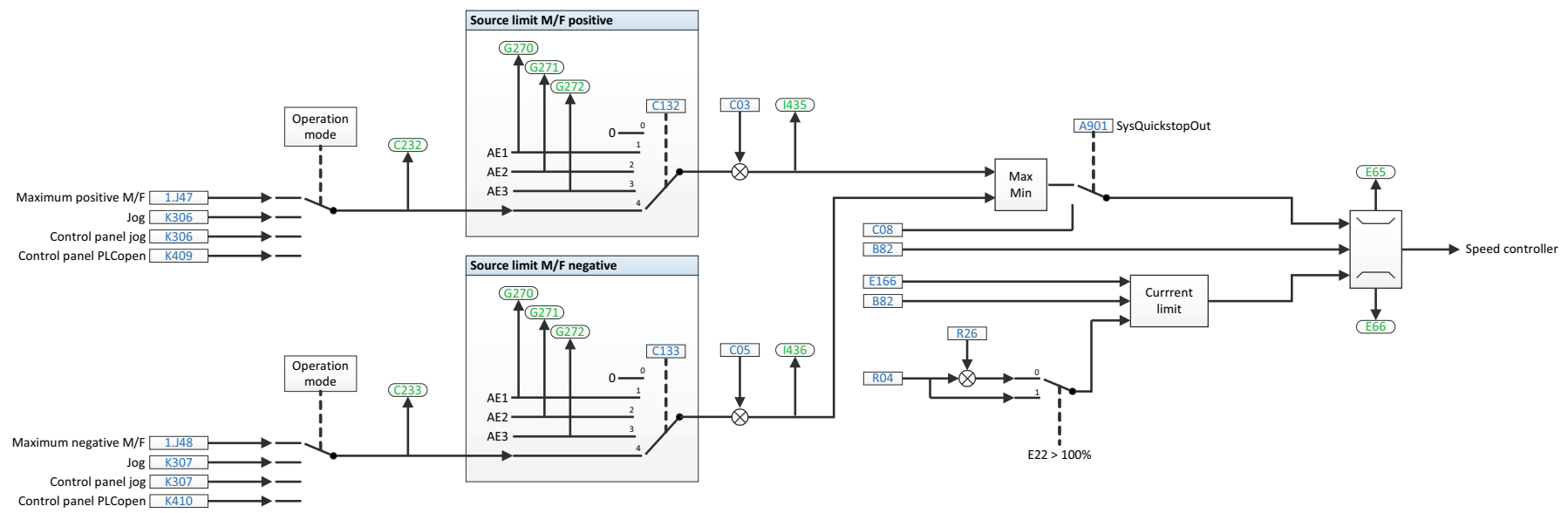


5.9.2 Command

Velocity, velocity override



Torque/force limit



5.10 Cycle times

Type	Cycle times	Relevant parameters
SD6: Application	1 ms, 2 ms, 4 ms, 8 ms	Adjustable in A150
SD6: Analog and digital inputs and outputs	1 ms, 2 ms, 4 ms, 8 ms	Adjustable in A150
SC6, SI6: Application	250 μ s, 500 μ s, 1 ms, 2 ms, 4 ms, 8 ms	Adjustable in A150
SC6, SI6: Digital inputs	250 μ s, 500 μ s, 1 ms, 2 ms, 4 ms, 8 ms	Adjustable in A150

Tab. 7: Cycle times

6 Appendix

6.1 Standard mapping of PROFINET and Drive Based synchronous

6.1.1 SC6, SI6: RxPZD

The following tables shows the PROFINET standard RxPZD mapping of the Drive Based and Drive Based synchronous applications. The mapping is defined in parameter array A90 for axis A and in A91 for axis B and can be adapted individually.

Byte	Data type	Name	Parameters
0	BYTE	Control byte device	1.A180
1	BYTE	Control byte command	1.J37
2 – 3	WORD	Control word application	1.I210
4	SINT	Command	1.J40
5	SINT	Motion-ID	1.J41
6 – 9	DINT	Position	1.J42
10– 13	REAL	Velocity 1	1.J43
14 – 17	REAL	Velocity Override	1.J56
18 – 21	REAL	Torque/Force reference	1.G469
22	BYTE	Control byte motion block	1.J01
23 – 24	INT	Reference motion block	1.J02

Tab. 8: SC6, SI6: PROFINET standard RxPZD mapping; first part, axis A

Byte	Data type	Name	Parameters
0	BYTE	Control byte device	2.A180
1	BYTE	Control byte command	2.J37
2 – 3	WORD	Control word application	2.I210
4	SINT	Command	2.J40
5	SINT	Motion-ID	2.J41
6 – 9	DINT	Position	2.J42
10– 13	REAL	Velocity 1	2.J43
14 – 17	REAL	Velocity Override	2.J56
18 – 21	REAL	Torque/Force reference	2.G469
22	BYTE	Control byte motion block	2.J01
23 – 24	INT	Reference motion block	2.J02

Tab. 9: SC6, SI6: PROFINET standard RxPZD mapping; second part, axis B

6.1.2 SC6, SI6: TxPZD

The following tables show the PROFINET standard TxPZD mapping of the Drive Based and Drive Based synchronous applications. The mapping is defined in parameter array A94 for axis A and in A95 for axis B and can be adapted individually.

Byte	Data type	Name	Parameters
0	BYTE	Status byte device	1.E200[0]
1	BYTE	Status byte device	1.E200[1]
2 – 3	WORD	Status word 2	1.E201
4	BYTE	Status byte application	1.I212
5	BYTE	Status byte command	1.J39
6 – 7	WORD	Status word application	1.I200
8 – 11	DINT	Current position	1.I80
12 – 15	REAL	Actual speed	1.I88
16 – 19	REAL	Actual torque/force	1.E90
20 – 21	WORD	Status word user-defined	1.A67
22	SINT	Operating condition	1.E80
23	BYTE	Status byte motion block	1.J302
24 – 25	INT	Actual motion block	1.J300
26	SINT	Device control state	1.E48

Tab. 10: SC6, SI6: PROFINET standard TxPZD mapping; first part, axis A

Byte	Data type	Name	Parameters
0	BYTE	Status byte device	2.E200[0]
1	BYTE	Status byte device	2.E200[1]
2 – 3	WORD	Status word 2	2.E201
4	BYTE	Status byte application	2.I212
5	BYTE	Status byte command	2.J39
6 – 7	WORD	Status word application	2.I200
8 – 11	DINT	Current position	2.I80
12 – 15	REAL	Actual speed	2.I88
16 – 19	REAL	Actual torque/force	2.E90
20 – 21	WORD	Status word user-defined	2.A67
22	SINT	Operating condition	2.E80
23	BYTE	Status byte motion block	2.J302
24 – 25	INT	Actual motion block	2.J300
26	SINT	Device control state	2.E48

Tab. 11: SC6, SI6: PROFINET standard TxPZD mapping; second part, axis B

6.1.3 SD6: RxPZD

The following table shows the PROFINET standard RxPZD mapping of the Drive Based and Drive Based synchronous applications. The mapping is defined in the parameter arrays A90 – A92 and can be adapted individually.

Byte	Data type	Name	Parameters
0	BYTE	Control byte device	A180
1	BYTE	Control byte command	1.J37
2 – 3	WORD	Control word application	I210
4	SINT	Command	1.J40
5	SINT	Motion-ID	1.J41
6 – 9	DINT	Position	1.J42
10– 13	REAL	Velocity 1	1.J43
14 – 17	REAL	Velocity Override	1.J56
18 – 21	REAL	Torque/Force reference	1.G469
22	BYTE	Control byte motion block	J01
23 – 24	INT	Reference motion block	J02

Tab. 12: SD6: PROFINET standard RxPZD mapping

6.1.4 SD6: TxPZD

The following table shows the PROFINET standard TxPZD mapping of the Drive Based and Drive Based synchronous applications. The mapping is defined in the parameter arrays A94 – A96 and can be adapted individually.

Byte	Data type	Name	Parameters
0	BYTE	Status byte device	E200[0]
1	BYTE	Status byte device	E200[1]
2 – 3	WORD	Status word 2	E201
4	BYTE	Status byte application	1.I212
5	BYTE	Status byte command	1.J39
6 – 7	WORD	Status word application	I200
8 – 11	DINT	Current position	1.I80
12 – 15	REAL	Actual speed	1.I88
16 – 19	REAL	Actual torque/force	E90
20 – 21	WORD	Status word user-defined	A67
22	SINT	Operating condition	E80
23	BYTE	Status byte motion block	J302
24 – 25	INT	Actual motion block	J300
26	SINT	Device control state	E48

Tab. 13: SD6: PROFINET standard TxPZD mapping

6.2 Detailed information

The documentation listed in the following table offers additional relevant information on the drive controllers of the 6th STOBER drive controller generation.

Current document versions can be found at <http://www.stoeber.de/en/downloads/>.

Device/Software	Documentation	Contents	ID
SD6 drive controller	Manual	System design, technical data, project configuration, storage, installation, connection, commissioning, operation, service, diagnostics	442426
PROFINET communication – SD6	Manual	Installation, electrical installation, data transfer, commissioning, detailed information	442710
EtherCAT communication – SD6	Manual	Installation, electrical installation, data transfer, commissioning, detailed information	442516
CANopen communication – SD6	Manual	Installation, electrical installation, data transfer, commissioning, detailed information	442637
SC6 drive controller	Manual	System design, technical data, project configuration, storage, installation, connection, commissioning, operation, service, diagnostics	442790
Multi-axis drive system with SI6 and PS6	Manual	System design, technical data, project configuration, storage, installation, connection, commissioning, operation, service, diagnostics	442728
PROFINET communication – SC6, SI6	Manual	Installation, electrical installation, data transfer, commissioning, detailed information	443039
EtherCAT communication – SC6, SI6	Manual	Installation, electrical installation, data transfer, commissioning, detailed information	443025
CiA 402 application – SC6, SI6	Manual	Project planning, configuration, parameterization, function test, detailed information	443080

6.3 Abbreviations

Abbreviation	Meaning
AI	Analog Input
ALT	Alternative
CAN	Controller Area Network
CB	Controller Based
CiA	CAN in Automation
CNC	Computerized Numerical Control
DI	Digital Input
EtherCAT	Ethernet for Control Automation Technology
IDT	Internal data transfer mechanism
IGB	Integrated Bus
LinM	Linear motor
LS	Limit Switch
LSB	Least Significant Bit
M/F	Torque or force
MEnc	Motor encoder
PEnc	Position encoder
PROFINET	Process Field Network
RS	Reference Switch
PLC	Programmable Logic Controller
VM	Virtual Master
ZP	Zero Pulse

7 Contact

7.1 Consultation, service and address

We would be happy to help you!

We offer a wealth of information and services to go with our products on our website:

<http://www.stoeber.de/en/service>

For additional or personalized information, contact our consultation and support service:

<http://www.stoeber.de/en/support>

Do you need our first-level support?:

Phone +49 7231 582-3060

applications@stoeber.de

Do you need a replacement device?:

Phone +49 7231 582-1128

replace@stoeber.de

Call our 24-hour service hotline:

Phone +49 7231 582-3000

Our address:

STÖBER Antriebstechnik GmbH + Co. KG

Kieselbronner Strasse 12

75177 Pforzheim, Germany

7.2 Your opinion is important to us

We created this documentation to the best of our knowledge with the goal of helping you build and expand your expertise productively and efficiently with our products.

Your suggestions, opinions, wishes and constructive criticism help us to ensure and further develop the quality of our documentation.

If you want to contact us for a specific reason, we would be happy to receive an e-mail from you at:

documentation@stoeber.de

Thank you for your interest.

Your STOBER editorial team

7.3 Close to customers around the world

We offer you committed, expert advice and support in over 40 countries worldwide:

STOBER AUSTRIA www.stoeber.at Phone +43 7613 7600-0 sales@stoeber.at	STOBER SOUTH EAST ASIA www.stober.sg sales@stober.sg
STOBER CHINA www.stoeber.cn Phone +86 512 5320 8850 sales@stoeber.cn	STOBER SWITZERLAND www.stoeber.ch Phone +41 56 496 96 50 sales@stoeber.ch
STOBER FRANCE www.stober.fr Phone +33 4 78.98.91.80 sales@stober.fr	STOBER TAIWAN www.stober.tw Phone +886 4 2358 6089 sales@stober.tw
STOBER ITALY www.stober.it Phone +39 02 93909570 sales@stober.it	STOBER TURKEY www.stober.com Phone +90 216 510 2290 sales-turkey@stober.com
STOBER JAPAN www.stober.co.jp Phone +81 3 5395 678 8 sales@stober.co.jp	STOBER UNITED KINGDOM www.stober.co.uk Phone +44 1543 458 858 sales@stober.co.uk
STOBER USA www.stober.com Phone +1 606 759 5090 sales@stober.com	

List of figures

Fig. 1	DS6: Program interface	10
Fig. 2	DriveControlSuite: Navigation using text links and symbols	12
Fig. 3	Synchronous operation: Model 1	31
Fig. 4	Synchronous operation: Model 2	38
Fig. 5	Synchronous operation: Model 3	42
Fig. 6	Motion block operating mode: Signal flows	62
Fig. 7	Command operating mode: Signal flows	80
Fig. 8	Components and configuration steps	85
Fig. 9	Endless rotational drive	89
Fig. 10	Limited rotational drive	89
Fig. 11	Endless translational drive	90
Fig. 12	Limited translational drive	90
Fig. 13	Limited translational drive – Linear motor	91
Fig. 14	Hardware limit switch memory	95
Fig. 15	Drive Based device state machine: Possible operating states and state change	126

List of tables

Tab. 1	Parameter groups	13
Tab. 2	Parameters – Data types, styles, possible values	14
Tab. 3	Parameter types.....	15
Tab. 4	Indirect trigger: Notations.....	74
Tab. 5	Referencing methods	100
Tab. 6	Drive Based synchronous: Available motion commands	131
Tab. 7	Cycle times: SD6	139
Tab. 8	SC6, SI6: PROFINET standard RxPZD mapping; first part, axis A	140
Tab. 9	SC6, SI6: PROFINET standard RxPZD mapping; second part, axis B.....	140
Tab. 10	SC6, SI6: PROFINET standard TxPZD mapping; first part, axis A.....	141
Tab. 11	SC6, SI6: PROFINET standard TxPZD mapping; second part, axis B.....	141
Tab. 12	SD6: PROFINET standard RxPZD mapping	142
Tab. 13	SD6: PROFINET standard TxPZD mapping	142

Glossary

Broadcast domain

Logical grouping of network devices within a local network that reaches all nodes via broadcast.

Control cascade

Complete model of the control structure with the position controller, velocity controller and current controller components.

Electronic nameplate

STOBER synchronous servo motors are generally equipped with absolute encoders that provide special memory. This memory includes the electronic nameplate, i.e. all type-relevant master data as well as special mechanical and electronic values of a motor. When you operate a drive controller with a STOBER synchronous servo motor and an absolute encoder, the electronic nameplate is read and all motor data transferred if the drive controller is connected online. The drive controller automatically determines the associated limit values and control parameters from this data.

IPv4 limited broadcast

Type of broadcast in a network with IPv4 (Internet Protocol version 4). The IP address 255.255.255.255 is entered as the destination. The content of the broadcast is not redirected by a router, which limits it to the local network.

PID controller

Universal controller type with a P-, I- and D-share. These three adjustment parameters make the controller flexible and ensure exact and highly dynamic control. However, by implication, it also necessitates a wide variety of variants. It is that much more important to ensure careful construction that is well-coordinated to the system. The application areas for this controller type are control circuits with systems of the second order and higher, which must be stabilized quickly and do not allow for any permanent control deviation.

Receive process data

Output data that a node receives with RxPDOs.



4 4 3 0 4 6 . 0 2

02/2021

STÖBER Antriebstechnik GmbH + Co. KG
Kieselbronner Str. 12
75177 Pforzheim
Germany
Tel. +49 7231 582-0
mail@stoeber.de
www.stober.com

24 h Service Hotline
+49 7231 582-3000



STÖBER

www.stober.com