

Edition 11/2023

FUNCTION MANUAL

SIMATIC

S7-1500

S7-1500/S7-1500T Motion Control Overview V8.0 as of STEP 7 V19

SIMATIC

S7-1500

S7-1500/S7-1500T Motion Control Overview V8.0 as of STEP 7 V19

Diagnostics Manual




Introduction (S7-1500, S7-1500T)	1
Safety instructions (S7-1500, S7-1500T)	2
Overview of functions (S7-1500, S7-1500T)	3
New features (S7-1500, S7-1500T)	4
Using versions (S7-1500, S7-1500T)	5
Configuring (S7-1500, S7-1500T)	6
Programming (S7-1500, S7-1500T)	7
Downloading to CPU (S7-1500, S7-1500T)	8
Diagnostics (S7-1500, S7-1500T)	9

S7-1500/S7-1500T Motion Control

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

 DANGER
indicates that death or severe personal injury will result if proper precautions are not taken.
 WARNING
indicates that death or severe personal injury may result if proper precautions are not taken.
 CAUTION
indicates that minor personal injury can result if proper precautions are not taken.
NOTICE
indicates that property damage can result if proper precautions are not taken.


If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

 WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

1	Introduction (S7-1500, S7-1500T).....	7
1.1	S7-1500 Motion Control Documentation Guide (S7-1500, S7-1500T).....	8
1.2	Function Manuals documentation guide (S7-1500, S7-1500T).....	9
1.2.1	Information classes Function Manuals (S7-1500, S7-1500T).....	9
1.2.2	Basic tools (S7-1500, S7-1500T).....	11
1.2.3	SIMATIC Technical Documentation (S7-1500, S7-1500T).....	13
2	Safety instructions (S7-1500, S7-1500T).....	15
2.1	Cybersecurity information (S7-1500, S7-1500T).....	15
3	Overview of functions (S7-1500, S7-1500T).....	16
3.1	Areas of application of S7-1500 Motion Control (S7-1500, S7-1500T).....	16
3.2	Functions in STEP 7 (S7-1500, S7-1500T).....	17
3.3	Principle of operation of S7-1500 Motion Control (S7-1500, S7-1500T).....	18
3.4	Description of technology objects (S7-1500, S7-1500T).....	19
3.5	HW Configuration (S7-1500, S7-1500T).....	21
3.6	Configuration limits (S7-1500, S7-1500T).....	22
4	New features (S7-1500, S7-1500T).....	24
4.1	New features V8.0 (S7-1500, S7-1500T).....	24
4.1.1	New general functions V8.0 (S7-1500, S7-1500T).....	24
4.1.2	New axis functions V8.0 (S7-1500, S7-1500T).....	25
4.1.3	New measuring input and output cam functions V8.0 (S7-1500, S7-1500T).....	26
4.1.4	New synchronous operation functions V8.0 (S7-1500, S7-1500T).....	26
4.1.5	New kinematics functions V8.0 (S7-1500, S7-1500T).....	27
4.1.6	Interpreter functions V8.0 (S7-1500, S7-1500T).....	28
4.2	New features V7.0 (S7-1500, S7-1500T).....	28
4.2.1	New general functions V7.0 (S7-1500, S7-1500T).....	28
4.2.2	New axis functions V7.0 (S7-1500, S7-1500T).....	30
4.2.3	New synchronous operation functions V7.0 (S7-1500, S7-1500T).....	31
4.2.4	New kinematics functions V7.0 (S7-1500, S7-1500T).....	32
4.3	Innovations V6.0 (S7-1500, S7-1500T).....	34
4.4	Innovations V5.0 (S7-1500, S7-1500T).....	36
4.5	Innovations V4.0 (S7-1500, S7-1500T).....	38
4.6	Innovations V3.0 (S7-1500, S7-1500T).....	39

5	Using versions (S7-1500, S7-1500T).....	40
5.1	Overview of versions (S7-1500, S7-1500T).....	40
5.2	Replacing devices (S7-1500, S7-1500T).....	43
5.3	Upgrade technology version (S7-1500, S7-1500T).....	44
5.4	Upgrading to technology version V8.0 (S7-1500, S7-1500T).....	44
5.5	Upgrading to technology version V7.0 (S7-1500, S7-1500T).....	45
5.6	Upgrade to technology version V3.0 to V6.0 (S7-1500, S7-1500T).....	47
5.7	Technology versions V1.0 and V2.0 (S7-1500).....	48
6	Configuring (S7-1500, S7-1500T).....	51
6.1	Basics on the configuration of technology objects (S7-1500, S7-1500T).....	51
6.1.1	Add technology object (S7-1500, S7-1500T).....	51
6.1.2	Copy technology object (S7-1500, S7-1500T).....	52
6.1.3	Delete technology object (S7-1500, S7-1500T).....	53
6.1.4	Group technology objects (S7-1500, S7-1500T).....	54
6.1.5	Working with the configuration editor (S7-1500, S7-1500T).....	54
6.1.6	Toolbar of the configuration (S7-1500, S7-1500T).....	55
6.2	Compare values (S7-1500, S7-1500T).....	55
6.3	Working with the parameter view (S7-1500, S7-1500T).....	56
6.3.1	Structure of the parameter view (S7-1500, S7-1500T).....	57
6.3.1.1	Toolbar (S7-1500, S7-1500T).....	57
6.3.1.2	Navigation (S7-1500, S7-1500T).....	58
6.3.1.3	Parameter table (S7-1500, S7-1500T).....	58
6.3.2	Opening the parameter view (S7-1500, S7-1500T).....	59
6.3.3	Working with the parameter view (S7-1500, S7-1500T).....	60
6.3.3.1	Filtering the parameter table (S7-1500, S7-1500T).....	60
6.3.3.2	Sorting the parameter table (S7-1500, S7-1500T).....	61
6.3.3.3	Indicating errors (S7-1500, S7-1500T).....	62
6.3.3.4	Editing start values in the project (S7-1500, S7-1500T).....	62
6.3.3.5	Monitoring values online in the parameter view (S7-1500, S7-1500T).....	64
6.3.3.6	Modifying values (S7-1500, S7-1500T).....	64
6.3.3.7	Comparing values (S7-1500, S7-1500T).....	65
6.4	Configuring technological modules and onboard I/O for Motion Control (S7-1500, S7-1500T).....	67
6.4.1	TM Count 1x24V/TM Count 2x24V (S7-1500, S7-1500T).....	68
6.4.2	TM PosInput 1/TM PosInput 2 (S7-1500, S7-1500T).....	70
6.4.3	TM Timer DIDQ 10x24V/TM Timer DIDQ 16x24V (S7-1500, S7-1500T).....	72
6.4.4	TM Pulse 2x24V (S7-1500, S7-1500T).....	74
6.4.5	TM PTO 2x24V (S7-1500, S7-1500T).....	75
6.4.6	TM PTO 4 (S7-1500, S7-1500T).....	77
6.4.7	CPU 1511C-1 PN/CPU 1512C-1 PN (onboard I/O devices) (S7-1500, S7-1500T).....	79
6.4.8	SIMATIC Drive Controller (onboard I/O) (S7-1500, S7-1500T).....	85
7	Programming (S7-1500, S7-1500T).....	86
7.1	Process response (S7-1500, S7-1500T).....	86

7.1.1	Organization blocks for Motion Control (S7-1500, S7-1500T).....	86
7.1.2	Process image partition "OB Servo PIP" (S7-1500, S7-1500T).....	89
7.1.3	Operational Sequence and Timeouts (S7-1500, S7-1500T).....	90
7.1.4	Operating modes (S7-1500, S7-1500T).....	94
7.2	Technology data block (S7-1500, S7-1500T).....	96
7.2.1	Evaluating the technology data block (S7-1500, S7-1500T).....	96
7.2.2	Evaluate StatusWord, ErrorWord and WarningWord (S7-1500, S7-1500T).....	98
7.2.3	Change restart-relevant data (S7-1500, S7-1500T).....	99
7.3	Motion Control instructions (S7-1500, S7-1500T).....	100
7.3.1	Motion Control instruction parameters (S7-1500, S7-1500T).....	100
7.3.2	Add Motion Control instructions (S7-1500, S7-1500T).....	103
7.3.3	Parameter transfer for function blocks (S7-1500, S7-1500T).....	105
7.4	Starting Motion Control jobs (S7-1500, S7-1500T).....	109
7.5	Tracking active jobs (S7-1500, S7-1500T).....	110
7.5.1	Motion Control instructions with "Done" parameter (S7-1500, S7-1500T).....	110
7.5.2	Motion Control instructions without "Done" parameter (S7-1500, S7-1500T).....	115
7.5.3	Motion Control instruction "MC_MoveJog" (S7-1500, S7-1500T).....	119
7.6	Ending Motion Control jobs (S7-1500, S7-1500T).....	121
7.7	Restart of technology objects (S7-1500, S7-1500T).....	123
7.8	Data types (S7-1500, S7-1500T).....	124
8	Downloading to CPU (S7-1500, S7-1500T).....	125
9	Diagnostics (S7-1500, S7-1500T).....	126
	Glossary.....	127
	Index.....	131

Introduction (S7-1500, S7-1500T)

Purpose of the documentation

This documentation provides important information that you need to configure and commission the integrated Motion Control functionality of the S7-1500 Automation systems.

Required basic knowledge

In order to understand this documentation, the following knowledge is required:

- General knowledge in the field of automation
- General knowledge in the field of drive engineering and motion control

Validity of the documentation

This documentation is valid for the S7-1500 product range.

Conventions

- The path specifications in the project tree assume that the "Technology objects" folder is open in the subtree of the CPU. The "Technology object" placeholder represents the name of the technology object.
Example: "Technology object > Configuration > Basic parameters".
- The <TO> placeholder represents the name set in tags for the respective technology object.
Example: <TO>.Actor.Type
- This documentation contains pictures of the devices described. The pictures may differ in minor details from the devices supplied.

You should also observe the notes that are marked as follows:

NOTE

A note contains important information about the product described in the documentation, about the handling of the product, and about sections in this documentation demanding your particular attention.

Industry Mall

The Industry Mall is the catalog and ordering system of Siemens AG for automation and drive solutions on the basis of Totally Integrated Automation (TIA) and Totally Integrated Power (TIP).

You can find catalogs for all automation and drive products on the Internet (<https://mall.industry.siemens.com>).

1.1 S7-1500 Motion Control Documentation Guide (S7-1500, S7-1500T)

Product information

Please also note the supplementary information on the Motion Control documentation:

- Product information on the S7-1500/1500T Motion Control documentation
<https://support.industry.siemens.com/cs/ww/en/view/109794046>
(<https://support.industry.siemens.com/cs/ww/en/view/109794046>)

Documentation

The documentation of the Motion Control functions is divided into the following documents:

- S7-1500/S7-1500T Motion Control Overview
<https://support.industry.siemens.com/cs/ww/en/view/109817883>
(<https://support.industry.siemens.com/cs/ww/en/view/109817883>)
This document describes the innovations in the technology versions, how to upgrade the technology version, functions that are used for all technology objects, and the operational sequence of Motion Control applications.
- S7-1500/S7-1500T Motion Control alarms and error IDs
<https://support.industry.siemens.com/cs/ww/en/view/109817890>
(<https://support.industry.siemens.com/cs/ww/en/view/109817890>)
This document describes the technology alarms of the technology objects and the error identifications of the Motion Control instructions.
- S7-1500/S7-1500T Axis functions
<https://support.industry.siemens.com/cs/ww/en/view/109817884>
(<https://support.industry.siemens.com/cs/ww/en/view/109817884>)
This document describes the drive and encoder connection and functions for single-axis movements.
- S7-1500/S7-1500T Synchronous operation functions
<https://support.industry.siemens.com/cs/ww/en/view/109817888>
(<https://support.industry.siemens.com/cs/ww/en/view/109817888>)
This document describes gearing, velocity synchronous operation and camming as well as cross-PLC synchronous operation.
- S7-1500/S7-1500T Measuring input and output cam functions
<https://support.industry.siemens.com/cs/ww/en/view/109817889>
(<https://support.industry.siemens.com/cs/ww/en/view/109817889>)
This document describes the detection of the actual position via a measuring input and the output of switching signals via output cam or cam track.
- S7-1500T kinematics functions
<https://support.industry.siemens.com/cs/ww/en/view/109817886>
(<https://support.industry.siemens.com/cs/ww/en/view/109817886>)
This document describes the control of kinematics with up to 6 interpolating axes.
- S7-1500T Interpreter functions
<https://support.industry.siemens.com/cs/ww/en/view/109817891>
(<https://support.industry.siemens.com/cs/ww/en/view/109817891>)
This documentation describes the control of technology objects via an interpreter program.

See also

Topic page "SIMATIC Technology - Motion Control: Overview and Important Links"

<https://support.industry.siemens.com/cs/ww/en/view/109751049>

(<https://support.industry.siemens.com/cs/ww/en/view/109751049>)

1.2 Function Manuals documentation guide (S7-1500, S7-1500T)

1.2.1 Information classes Function Manuals (S7-1500, S7-1500T)



The documentation for the SIMATIC S7-1500 automation system, for the 1513/1516pro-2 PN, SIMATIC Drive Controller CPUs based on SIMATIC S7-1500 and the SIMATIC ET 200MP, ET 200SP, ET 200AL and ET 200eco PN distributed I/O systems is arranged into three areas.

This arrangement enables you to access the specific content you require.

You can download the documentation free of charge from the Internet

(<https://support.industry.siemens.com/cs/ww/en/view/109742705>).

Basic information



The system manuals and Getting Started describe in detail the configuration, installation, wiring and commissioning of the SIMATIC S7-1500, SIMATIC Drive Controller, ET 200MP, ET 200SP, ET 200AL and ET 200eco PN systems. Use the corresponding operating instructions for 1513/1516pro-2 PN CPUs.

The STEP 7 online help supports you in the configuration and programming.

Examples:

- Getting Started S7-1500
- System manuals
- Operating instructions ET 200pro and 1516pro-2 PN CPU
- Online help TIA Portal

Device information



Equipment manuals contain a compact description of the module-specific information, such as properties, wiring diagrams, characteristics and technical specifications.

Examples:

- Equipment manuals for CPUs
- Equipment manuals for interface modules
- Equipment manuals for digital modules
- Equipment manuals for analog modules
- Equipment manuals for communication modules
- Equipment manuals for technology modules
- Equipment manuals for power supply modules
- Equipment manuals for BaseUnits

General information



The function manuals contain detailed descriptions on general topics relating to the SIMATIC Drive Controller and the S7-1500 automation system.

Examples:

- Function Manual Diagnostics
- Function Manual Communication
- Function Manuals Motion Control
- Function Manual Web Server
- Function Manual Cycle and Response Times
- PROFINET Function Manual
- PROFIBUS Function Manual

Product Information

Changes and supplements to the manuals are documented in a Product Information. The Product Information takes precedence over the device and system manuals.

You will find the latest Product Information on the Internet:

- S7-1500/ET 200MP (<https://support.industry.siemens.com/cs/de/en/view/68052815>)
- SIMATIC Drive Controller (<https://support.industry.siemens.com/cs/de/en/view/109772684/en>)
- Motion Control (<https://support.industry.siemens.com/cs/de/en/view/109794046/en>)
- ET 200SP (<https://support.industry.siemens.com/cs/de/en/view/73021864>)
- ET 200eco PN (<https://support.industry.siemens.com/cs/ww/en/view/109765611>)

Manual Collections

The Manual Collections contain the complete documentation of the systems put together in one file.

You will find the Manual Collections on the Internet:

- S7-1500/ET 200MP/SIMATIC Drive Controller (<https://support.industry.siemens.com/cs/ww/en/view/86140384>)
- ET 200SP (<https://support.industry.siemens.com/cs/ww/en/view/84133942>)
- ET 200AL (<https://support.industry.siemens.com/cs/ww/en/view/95242965>)
- ET 200eco PN (<https://support.industry.siemens.com/cs/ww/en/view/109781058>)

1.2.2 Basic tools (S7-1500, S7-1500T)

Tools

The tools described below support you in all steps: from planning, over commissioning, all the way to analysis of your system.

TIA Selection Tool

The TIA Selection Tool tool supports you in the selection, configuration, and ordering of devices for Totally Integrated Automation (TIA).

As successor of the SIMATIC Selection Tools, the TIA Selection Tool assembles the already known configurators for automation technology into a single tool.

With the TIA Selection Tool, you can generate a complete order list from your product selection or product configuration.

You can find the TIA Selection Tool on the Internet.

<https://support.industry.siemens.com/cs/ww/en/view/109767888>

SIMATIC Automation Tool

You can use the SIMATIC Automation Tool to perform commissioning and maintenance activities on various SIMATIC S7 stations as bulk operations independent of TIA Portal.

The SIMATIC Automation Tool offers a wide range of functions:

- Scanning of a PROFINET/Ethernet system network and identification of all connected CPUs
- Assignment of addresses (IP, subnet, Gateway) and device name (PROFINET device) to a CPU
- Transfer of the date and the programming device/PC time converted to UTC time to the module
- Program download to CPU
- RUN/STOP mode switchover
- CPU localization through LED flashing
- Reading out of CPU error information
- Reading the CPU diagnostic buffer
- Reset to factory settings
- Firmware update of the CPU and connected modules

You can find the SIMATIC Automation Tool on the Internet.

<https://support.industry.siemens.com/cs/ww/en/view/98161300>

PRONETA

SIEMENS PRONETA (PROFINET network analysis) is a commissioning and diagnostic tool for PROFINET networks. PRONETA Basic has two core functions:

- In the network analysis, you get an overview of the PROFINET topology. Compare a real configuration with a reference installation or make simple parameter changes, e.g. to the names and IP addresses of the devices.
- The "IO test" is a simple and rapid test of the wiring and the module configuration of a plant, including documentation of the test results.

You can find SIEMENS PRONETA Basic on the Internet:

(<https://support.industry.siemens.com/cs/ww/en/view/67460624>)

SIEMENS PRONETA Professional is a licensed product that offers you additional functions. It offers you simple asset management in PROFINET networks and supports operators of automation systems in automatic data collection/acquisition of the components used through various functions:

- The user interface (API) offers an access point to the automation cell to automate the scan functions using MQTT or a command line.
- With PROFlenergy diagnostics, you can quickly detect the current pause mode or the readiness for operation of devices that support PROFlenergy and change these as needed.
- The data record wizard supports PROFINET developers in reading and writing acyclic PROFINET data records quickly and easily without PLC and engineering.

You can find SIEMENS PRONETA Professional on the Internet.

(<https://www.siemens.com/proneta-professional>)

SINETPLAN

SINETPLAN, the Siemens Network Planner, supports you in planning automation systems and networks based on PROFINET. The tool facilitates professional and predictive dimensioning of your PROFINET installation as early as in the planning stage. In addition, SINETPLAN supports you during network optimization and helps you to exploit network resources optimally and to plan reserves. This helps to prevent problems in commissioning or failures during productive operation even in advance of a planned operation. This increases the availability of the production plant and helps improve operational safety.

The advantages at a glance

- Network optimization thanks to port-specific calculation of the network load
- Increased production availability thanks to online scan and verification of existing systems
- Transparency before commissioning through importing and simulation of existing STEP 7 projects
- Efficiency through securing existing investments in the long term and the optimal use of resources

You can find SINETPLAN on the Internet

(<https://new.siemens.com/global/en/products/automation/industrial-communication/profinet/sinetplan.html>).

1.2.3 SIMATIC Technical Documentation (S7-1500, S7-1500T)

Additional SIMATIC documents will complete your information. You can find these documents and their use at the following links and QR codes.

The Industry Online Support gives you the option to get information on all topics. Application examples support you in solving your automation tasks.

Overview of the SIMATIC Technical Documentation

Here you will find an overview of the SIMATIC documentation available in Siemens Industry Online Support:



Industry Online Support International

<https://support.industry.siemens.com/cs/ww/en/view/109742705>

Watch this short video to find out where you can find the overview directly in Siemens Industry Online Support and how to use Siemens Industry Online Support on your mobile device:



Quick introduction to the technical documentation of automation products per video <https://support.industry.siemens.com/cs/us/en/view/109780491>



YouTube video: Siemens Automation Products - Technical Documentation at a Glance <https://youtu.be/TwLSxxRQQsA>

Retention of the documentation

Retain the documentation for later use.

For documentation provided in digital form:

- Download the associated documentation after receiving your product and before initial installation/commissioning. Use the following download options:
 - Industry Online Support International: <https://support.industry.siemens.com>
The article number is used to assign the documentation to the product. The article number is specified on the product and on the packaging label. Products with new, non-compatible functions are provided with a new article number and documentation.
 - ID link:
Your product may have an ID link. The ID link is a QR code with a frame and a black frame corner at the bottom right. The ID link takes you to the digital nameplate of your product. Scan the QR code on the product or on the packaging label with a smartphone camera, barcode scanner, or reader app. Call up the ID link.
- Retain this version of the documentation.

Updating the documentation

The documentation of the product is updated in digital form. In particular in the case of function extensions, the new performance features are provided in an updated version.

1. Download the current version as described above via the Industry Online Support or the ID link.
2. Also retain this version of the documentation.

mySupport

With "mySupport" you can get the most out of your Industry Online Support.

Registration	You must register once to use the full functionality of "mySupport". After registration, you can create filters, favorites and tabs in your personal workspace.
Support requests	Your data is already filled out in support requests, and you can get an overview of your current requests at any time.
Documentation	In the Documentation area you can build your personal library.
Favorites	You can use the "Add to mySupport favorites" to flag especially interesting or frequently needed content. Under "Favorites", you will find a list of your flagged entries.
Recently viewed articles	The most recently viewed pages in mySupport are available under "Recently viewed articles".
CAX data	The CAX data area gives you access to the latest product data for your CAX or CAE system. You configure your own download package with a few clicks: <ul style="list-style-type: none"> • Product images, 2D dimension drawings, 3D models, internal circuit diagrams, EPLAN macro files • Manuals, characteristics, operating manuals, certificates • Product master data

You can find "mySupport" on the Internet. (<https://support.industry.siemens.com/My/ww/en>)

Application examples

The application examples support you with various tools and examples for solving your automation tasks. Solutions are shown in interplay with multiple components in the system - separated from the focus on individual products.

You can find the application examples on the Internet.
(<https://support.industry.siemens.com/cs/ww/en/ps/ae>)

Safety instructions (S7-1500, S7-1500T)

2.1 Cybersecurity information (S7-1500, S7-1500T)

Siemens provides products and solutions with industrial cybersecurity functions that support the secure operation of plants, systems, machines, and networks.

In order to protect plants, systems, machines, and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial cybersecurity concept. Siemens' products and solutions constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.

For more information on protective industrial cybersecurity measures for implementation, please visit (<https://www.siemens.com/global/en/products/automation/topic-areas/industrial-cybersecurity.html>).

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customers' exposure to cyber threats.

To stay informed about product updates at all times, subscribe to the Siemens Industrial Cybersecurity RSS Feed under (<https://new.siemens.com/global/en/products/services/cert.html>).

Overview of functions (S7-1500, S7-1500T)

3.1 Areas of application of S7-1500 Motion Control (S7-1500, S7-1500T)

Machines and production lines must increasingly be adapted to different formats, sizes, product types and production processes. S7-1500 Motion Control offers a precise, dynamic and easy-to-implement solution for open- and closed-loop control of electrically driven machine axes.

Drives with PROFIdrive capability, drives with analog setpoint interface and stepper motors are controlled by means of standardized Motion Control instructions in accordance with PLCopen. The axis control panel and comprehensive online and diagnostic functions support easy commissioning and optimization of drives. S7-1500 Motion Control is fully integrated into the system diagnostics of the S7-1500 CPU.

Supported CPUs

The following CPUs support the Motion Control functions:

- Advanced Controller S7-1500(F)/S7-1500T(F)
- Distributed Controller S7-1500SP (F)/S7-1500SP T(F)
- Software Controller S7-1507S (F), S7-1508S (F)/S7-1508S T(F)
- Open Controller S7-1515SP PC2 (F)/S7-1515SP PC2 T(F)
- Drive Controller S7-150xD TF

The S7-1500T Technology CPUs offer extended functions. The fail-safe S7-1500(T)F CPUs enable you to use safety functions additionally.

Motion tasks and functions

S7-1500 Motion Control supports the following functions:

Motion task/Function	S7-1500	S7-1500T
Moving S7-1500 Motion Control enables, for example, continuous motions in roller, belt, chain and other conveyors.		
Speed-controlled continuous motions	✓	✓
Connection of drives		
Torque limiting		
Positioning S7-1500 Motion Control enables precise, cyclic positioning for motion tasks calling for dynamic response and precision.		
Precise absolute and relative positioning (closed-loop position control)	✓	✓
High dynamic response		
Traversing range limits through HW and SW limit switches		

Motion task/Function	S7-1500	S7-1500T
Backlash compensation	✓	✓
Linear and rotation axes		
Switchover of active encoder	-	✓
Direct specification of motion setpoints		
Processing For continuous processing steps demanding accurate speed, torque and position, S7-1500 Motion Control enables operation of production machines with coordinated axis motions, e.g. flying saw, presses and winder applications.		
Gearing without specification of the synchronous position	✓	✓
Gearing with specification of the synchronous position	-	✓
Velocity gearing		
Camming		
Cross-PLC synchronous operation		
Handling S7-1500 Motion Control enables precise and efficient handling control with up to four mechanically-coupled axes, e.g. for palletizing and pick and place applications. The additional Motion Control package "S7-1500T Motion Control KinPlus" enables operation of up to six mechanically coupled axes.		
Calibration of coordinate systems and zones	-	✓
Moving a kinematics in a linear manner		
Moving a kinematics in a circular manner		
Moving a kinematics with a synchronous "point-to-point" motion		
Conveyor tracking		
Zone monitoring		

3.2 Functions in STEP 7 (S7-1500, S7-1500T)

TIA Portal

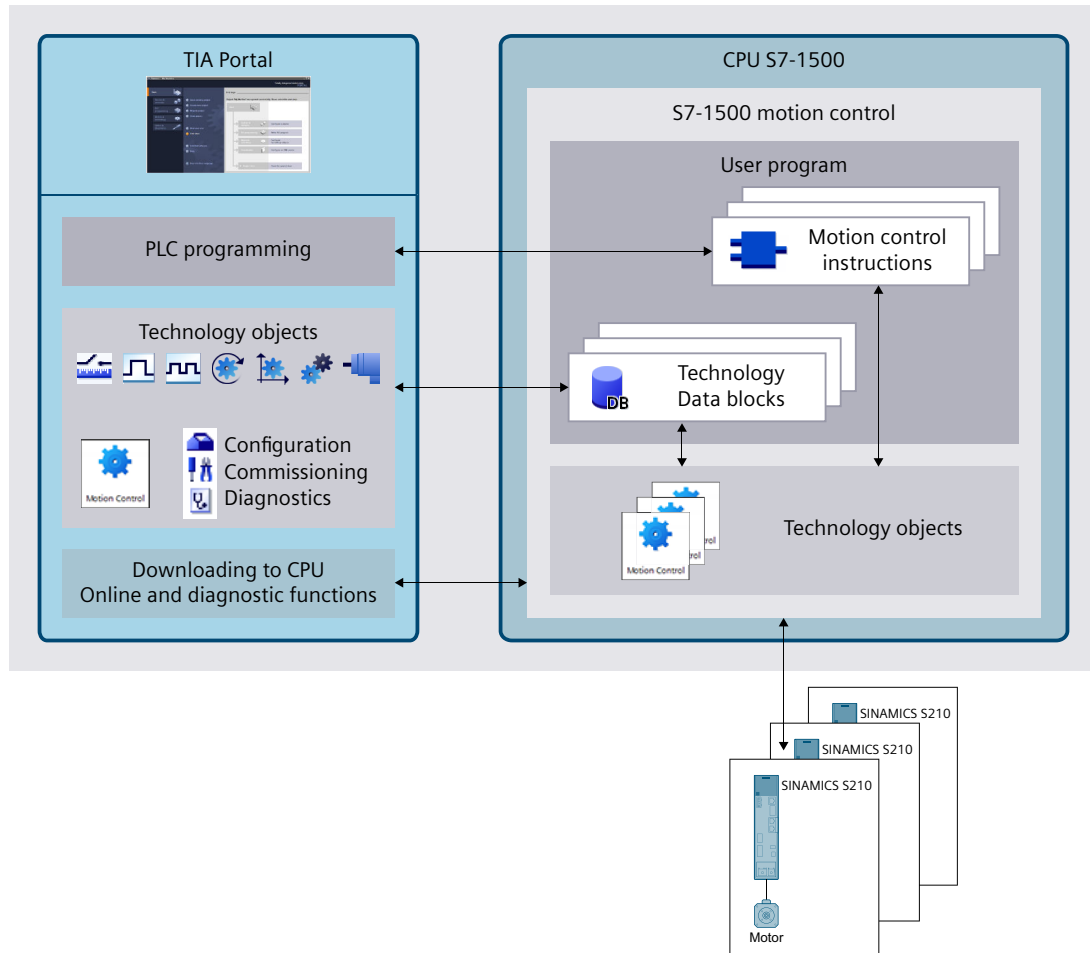
The TIA Portal supports you in the planning and commissioning of Motion Control functionality:

- Integrating and configuring the hardware
- Creating and configuring the technology objects
- Creating the user program
- Downloading to the CPU
- Commissioning the axes
- Optimizing the drives
- Testing and diagnostics

You use the TIA Portal to configure the hardware, the technology objects as well as your user program. You download the program you created to the CPU. You test your user program and diagnose the hardware with the online and diagnostic functions of the TIA Portal. Automatic data adaptation prevents configuration errors and input errors between components.

3.3 Principle of operation of S7-1500 Motion Control (S7-1500, S7-1500T)

You create a project, configure technology objects, and download the configuration to the CPU with the TIA Portal. The Motion Control functionality is processed in the CPU. You control the technology objects by using the Motion Control instructions in your user program. The TIA Portal provides additional functions for commissioning, optimization and diagnostics. The following figure provides a schematic representation of the user interfaces and the integration of Motion Control into the S7-1500 CPU. The concepts are then briefly explained:



Technology objects



Technology objects represent real objects in the controller, such as an axis. The configuration data of the technology objects mirrors the properties of the real object. You call the functions of the technology objects by means of Motion Control instructions in your user program. These functions are executed in the Motion Control organization blocks ([Page 86](#)) independently of the user program. The technology objects perform open- and closed-loop control of the motions of the real objects and report back status information, such as the current position.

Technology object data block



You configure the properties of real objects using the technology objects. The configuration data of the technology objects is stored in a technology object data block (Page 96). The technology object data block contains all configuration data, setpoint and actual values and status information of the technology object. The TIA Portal automatically creates the technology object data block when the technology object is created. You perform read and write access to the data of the technology object data block from your user program.

Motion Control instructions



With the Motion Control instructions you perform the desired functionality in the technology objects. The Motion Control instructions are available in the TIA Portal under "Instructions > Technology > Motion Control". The instructions can be called at all execution levels.





The Motion Control instructions conform to PLCopen (version 2.0).

User program












The Motion Control instructions and the technology object data block represent the programming interfaces for the technology objects. You use Motion Control instructions to transfer Motion Control jobs for the technology objects in your user program. The technology objects process the jobs in the Motion Control organization blocks, which are called independent of the user program, and report back the current status to the Motion Control instruction. Each time the Motion Control instruction is called, the current status of the current job is displayed at the output parameters of the Motion Control instruction. You access status information of the technology object and change specific configuration parameters during runtime using the technology data block.

3.4 Description of technology objects (S7-1500, S7-1500T)

The S7-1500 and S7-1500T CPUs support the following technology objects:

Technology object	Description	S7-1500	S7-1500T
 Speed axis	The speed axis technology object ("TO_SpeedAxis") is used to specify the speed for a drive. The technology object calculates speed setpoints taking into account the preset dynamic settings and outputs them to the drive. You program the motion of the axis with Motion Control instructions. All motions of the speed axis take place as speed-controlled motions.	✓	✓
 Positioning axis	The positioning axis technology object ("TO_PositioningAxis") is used to position a drive with closed-loop position control. The technology object calculates position setpoints taking into account the preset dynamic settings and outputs corresponding speed setpoints to the drive. You issue positioning jobs to the axis with Motion Control instructions in your user program.	✓	✓

3.4 Description of technology objects (S7-1500, S7-1500T)

Technology object	Description	S7-1500	S7-1500T
 Synchronous axis	The synchronous axis technology object ("TO_SynchronousAxis") includes all functions of the positioning axis technology object. The synchronous axis can also be interconnected with a leading value so that it, as the following axis, follows the position change of a leading axis during synchronous operation.	✓	✓
 External encoder	The external encoder technology object ("TO_ExternalEncoder") detects a position and makes this available to the controller. The detected position can be evaluated in the user program.	✓	✓
 Measuring input	The measuring input technology object ("TO_MeasuringInput") is used for quick, accurate and event-dependent detection of the actual position of an axis or external encoder when a signal change occurs at the measuring input.	✓	✓
 Output cam	The output cam technology object ("TO_OutputCam") generates switching signals depending on the position of an axis or external encoder. You can evaluate the switching signals in the user program or feed them to digital outputs.	✓	✓
 Cam track	The cam track technology object ("TO_CamTrack") generates a switching signal sequence depending on the position of an axis or external encoder. A cam track can consist of up to 32 individual output cams. The switch signals are output as a track, which you can evaluate in the user program or connect to digital outputs.	✓	✓
 Cam	The cam technology object ("TO_Cam", "TO_Cam_10k") describes the dependency of an output variable on an input variable in a unit-neutral manner. The cam defines a transfer function $f(x)$ that is used to couple the leading and following axes during camming. The transfer function $f(x)$ is defined using data points and/or segments. Missing function ranges are interpolated. The cam technology object of the type "TO_Cam" can contain up to 1000 points and 50 segments. The cam technology object of the type "TO_Cam_10k" can contain up to 10,000 points and 50 segments.	-	✓
 Leading axis proxy	In the case of a cross-PLC synchronous operation, the leading axis proxy technology object ("TO_LeadingAxisProxy") represents the leading axis for the local synchronous operation within a CPU. The leading axis proxy evaluates the leading value telegram and provides the external leading value for the local following axes.	-	✓
 Kinematics	The kinematics technology object ("TO_Kinematics") is used to interconnect individual axes to a kinematics. The technology object calculates motion setpoints for the tool center point (TCP) of a kinematics taking into account the preset dynamic settings.	-	✓
 Interpreter	The technology object Interpreter ("TO_Interpreter") processes the instructions from Interpreter programs and issues motion commands to technology objects.	-	✓
 Interpreter program	The Interpreter program technology object ("TO_InterpreterProgramm") includes an editor for programming and testing Interpreter programs.. Interpreter programs from the "Interpreter program" technology object can be loaded into the Interpreter.	-	✓
 Interpreter mapping	The technology object Interpreter Mapping ("TO_InterpreterMapping") defines which technology objects and PLC tags an Interpreter program can access.	-	✓

You can find more in-depth information on the individual technology objects and their possible applications in the Motion Control documentation [\(Page 8\)](#).

3.5 HW Configuration (S7-1500, S7-1500T)

Drives and encoders

Drives ensure the motion of the axis. They are integrated in the hardware configuration. When you execute a Motion Control job in your user program, the technology object takes over the control of the drive and the reading in of values of encoders.

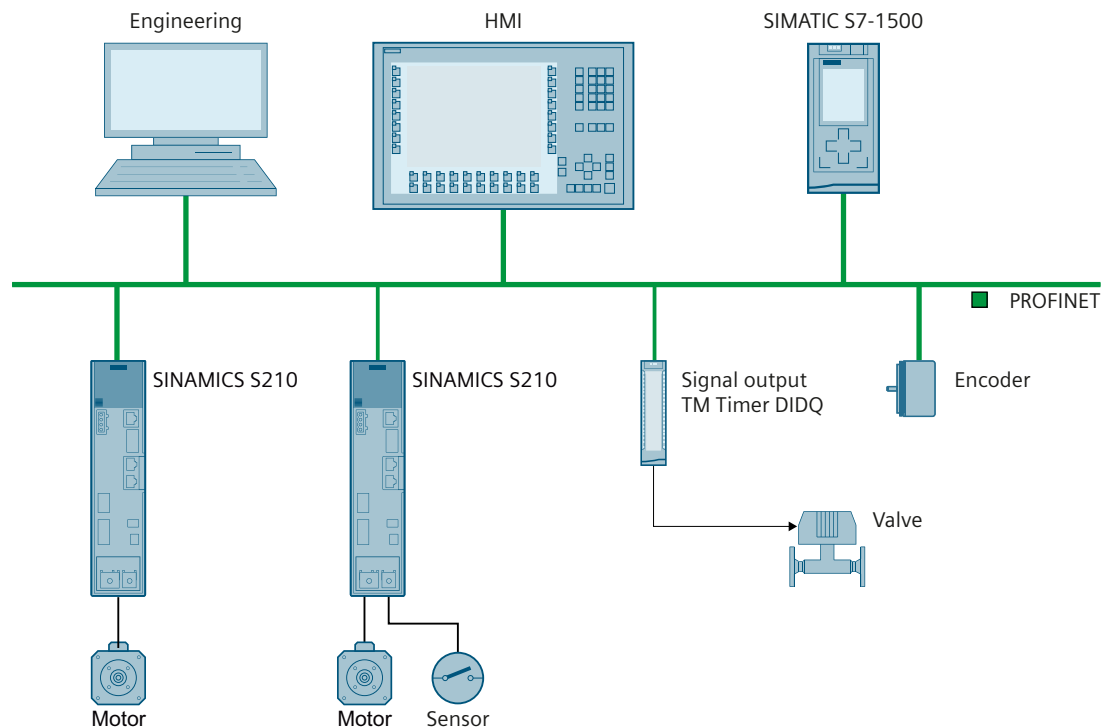
Drives and encoders with PROFIdrive capability are connected by means of PROFIdrive telegrams. The following connections are possible:

- PROFINET IO
- PROFIBUS DP
- Technology module (TM)
- SINAMICS Integrated (SIMATIC Drive Controller)

Drives with analog setpoint interfaces are connected using an analog output (AQ) and an optional enable signal. Analog inputs and outputs are made available by means of corresponding I/O modules.

A drive is also called an actuator, and an encoder is also called a sensor.

The following figure shows an example configuration in which all components are connected to the CPU by means of PROFINET IO:



3.6 Configuration limits (S7-1500, S7-1500T)

The technology objects occupy memory in the CPU's RAM. The internal memory requirements of the technology objects are mapped by the Motion Control and Extended Motion Control resources. Each CPU offers a defined set of these resources. You can find the total resources available in the technical specifications of the utilized CPU.

You can find an overview of the Motion Control and Extended Motion Control resources consumption of a CPU in the TIA Portal under "Tools > Resources".

Motion Control resources

The following table shows how many Motion Control resources a technology object requires on an S7-1500 or S7-1500T CPU.

Technology object	Motion Control resources used
Speed axis	40
Positioning axis	80
Synchronous axis	160
External encoder	80
Measuring input	40
Output cam	20
Cam track	160

Extended Motion Control resources (S7-1500T)

The following table shows how many Extended Motion Control resources a technology object requires on an S7-1500T CPU.

Technology object	Extended Motion Control resources used
Leading axis proxy	3
Cyclic cam of the type "TO_Cam"	2
Cyclic cam of the type "TO_Cam_10k"	20
Kinematics	30
Interpreter	60
Interpreter program	0
Interpreter mapping	0

Application cycle

As the number of technology objects used increases, the computing time needed by CPU to process the technology objects increases. The Motion Control application cycle can be adapted according to the number of technology objects used.

Selecting the S7-1500 with the TIA Selection Tool

To identify the right SIMATIC S7-1500 CPU for your Motion Control application, use the TIA Selection Tool (<https://support.industry.siemens.com/cs/ww/en/view/109767888>).

The TIA Selection Tool takes into account the following criteria when selecting the CPU:

- Used technology, e.g. Kinematics
- Configuration limits of the technology objects
- Cycle time
- Safety functions

New features (S7-1500, S7-1500T)

4.1 New features V8.0 (S7-1500, S7-1500T)

4.1.1 New general functions V8.0 (S7-1500, S7-1500T)

Hardware

The following CPUs with an *03-0AB0 article number are available with a new hardware release:

- 1511C
- 1512C

These CPUs offer the following improved properties with firmware version V3.1:

- The work memory has been increased.
- Motion Control resources have been increased to 1120.
- The Motion Control performance has been increased by at least 25%.

The following CPUs with an *03-0AB0 article number are available with a new hardware release:

- 1513pro(F)
- 1516pro(F)

These CPUs offer the following improved properties with firmware version V3.1:

- The work memory has been increased.
- The Motion Control performance has been increased by at least 25%.

Technology objects

- You can organize technology objects in groups.
- The user interface no longer supports technology objects of Technology versions V1.0 and V2.0. You cannot create new technology objects of these versions. You also cannot change back to these technology objects after upgrading them. Existing technology objects of these versions in the project can continue to be used.

Programming

- You can use organization blocks for Motion Control in software units. For this, the names of the organization blocks must adhere to the naming conventions according to IEC 61131-3. The "-" character is replaced by "_". The new names are independent of the firmware used and the technology version.

Example: The previous name "MC-Servo" has been renamed to "MC_Servo".

- Name values for software units are available. Named values are values that have a unique and understandable name assigned to them. Named values can be referenced in the program and increase the readability and maintainability. The concept of the named value data types is comparable with the concept of the enumerations in accordance with IEC 61131-3.

4.1.2 New axis functions V8.0 (S7-1500, S7-1500T)

Technology version V8.0 contains the following new features:

Axis control panel

- Setting of the absolute encoder
 - The axis control panel provides the new operating modes Absolute Encoder Calibration Absolute and Absolute Encoder Calibration Relative.
- Display and adjustment of velocity override
 - The axis control panel includes 2 new control elements for adjusting the velocity override. A slider, which must be continuously actuated during the axis movements, replaces the control elements for starting and stopping.

Axis control via Interpreter

You can control the speed axis, positioning axis and synchronous axis technology objects via an Interpreter program.

Measuring gearbox for positioning axis and synchronous axis technology objects

- A measuring gearbox is available for encoders of the positioning axis and synchronous axis technology objects. For both technology object types, up to 4 encoders can be configured and are expanded with a measuring gearbox.

Torque precontrol

- For the position controller of the S7-1500 CPU, you can configure torque precontrol.

Dynamic filter with sliding window demand

- For the dynamic filter the new mode "Sliding window demand filter" is available.

Standstill signal on external encoder

- The standstill signal is available with external encoder.

Virtual axis

- The position and velocity setpoints of a virtual axis are directly adopted as actual values with an application cycle delay. The feedback loop and the drive model are not simulated.

Alarm response

- Three stop modes of the drive can be configured for the alarm response "Remove enable".

4.1.3 New measuring input and output cam functions V8.0 (S7-1500, S7-1500T)

Technology version V8.0 contains the following new features:

Cyclic measuring for central measuring input

- Cyclic measuring for central measuring input via telegram 39x is supported.

Monitoring probe

- Using the measuring input type "Measuring via monitoring", the measuring input can capture the measured signal of another configured measuring input.

4.1.4 New synchronous operation functions V8.0 (S7-1500, S7-1500T)

Technology version V8.0 contains the following new features:

Camming (S7-1500T)

- With the Motion Control instruction "MC_GetCamFollowingValueCyclic", you can read the following value that is defined for a leading value from a cyclic cam. In this process, you can specify scaling and offset of the cyclic cam without altering the technology object cam. Job processing is done synchronously so that the result is output directly.
- The Motion Control instruction "MC_GetCamLeadingValue" has been extended with parameters for specifying the offset and scaling of the cam, without altering the cam technology object.
 - MasterOffset
 - SlaveOffset
 - MasterScaling
 - MasterScaling

In addition, you can define an approach direction for the sought leading value with the "ApproachDirection" parameter.

- The Motion Control instruction "MC_GetCamFollowingValue" has been extended with parameters for specifying the offset and scaling of the cam, without altering the cam technology object.
 - MasterOffset
 - SlaveOffset
 - MasterScaling
 - MasterScaling

4.1.5 New kinematics functions V8.0 (S7-1500, S7-1500T)

Technology version V8.0 contains the following new features:

- You can control a kinematics technology object via an Interpreter program.
You can find information on the Interpreter programs in the documentation "S7-1500 Interpreter functions".
- Improved blending behavior during conveyor tracking
Blending is possible for the following motions:
 - From a motion in the tracked OCS to a motion job that desynchronizes the TCP
 - From a motion job that synchronizes the TCP with the tracked OCS to the subsequent motion job in the tracked OCS
 - To blend the intermediate point in the WCS during a movement from one tracked OCS to another tracked OCS, you can also only issue the synchronization job during desynchronization "TrackingState" = 4.
- You can enable dynamic adaptation (for motion jobs) in all phases of conveyor tracking.
- The new tag structure "<TO>.StatusInterpreterMotion <tag name>" contains status information on motion jobs controlled by a technology object Interpreter.
- The array of the tag "<TO>.StatusTool.Frame[1..3].<tag name>" was extended from [1..1] to [1..3]:

S7-1500T Motion Control KinPlus

- The "MC_InverseKinematicsTransformation" Motion Control instruction has been extended to include the "TurnJoint" parameter for the specification of the target joint position ranges.

4.1.6 Interpreter functions V8.0 (S7-1500, S7-1500T)

Technology version V8.0 contains the following Interpreter functions:

Technology objects

The following technology objects are available:

- Interpreter technology object
- Interpreter program technology object
- Interpreter mapping technology object

Programming

MCL

- The programming language Motion Control Language (MCL) is supported. In MCL, you can program sequential motion jobs for kinematics and axes.
- The Interpreter program technology object includes a programming editor for creating and editing MCL programs.

User program

The following Motion Control instructions are available to control the Interpreter technology object in the user program:

- MC_LoadProgram: Load/unload the Interpreter program
- MC_RunProgram: Start execution of the Interpreter program
- MC_StopProgram: Stop execution of the Interpreter program

4.2 New features V7.0 (S7-1500, S7-1500T)

4.2.1 New general functions V7.0 (S7-1500, S7-1500T)

Technology version V7.0 contains the following new features:

Hardware

- The following new CPUs are available:
 - 1514SP/SP F
 - 1514SP T/TF
 - 1508S T/TF
- The following CPUs with an *03-OAB0 article number are available with a new hardware release:
 - 1510SP
 - 1511, 1511F/T/TF
 - 1512SP
 - 1513, 1513F
 - 1515, 1515F/T/TF
 - 1516, 1516F

These CPUs offer the following improved properties with firmware version V3.0:

- The work memory has been increased.
- Motion control resources have been increased:

CPU	Motion Control resources
1510SP 1511, 1511F/T/TF 1512SP 1513, 1513F	1120

- The extended Motion Control resources of the technology CPUs have been increased:

CPU	Extended Motion Control Resources
1511T/TF	90

- The Motion Control performance of the technology CPUs has been increased by at least 25%.
- The CPUs 1516T/TF offer the following improved properties with firmware version V3.0:
 - The code work memory has been expanded to 3 MB. The data work memory has been expanded to 7.5 MB.
 - The CPU performance and the work memory have been increased according to the new hardware release of the CPU 1516 (standard).
- The CPU 1504D TF offers the following improved properties with firmware version V3.0:
 - The code work memory has been expanded to 4 MB. The data work memory has been expanded to 6 MB.
 - Motion Control resources have been increased to 3200.
 - Extended Motion Control resources have been increased to 160.
- The CPU 1507D TF offers the following improved properties with firmware version V3.0:
 - The code work memory has been expanded to 15 MB. The data work memory has been expanded to 40 MB.

Programming

- Temporary references to technology objects in combination with "DB_ANY".
You can assign a tag of data type "DB_ANY" of a type-specific reference to a technology object during runtime. If the tag contains a matching technology object type, then the reference becomes valid. After checking the validity, you can use the reference to access technology object data or Motion Control instructions. A conversion function depending on the technology object type is no longer required.
Parameter transfer for function blocks [\(Page 105\)](#)

4.2.2 New axis functions V7.0 (S7-1500, S7-1500T)

Technology version V7.0 contains the following new features:

Drive and encoder connection

- When connecting a drive via data block, you can configure the communication times T_i , T_o , and T_{DC} for calculating the following error.

Motion control and dynamic limits

- The "MC_MotionInSuperimposed" instruction is available. With this instruction you can assign the axis cyclic applicative motion setpoints for additional distance, velocity and acceleration in addition to the basic motion.
- The "MC_HaltSuperimposed" instruction is available. This instruction decelerates a superimposed motion on the axis generated with the instructions "MC_MoveSuperimposed", "MC_MotionInSuperimposed", or "MC_HaltSuperimposed" to zero velocity.

Traversing range limitation

- Extension SW limit switch:
 - You can additionally configure "Stop at SW limit switch with programmable dynamics".
 - When overtraveling the SW limit switches, emergency stop and retention of the axis enable is possible.
- Extension HW limit switch:
 - When configuring the positioning axis and synchronous axis, the HW limit switches can be configured as "traversable" as before, and now as "non-traversable".
 - When approaching a HW limit switch that cannot be overtraveled, you can configure the alarm response.

Homing

- The "MC_SaveAbsoluteEncoderData" instruction is available. Use this instruction to save the absolute adjustment data for CPU device replacement on the SIMATIC Memory Card.
- With the "MC_Home" instruction, the new incremental encoder adjustment homing type "MC_Home.Mode" = 13 is available. The axis does not perform a compensating motion and the offset of the position is not applied at all encoders of the axis. This makes it easier to readjust individual encoders.
- Active and passive homing of absolute encoders with "MC_Home" is available.
- The homing status of the "<TO>.StatusSensor[1..4].Adjusted" encoders is displayed.
- The S7-1500T CPU supports absolute adjustment/incremental adjustment of non-operational encoders. This means that the respective encoder no longer has to be enabled for homing.

Configuring a control loop

- For axes in a synchronous system, the dynamic filter enables dynamic adaptation of axes with higher dynamic responses to the axis with the lowest dynamic response.

Position monitoring

- The parameter assignment of the additional delay time for following error calculation when the dynamic filter is active is available.

"MC_Power.StopMode" = 3

- The "MC_Power" instruction supports "StopMode" = 3 (coast down).

4.2.3 New synchronous operation functions V7.0 (S7-1500, S7-1500T)

Technology version V7.0 contains the following new features:

Velocity gearing (S7-1500T)

- The "MC_GearInVelocity" instruction is available. With an "MC_GearInVelocity" job, you can start velocity gearing between a leading axis and a following axis.
- Continuous changing of the gear ratio during operation is possible. While that is happening, the following axis remains in the status "Synchronous".

Camming (S7-1500T)

- The "MC_CamIn" instruction has been expanded to include the synchronization profile "SyncProfileReference" = 6. With this setting, the following axis synchronizes in advance over a pre-definable leading value path, starting from the current leading value position. The cam is offset accordingly in the master value range.
- For interpolating the cam technology object, you can improve the system performance using the properties of MC-Interpolator [OB92]. This is possible with the modular S7-1500T(F) CPUs and the SIMATIC Drive Controller as of firmware version V3.0.

4.2.4 New kinematics functions V7.0 (S7-1500, S7-1500T)

Technology version V7.0 contains the following new features:

- For motion preparation of the kinematics technology object, you can improve the system performance using the properties of MC-LookAhead [OB97]. This is possible with the modular S7-1500T(F) CPUs and the SIMATIC Drive Controller as of firmware version V3.0.

NOTE

Upgrade

If you are using user-defined kinematics, disable the system performance improvement after upgrading to technology version V7.0.

Note the information provided in the section "Upgrading to technology version V7.0" ([Page 45](#)).

- You configure mechanical axis couplings in the "Interconnections" configuration window.
- In the new "Joints" configuration window, you configure the joints of the kinematics types with more than four kinematics axes.
- Kinematics control panel:
 - You can now execute the functions of the kinematics control panel with new sliders. You can use the new sliders to finely adjust velocities of motions.
 - 3D visualization provides visual support for commissioning the kinematics technology object.
 - In the JCS, you can jog each kinematics joint individually forward, backward or to a set target position.
 - You can find the dynamics settings for commissioning in the Inspector window under "Properties".
 - The status bits in the "Control panel status" diagnostics window show you whether all kinematics axes are switched on and homed and whether there are errors on the kinematics or the kinematics axes. You can also show additional diagnostics windows via the toolbar.
 - Error messages for commissioning via the kinematics control panel are displayed in a yellow banner above the toolbar.
- Kinematics trace:
 - You can record a maximum of 64 signals in a kinematics trace.
 - You can show or hide the TCP orientation as trace in the kinematics trace.
- The "MC_MoveDirectAbsolute" and "MC_MoveDirectRelative" Motion Control instructions have been extended by the "TurnJoint" parameter.
- The "MC_KinematicsTransformation" and "MC_InverseKinematicsTransformation" Motion Control instructions have been extended by the "AxesCoordSystem" parameter.
- The arrays of the following instruction parameters have been extended from [1..4] to [1..6]:
 - MC_MoveLinearAbsolute.Position
 - MC_MoveLinearRelative.Distance
 - MC_MoveCircularAbsolute.EndPoint
 - MC_MoveCircularRelative.EndPoint
 - MC_MoveDirectAbsolute.Position
 - MC_MoveDirectRelative.Distance
 - MC_KinematicsTransformation.AxesPosition

- MC_KinematicsTransformation.AxesVelocity
- MC_KinematicsTransformation.AxesAcceleration
- MC_InverseKinematicsTransformation.Position
- MC_InverseKinematicsTransformation.Velocity
- MC_InverseKinematicsTransformation.Acceleration

NOTE

Upgrade

When upgrading to technology version V7.0, you must adapt the user program.

Note the information provided in the section "Upgrading to technology version V7.0" ([Page 45](#)).

- The tag structure of the following tags has been extended by the coordinates B and C:
 - <TO>.Tcp
 - <TO>.KcsFrame
 - <TO>.OcsFrame
 - <TO>.Tool
 - <TO>.TcpInWcs
 - <TO>.TcpInOcs
 - <TO>.StatusOcsFrame[1..3]
 - <TO>.FlangeInKcs
 - <TO>.StatusTool
 - <TO>.StatusConveyor[1..3]
 - <TO>.StatusWorkspaceZone[1..10]

S7-1500T Motion Control KinPlus

- You can use the "S7-1500T Motion Control KinPlus" Motion Control package to configure and commission predefined kinematics types with more than four kinematics axes.
- The following kinematics types are available:
 - Cartesian portal 3D with 2 orientations A, B
 - 6-axis articulated arm with central hand
 - Delta picker 3D with 2 orientations A, B
 - User defined kinematics 3D with 3 orientations
- Joints:
 - The joint coordinate system (JCS) is available.
 - The tag "<TO>.Joint" contains the configuration of the joints for kinematics types with more than four interpolating kinematics axes.
 - The tag "<TO>.JointData" contains the setpoints of the kinematics motion for the joints.
- The tag "<TO>.AxisCoupling" contains the configuration of the mechanical axis coupling for kinematics types with more than four interpolating kinematics axes.

4.3 Innovations V6.0 (S7-1500, S7-1500T)

The technology version V6.0 contains the following new features:

General functions

- All diagnostics information is available through the web server.
- You can use the security settings to restrict access to the technology objects.
- A restart of the technology objects kinematics, output cam, cam track and measuring input no longer cancels an active job with "ErrorID" = 16#8001 but with "ErrorID" = 16#800D instead.
- Technology object data connection via data block: As of TIA Portal V17, you can connect the tag structures of the data type "PD_TELx" that are defined in arrays (Array [0..x] of "PD_TELx"), PLC data types or structures within a data block. This function is available for technology objects as of technology version V3.0 and higher.

Axis functions

- A backlash compensation can be configured for the positioning axis and synchronous axis technology objects.
- A linear axis can be configured with linear motor for the positioning axis and synchronous axis technology objects.
- The actual velocity can be calculated from the actual speed NIST_B from the drive telegram for the positioning axis, synchronous axis and external encoder technology objects.
- The "<TO>.PositionMonitoring.MinDwellTime" tag of the positioning axis and synchronous axis technology objects has the start value 0.01 s as default setting.
- Axis control panel: The "Deceleration" element in the "Control" area has the default 100% of the configured value in "Technology objects > Configuration > Extended parameters > Limits > Dynamics limits".

Synchronous operation functions

Gearing (S7-1500T):

- The "MC_PhasingAbsolute" and "MC_PhasingRelative" instructions have been expanded by additional parameters. You can traverse the leading value offset of following axis from a defined or from the current leading value position.
- In gearing, a following value offset is possible via an "MC_OffsetAbsolute" - or "MC_OffsetRelative" job. You can traverse the following value offset of following axis from a defined leading value position or from the current leading value position.
- With a "MC_GearOut" job, you can desynchronize a gearing.

Camming (S7-1500T):

- In contrast to the cam disc technology object of type "TO_Cam", the transfer function $f(x)$ of the new cam technology object of type "TO_Cam_10k" can comprise up to 10,000 interpolation points.
- A diagnostics area is available for the cam technology object via project tree.

- The "MC_CamIn" instruction has been expanded to include the synchronization profile "SyncProfileReference" = 5. With this setting you can rescale the active cam or replace a new cam at the end of the active cam.
- In camming, a leading value offset is possible via a "MC_PhasingAbsolute" or "MC_PhasingRelative" job. You can traverse the leading value offset of following axis from a defined or from the current leading value position.
- In camming, a following value offset is possible via an "MC_OffsetAbsolute" job or "MC_OffsetRelative" job. You can traverse the following value offset of following axis from a defined leading value position or from the current position. .
- With a "MC_CopyCamData" job, you can copy calculated cam elements to a cam. You can also overwrite any existing cam elements in this step. Copying of cam elements is also possible during active camming.
- With a "MC_CamOut" job you can desynchronize a camming.

Cross-PLC synchronous operation (S7-1500T):

- A cross-PLC synchronous operation between technology objects is possible with different technology versions \geq V5.0 on CPUs with different firmware versions \geq V2.8.
- You can provide a cross-PLC leading value over up to eight different transfer areas.

Kinematics functions (S7-1500T)

- In the "Calibration" configuration window can calibrate workspace zones in online and offline mode.
- With a "MC_KinematicsMotionSimulation" job, you can simulate the motion of a kinematics with interconnected axes that are not enabled and exit simulation mode.
- You can configure the maximum smoothing distance in the configuration menu "Technology object > Configuration > Extended parameters> Job sequence" or directly via the "<TO>.Transition.FactorBlendingLength" tag. You can move the kinematics with smoothing distances > 50% of the shorter distance.
- In the "Technology object > Configuration > Extended parameters > Dynamics" configuration menu, you can define the default values for the dynamic responses for synchronous "point-to-point" motions.
- In the kinematics control panel, the dynamic adaptation is effective in "Jog to target position" mode and the dynamic limits of the kinematics axes are taken into account. The "dynamic adaptation without segmentation of the path" is always used.
- The tag "<TO>.StatusWord.InSimulation" indicates the simulation mode of a kinematics.
- The "<TO>.AxesData" tag structure shows the axis data from the kinematics motion.
- The "<TO>.StatusPath.AccumulatedPathLength" tag shows the sum of the distance already traveled.
- The "<TO>.StatusPath.TotalPathLength" tag shows the sum of the total distance (distance already traveled and calculated distance).
- The tag "<TO>.StatusMotionQueue.NumberOfPreparedCommands" shows the number of prepared commands in the job sequence.

4.4 Innovations V5.0 (S7-1500, S7-1500T)

Technology version V5.0 contains the following new features:

General functions

- The values for position, velocity and angle can be configured with higher resolution.
- In the central operation of the technological modules, the clock synchronization is supported via the active backplane bus.
- The organization block MC-PreInterpolator [OB68] enables iposynchronous processing of motion control instructions.
- The default value of the CPU communication load for the following CPUs was reduced from 50% to 20%:
 - S7-1505SP T/TF
 - S7-1511T/TF
 - S7-1515T/TF
 - S7-1516T/TF
- The default values of the bus clocks are:
 - 4 ms for CPUs 1511T/TF and 1515T/TF
 - 2 ms for CPU 1516T/TF
 - 1 ms for CPU 1517T/TF

Axis functions

- In the axis control panel, the dynamic values are retained until the axis control panel is closed.
- With a "MC_Stop" job, you can stop an axis and prevent new jobs.
- With a "MC_Home" job, set positions can be set absolutely or relatively.
- With a "MC_Reset" job, alarms in the drive can be acknowledged without a pending error at the technology object.
- Selected bits in control word 1 and control word 2 can be controlled with a "MC_SetAxisSTW" job.
- With a "MC_WriteParameter" job, hardware limit switches can be enabled and disabled.
- The "<TO>.VelocitySetpoint" tag indicates the active velocity setpoint.
- The tag "<TO>.ModuloCycle" indicates the number of modulo cycles of the setpoint.
- The tag "<TO>.ActualModuloCycle" indicates the number of modulo cycles of the actual value.

Measuring input and output cam functions (S7-1500T)

- The X142 interface of the SIMATIC Drive Controller is supported.

Synchronous operation functions (S7-1500T)

- With a cross-PLC synchronous operation, a synchronous operation is possible between axes which are located on different CPUs within a project.
- A cross-PLC synchronous operation can be configured using the leading axis proxy technology object.
- The actual value extrapolation has been extended.
- With a "MC_LeadingValueAdditive" job, an additive leading value can be specified for the following axis cyclically to a master value.
- With a "MC_GearInPos" job, a subsequent synchronization via leading value distance is possible for gearing.
- With a "MC_CamIn" job, a subsequent synchronization via leading value distance is possible for camming.
- During synchronization, the tag "<TO>.StatusSynchronizedMotion.StatusWord" indicates that the dynamic limits have been exceeded.
- The tag "<TO>.StatusSynchronizedMotion.WaitingFunctionState" indicates a pending synchronous operation.

Kinematics functions (S7-1500T)

- You can configure the predefined kinematics type "SCARA 2D with orientation".
- A model of the kinematics with the configured zones is displayed in the "Diagnostics" window.
- It is possible to calibrate object coordinate systems offline and online using the "Calibration" window.
- The movement preparation of the kinematics technology object is calculated in the MC-LookAhead [OB97] organization block.
- A kinematics with a synchronous "point-to-point" motion can be traversed absolutely with a "MC_MoveDirectAbsolute" job.
- A kinematics with a synchronous "point-to-point" motion can be traversed relatively with a "MC_MoveDirectRelative" job.
- With a "MC_KinematicsTransformation" job, a forward transformation can be calculated.
- With a "MC_InverseKinematicsTransformation" job, an inverse transformation can be calculated.
- With a "MC_TrackConveyorBelt" job, conveyor tracking is possible.

4.5 Innovations V4.0 (S7-1500, S7-1500T)

Technology version V4.0 contains the following new features:

Axis functions

- Exchange of torque data with the drive in the technological units of the technology object.
 - Additive setpoint torque
 - Current actual torque
 - Permissible torque range
- Extension of the data structure of the positioning axis and synchronous axis for using the kinematics technology object
- Use of optimized data blocks (drive/encoder connection)
- Motion specification via "MotionIn" instructions (S7-1500T)

Synchronous operation functions (S7-1500T)

- Direct synchronous setting with "MC_CamIn" V4.0 (S7-1500T)

Kinematics functions (S7-1500T)

- You can use the following kinematics with the kinematics technology object:
 - Cartesian portals
 - Roller picker
 - SCARA
 - Articulated arm
 - Delta picker
 - Cylindrical robots
 - Tripod
 - User-defined kinematics
- You can record the motions of the kinematics in the kinematics trace.
- With the kinematics control panel you can move the kinematics during the commissioning.
- The status of the motion, errors and warnings are displayed in the diagnostics.

4.6 Innovations V3.0 (S7-1500, S7-1500T)

Technology version V3.0 contains the following new features:

General functions

- Technology CPU S7-1500T

Axis functions

- Force/torque limiting
- Fixed stop detection
- Axis type virtual axis
- MC-PreServo [OB67] and MC-PostServo [OB95]
- Using multiple encoders (S7-1500T)

Synchronous operation functions (S7-1500T)

- Cyclic cam technology object (S7-1500T)
- Gearing with "MC_GearInPos" (S7-1500T)
- Camming with "MC_CamIn" (S7-1500T)

Measuring input and output cam functions

- Technology object measuring input
- Output cam technology object
- Cam track technology object

Using versions (S7-1500, S7-1500T)

5.1 Overview of versions (S7-1500, S7-1500T)

With S7-1500 motion control, the version of the technology also defines the version the technology objects and of the motion control instructions. Only one technology version can be operated on a CPU. The overview shown below includes S7-1500 and S7-1500T.

An overview of versions V1.0 and V2.0 of the technology modules is available here [\(Page 48\)](#).

CPU firmware version and technology version

The following table shows which technology version is available as of which firmware version of the CPU.

CPU firmware version ¹⁾	V8.0	V7.0	V6.0	V5.0	V4.0	V3.0
V3.1 ²⁾³⁾	✓	✓	✓	✓	✓	✓
V3.0 ²⁾³⁾	-	✓	✓	✓	✓	✓
V2.9 ²⁾	-	-	✓	✓	✓	✓
V2.8	-	-	-	✓	✓	✓
V2.5, V2.6	-	-	-	-	✓	✓
V2.0, V2.1	-	-	-	-	-	✓

1) The following applies analogously to the CPU firmware of the software controller: V3.0 = V30.0, V3.1 = 30.1, V3.2 = V30.2, ...

2) The CPUs 1518T-4 PN/DP and 1518TF-4 PN/DP only support technology versions as of V6.0.

3) The CPUs with article number *03-0AB0 support technology versions as of V7.0.

Technology objects of a technology version

The following table shows as of which technology version a technology object is available.

Technology object	V8.0	V7.0	V6.0	V5.0	V4.0	V3.0
Speed axis	✓	✓	✓	✓	✓	✓
Positioning axis	✓	✓	✓	✓	✓	✓
External encoder	✓	✓	✓	✓	✓	✓
Synchronous axis	✓	✓	✓	✓	✓	✓
Measuring input	✓	✓	✓	✓	✓	✓
Output cam	✓	✓	✓	✓	✓	✓
Cam track	✓	✓	✓	✓	✓	✓
"TO_Cam" cyclic cam (S7-1500T)	✓	✓	✓	✓	✓	✓
"TO_Cam_10k" cyclic cam (S7-1500T)	✓	✓	✓	-	-	-

Technology object	V8.0	V7.0	V6.0	V5.0	V4.0	V3.0
Leading axis proxy (S7-1500T)	✓	✓	✓	✓	-	-
Kinematics (S7-1500T)	✓	✓	✓	✓	✓	-
Interpreter (S7-1500T)	✓	-	-	-	-	-
Interpreter program (S7-1500T)	✓	-	-	-	-	-
Interpreter mapping (S7-1500T)	✓	-	-	-	-	-

S7-1500 motion control instructions

The following table shows from which technology version a motion control instruction is available.

instructions	V8.0	V7.0	V6.0	V5.0	V4.0	V3.0
MC_Power	✓	✓ ¹⁾	✓	✓	✓	✓
MC_Reset	✓ ¹⁾	✓	✓	✓ ¹⁾	✓	✓
MC_Home	✓	✓ ¹⁾	✓	✓ ¹⁾	✓	✓
MC_Halt	✓	✓	✓	✓	✓ ¹⁾	✓ ¹⁾
MC_HaltSuperimposed	✓	✓	-	-	-	-
MC_MoveAbsolute	✓	✓	✓	✓	✓ ¹⁾	✓ ¹⁾
MC_MoveRelative	✓	✓	✓	✓	✓ ¹⁾	✓
MC_MoveVelocity	✓	✓	✓	✓	✓ ¹⁾	✓
MC_MoveJog	✓	✓	✓	✓	✓ ¹⁾	✓
MC_MoveSuperimposed	✓	✓	✓	✓	✓ ¹⁾	✓
MC_SetSensor	✓	✓	✓	✓	✓	✓
MC_Stop	✓	✓	✓	✓	-	-
MC_SetAxisSTW	✓	✓	✓	✓	-	-
MC_WriteParameter	✓	✓ ¹⁾	✓	✓	-	-
MC_SaveAbsoluteEncoderData	✓	✓	-	-	-	-
MC_MeasuringInput	✓	✓	✓	✓	✓	✓
MC_MeasuringInputCyclic	✓	✓	✓	✓	✓	✓
MC_AbortMeasuringInput	✓	✓	✓	✓	✓	✓
MC_OutputCam	✓	✓	✓	✓	✓	✓
MC_CamTrack	✓	✓	✓	✓	✓	✓
MC_GearIn	✓	✓	✓	✓	✓	✓ ¹⁾
MC_GearInPos	✓	✓	✓	✓ ¹⁾	✓	✓
MC_GearInVelocity	✓	✓	-	-	-	-
MC_PhasingRelative	✓	✓	✓ ¹⁾	✓	✓	✓
MC_PhasingAbsolute	✓	✓	✓ ¹⁾	✓	✓	✓
MC_OffsetRelative	✓	✓	✓	-	-	-

1) Instruction revised in the version

2) Can only be used as of firmware version V2.8.2

5.1 Overview of versions (S7-1500, S7-1500T)

instructions	V8.0	V7.0	V6.0	V5.0	V4.0	V3.0
MC_OffsetAbsolute	✓	✓	✓	-	-	-
MC_CamIn	✓	✓ ¹⁾	✓ ¹⁾	✓ ¹⁾	✓ ¹⁾	✓
MC_SynchronizedMotionSimulation	✓	✓	✓	✓	✓	✓
MC_GearOut	✓	✓	✓	-	-	-
MC_CamOut	✓	✓	✓	-	-	-
MC_LeadingValueAdditive	✓	✓	✓	✓	-	-
MC_InterpolateCam	✓	✓	✓	✓	✓	✓
MC_GetCamLeadingValue	✓ ¹⁾	✓	✓	✓	✓	✓
MC_GetCamFollowingValue	✓ ¹⁾	✓	✓	✓	✓	✓
MC_GetCamFollowingValueCyclic	✓	-	-	-	-	-
MC_CopyCamData	✓	✓	✓	-	-	-
MC_MotionInVelocity	✓	✓	✓	✓	✓	-
MC_MotionInPosition	✓	✓	✓	✓	✓	-
MC_MotionInSuperimposed	✓	✓	-	-	-	-
MC_TorqueAdditive	✓	✓	✓	✓	✓	-
MC_TorqueRange	✓	✓	✓	✓	✓	-
MC_TorqueLimiting	✓	✓	✓	✓	✓	✓
MC_GroupInterrupt	✓	✓	✓	✓	✓	-
MC_GroupContinue	✓	✓	✓	✓	✓	-
MC_GroupStop	✓	✓	✓	✓	✓	-
MC_MoveLinearAbsolute	✓	✓ ¹⁾	✓	✓	✓	-
MC_MoveLinearRelative	✓	✓ ¹⁾	✓	✓	✓	-
MC_MoveCircularAbsolute	✓	✓ ¹⁾	✓	✓	✓	-
MC_MoveCircularRelative	✓	✓ ¹⁾	✓	✓	✓	-
MC_MoveDirectAbsolute	✓	✓ ¹⁾	✓	✓	-	-
MC_MoveDirectRelative	✓	✓ ¹⁾	✓	✓	-	-
MC_TrackConveyorBelt	✓ ¹⁾	✓	✓	✓ ²⁾	-	-
MC_KinematicsMotionSimulation	✓	✓	✓	-	-	-
MC_DefineWorkspaceZone	✓	✓	✓	✓	✓	-
MC_DefineKinematicsZone	✓	✓	✓	✓	✓	-
MC_SetWorkspaceZoneActive	✓	✓	✓	✓	✓	-
MC_SetWorkspaceZoneInactive	✓	✓	✓	✓	✓	-
MC_SetKinematicsZoneActive	✓	✓	✓	✓	✓	-
MC_SetKinematicsZoneInactive	✓	✓	✓	✓	✓	-
MC_DefineTool	✓	✓	✓	✓	✓	-
MC_SetTool	✓	✓	✓	✓	✓	-
MC_SetOcsFrame	✓	✓ ¹⁾	✓	✓	✓	-

¹⁾ Instruction revised in the version

²⁾ Can only be used as of firmware version V2.8.2

instructions	V8.0	V7.0	V6.0	V5.0	V4.0	V3.0
MC_KinematicsTransformation	✓	✓ ¹⁾	✓	✓	-	-
MC_InverseKinematicsTransformation	✓	✓ ¹⁾	✓	✓	-	-
MC_LoadProgram	✓	-	-	-	-	-
MC_RunProgram	✓	-	-	-	-	-
MC_StopProgram	✓	-	-	-	-	-

1) Instruction revised in the version

2) Can only be used as of firmware version V2.8.2

5.2 Replacing devices (S7-1500, S7-1500T)

You can replace an S7-1500 for an S7-1500T of the same design and vice versa. The behavior with respect to functions and the existing configuration is different depending on what is being replaced.

- S7-1500 ⇒ S7-1500T

The functions of the S7-1500 are expanded to include additional parameters for the extended functions of the S7-1500T. The additional parameters are preassigned with default values and must be configured appropriately.

- S7-1500T ⇒ S7-1500

- Extended functions are only supported by an S7-1500T and are no longer available after a replacement with S7-1500.
- Unsupported function blocks are marked.
- Unsupported technology objects are displayed in an error message after compilation and must be deleted.

5.3 Upgrade technology version (S7-1500, S7-1500T)

To use all the advantages of a new technology version, you need to change the technology version of existing projects for a CPU.

There are two ways of changing the technology version of a CPU:

- Changing the version of the Motion Control instructions
- Adding a technology object with an alternative version

The version of a technology object or a Motion Control instruction is indicated in the properties of the technology object in the "General > Information" tab, "Version" field.

The technology objects and Motion Control instructions are only converted to the selected version of the technology during compilation.

Changing the version of the motion control instructions

Requirement: The firmware of the CPU supports the required technology version.

To change the version of the motion control instructions, proceed as follows:

1. Open the program editor (e.g. by opening the OB1).
2. In the "Instructions" task card, select the higher technology version in the "Technology > Motion control" folder.
The invalid technology objects and Motion Control instructions are highlighted in red after the CPU is replaced.
3. Save and compile the project.
The version of the technology objects and Motion Control instructions is changed to the appropriate higher technology version during the project compile.
Pay attention to any error information that is displayed during compilation. Deal with the causes of the errors indicated.
4. Check the configuration of the technology objects.
5. Take into consideration the notes on upgrading to a technology version, including those on the skipped versions.

Result: You have upgraded the technology version of the CPU.

5.4 Upgrading to technology version V8.0 (S7-1500, S7-1500T)

Blending behavior during conveyor tracking

After upgrading to technology version 8.0, blending is possible for the following motions:

- In a motion job that desynchronizes the TCP from the tracked OCS
- From a motion job that synchronizes the TCP with the tracked OCS to the subsequent motion job in the tracked OCS

To maintain compatibility with technology version \leq V7.0, parameterize the motion transition on the following job with "BufferMode" = 1. The current motion job is then not blended.

If you want to blend the current motion job, parameterize the motion transition on the following job with "BufferMode" = 2 or 5 and parameterize the blending distance "TransitionParameter[1]".

Dynamic adaptation during conveyor tracking

Activate the Dynamic adaptation during the conveyor tracking by setting the value for "DynamicAdaption" to "1" or "2" on the motion job. Alternatively, configure the default setting on the technology object by setting the "<TO>.DynamicDefaults.DynamicAdaption" variable to "1" or "2" and specifying the value "-1" on the motion job.

Also configure the dynamic range reserve (<TO>.Conveyor.DynamicReserve).

Newly designed mode of the virtual axis

In technology version V8.0, the behavior of the virtual axis has been changed. The behavior of a virtual axis is no longer identical to the behavior of an axis in simulation. The setpoints are directly adopted as actual values with an application cycle delay. Further configurations do not affect the calculation of the actual values.

To maintain compatibility with virtual axes of technology versions \leq V7.0 for an axis, switch the axis to simulation (<TO>.Simulation.Mode = 1) and deactivate the virtual axis (<TO>.VirtualAxis.Mode = 0).

Organization blocks for Motion Control in software units

When upgrading existing TIA projects, the names of the organization blocks for Motion Control are retained. To use the organization blocks in software units, you must change the names manually according to the IEC 61131-3 naming convention.

5.5 Upgrading to technology version V7.0 (S7-1500, S7-1500T)

Extend default Motion Control instructions for kinematics movements and transformation information

Extend the ARRAY [1..4] OF LREAL to ARRAY [1..6] OF LREAL for target coordinates at the Motion Control instructions for kinematics movements and for the default at the Motion Control instructions for transformation information. The parameters have been extended at the instructions for programming kinematics with more than four kinematics axes. As the same instructions are used for programming kinematics with up to four kinematics axes, you must extend the array. To program kinematics with up to four kinematics axes, parameterize array elements 5 and 6 with "0.0".

The following parameters of the instructions have been extended to six elements:

Instruction	Parameters	Technology version	
		\leq V6.0	V7.0
MC_MoveLinearAbsolute	Position	ARRAY [1..4] OF LREAL	ARRAY [1..6] OF LREAL
MC_MoveLinearRelative	Distance		
MC_MoveCircularAbsolute	EndPoint		
MC_MoveCircularRelative	EndPoint		
MC_MoveDirectAbsolute	Position		
MC_MoveDirectRelative	Distance		
MC_KinematicsTransformation	AxesPosition		

Instruction	Parameters	Technology version	
		≤ V6.0	V7.0
MC_KinematicsTransformation	AxesVelocity	ARRAY [1..4] OF LREAL	ARRAY [1..6] OF LREAL
	AxesAcceleration		
MC_InverseKinematicsTransformation	Position		
	Velocity		
	Acceleration		

Upgrading data types with user-defined transformation

When upgrading to technology version V7.0, the "TO_Struct_TransformationAxisData_V1" and "TO_Struct_TransformationCartesianData_V1" data types are automatically upgraded in the "TransformationParameter" system data block.

Manually update the following data types to V2 at all points of use in the project:

Technology version	
≤ V6.0	V7.0
TO_Struct_TransformationAxisData_V1	TO_Struct_TransformationAxisData_V2
TO_Struct_TransformationCartesianData_V1	TO_Struct_TransformationCartesianData_V2
REF_TO TO_Struct_TransformationParameter_V1	REF_TO TO_Struct_TransformationParameter_V2

Disabling system performance improvement for user-defined kinematics

When the technology version is upgraded to V7.0, system performance improvement for motion preparation of the kinematics technology object is enabled.

If you are using user-defined kinematics, clear the "Improve system performance" check box in the MC-LookAhead [OB97] properties under "General > Multi-core processor".

5.6 Upgrade to technology version V3.0 to V6.0 (S7-1500, S7-1500T)

Upgrading to technology version V6.0

Using "DB_ANY" data type with the cam technology object

On motion control instructions from technology version V6.0, direct assignment of the "DB_ANY" data type at the "Cam" parameter results in an access error. Due to this access error, the CPU changes to the "STOP" state.

Use a data type conversion function for the cam technology object of the type "TO_Cam" or "TO_Cam_10k". A description of the procedure can be found in the section "Parameter transfer for function blocks [\(Page 105\)](#)".

Upgrading to technology version V5.0

Version-dependent organization block "MC-LookAhead [OB97]"

If you switch from technology version V4.0 to \geq V5.0, an "MC-LookAhead [OB97]" organization block (OB) is automatically created.

When switching from technology version V4.0 to \geq V5.0, you still have the same options as with a technology version V4.0.

If you use multiple kinematics, only one "MC-LookAhead [OB97]" OB is used for all kinematics used. In the "MC-LookAhead [OB97]" OB, the motion formatting of the kinematics technology object is calculated.

Upgrading to technology version V4.0

Version-based UDT names

The following table shows the version-based UDT names for the control words and status words of the SIEMENS telegrams 10x:

UDT name < V4.0	UDT name \geq V4.0	WORD data type
PD_STW1	PD_STW1_611Umode	Control word 1 (STW1)
PD_STW2	PD_STW2_611Umode	Control word 2 (STW2)
PD_ZSW1	PD_ZSW1_611Umode	Status word 1 (ZSW1)
PD_ZSW2	PD_ZSW2_611Umode	Status word 2 (ZSW2)

If you switch from technology version < V4.0 to \geq V4.0 or vice versa, an error occurs during the compilation. You have to adapt the UDT names manually.

Upgrading to technology version V3.0

Tags of the technology object

Starting from technology version V3.0, all input and output addresses are specified using data type "VREF". This results in the following changes to the tags of the technical object:

Tag of technology object	Changes starting from V3.0
<TO>.Actor.Interface.AddressIn	Data type: VREF
<TO>.Actor.Interface.AddressOut	Data type: VREF
<TO>.Sensor[1..4].Interface.AddressIn	Data type: VREF
<TO>.Sensor[1..4].Interface.AddressOut	Data type: VREF
<TO>.Actor.Interface.EnableDriveOutputAddress	Data type: VREF
<TO>.Actor.Interface.EnableDriveOutputBitNumber	Tag eliminated
<TO>.Actor.Interface.DriveReadyInputAddress	Data type: VREF
<TO>.Actor.Interface.DriveReadyInputBitNumber	Tag eliminated
<TO>.Sensor[1..4].ActiveHoming.DigitalInputAddress	Data type: VREF
<TO>.Sensor[1..4].ActiveHoming.DigitalInputBitNumber	Tag eliminated
<TO>.Sensor[1..4].PassiveHoming.DigitalInputAddress	Data type: VREF
<TO>.Sensor[1..4].PassiveHoming.DigitalInputBitNumber	Tag eliminated
<TO>.PositionLimits_HW.MinInputAddress	Data type: VREF
<TO>.PositionLimits_HW.MinInputBitNumber	Tag eliminated
<TO>.PositionLimits_HW.MaxInputAddress	Data type: VREF
<TO>.PositionLimits_HW.MaxInputBitNumber	Tag eliminated

5.7 Technology versions V1.0 and V2.0 (S7-1500)

CPU FW and technology versions V1.0 and V2.0

The following table shows which version of the technology is available as of which FW version of CPU.

CPU FW	V2.0	V1.0
V1.6 or higher	✓	-
V1.5	-	✓
V1.1	-	✓
V1.0	-	✓

When changing to a CPU \geq V1.6, you must change the technology version accordingly. Card replacement from a CPU $<$ V1.6 to a \geq CPU V1.6 is supported. In the TIA Portal, you can use a CPU \geq 1.6 only to work on projects with a correspondingly higher technology version.

Technology objects in the technology versions V1.0 and V2.0

The user interface no longer supports technology objects of Technology versions V1.0 and V2.0. You cannot create new technology objects of these versions. You also cannot change back to these technology objects after upgrading them. Existing technology objects of these versions in the project can continue to be used.

The following table shows as of which technology version a technology object is available.

Technology object	V2.0	V1.0
Speed axis	✓	✓
Positioning axis	✓	✓
External encoder	✓	✓
Synchronous axis	✓	-

S7-1500 Motion control instructions in the technology versions V1.0 and V2.0

The following table shows from which technology version a Motion Control instruction is available.

Instructions	V2.0	V1.0
MC_Power	✓	✓
MC_Reset	✓	✓
MC_Home	✓ ¹	✓
MC_Halt	✓	✓
MC_MoveAbsolute	✓	✓
MC_MoveRelative	✓	✓
MC_MoveVelocity	✓	✓
MC_MoveJog	✓	✓
MC_MoveSuperimposed	✓	-
MC_GearIn	✓	-

1) Instruction revised in the version

Innovations V2.0

Technology Version V2.0 contains the following new features:

- Standardization of the "MC_Home.Mode" parameter for S7-1200 Motion Control and S7-1500 Motion Control
- Simulation mode
- Support of safety functions of the drive

Upgrading to Technology version V2.0

Parameter "Mode" of the Motion Control instruction "MC_Home"

The "MC_Home.Mode" parameter for S7-1200 Motion Control and S7-1500 Motion Control has been standardized within the framework of technology version V2.0. This results in a new assignment of the parameter values for the "MC_Home.Mode" parameter.

The table below shows a comparison of the "MC_Home.Mode" parameter for technology V1.0 and \geq V2.0:

MC_Home.HomingMode V1.0	Parameter value	MC_Home.Mode \geq V2.0
Direct homing (absolute)	0	Direct homing (absolute)
Direct homing (relative)	1	Direct homing (relative)
Passive homing	2	Passive homing (without reset)
Passive homing (with configured home position)	3	Active homing
Active homing	4	Reserved
Active homing (with configured home position)	5	Active homing (with configured home position)
Absolute encoder adjustment (relative)	6	Absolute encoder adjustment (relative)
Absolute encoder adjustment (absolute)	7	Absolute encoder adjustment (absolute)
Passive homing (without reset)	8	Passive homing
Abort passive homing	9	Abort passive homing
-	10	Passive homing (with configured home position)

You can find additional information about the "MC_Home.Mode" parameter in the description of the Motion Control instruction "MC_Home".

Resetting the "Mode" parameter of the Motion Control instruction "MC_Home"

When you upgrade the technology version from V1.0 to \geq V2.0, the "MC_Home.HomingMode" parameter (V1.0) is then renamed to "MC_Home.Mode" (\geq V2.0). The assignment of the parameter values is changed as well.

To reset the "MC_Home.Mode" parameter (V2.0), follow these steps:

- To change the technology version, follow the instructions given above.

When compiling the project, the "MC_Home.HomingMode" parameter (V1.0) is renamed to "MC_Home.Mode" (\geq V2.0):

 - The assignment of the parameter values is changed.

You can find additional information about the "MC_Home.Mode" parameter in the description of the Motion Control instruction "MC_Home".
 - The value configured at the "MC_Home.HomingMode" parameter (V1.0) is lost. As a note on renaming, the following text is entered as the parameter value in the "MC_Home.Mode" parameter (\geq V2.0).

"The interface has changed. You can find additional information in the description of the Motion Control instruction MC_Home".
 - There is a message in the "Info > Compile" tab in the Inspector window stating that the operand has the wrong data type.
- Change the value of the "MC_Home.Mode" parameter (\geq V2.0) in your user program according to the new assignment.
- Save and compile the project.

Configuring (S7-1500, S7-1500T)

6.1 Basics on the configuration of technology objects (S7-1500, S7-1500T)

6.1.1 Add technology object (S7-1500, S7-1500T)

The following describes how to add a technology object in the project tree.

Requirements

- A project with a CPU S7-1500/S7-1500T is created.
- The CPU supports the required technology object.
- For a lower-level technology object, the higher-level technology object must have been created.

Procedure

To add a technology object, follow these steps:

1. Open the CPU's folder in the project navigator.
2. Open the "Technology Objects" folder.
3. Double-click "Add new object".
The "Add new object" dialog opens.
4. Select the required technology object. You can infer the function of the technology object from the displayed description.
5. If you add output cams, cam tracks, measuring inputs to a technology object, select the higher-level technology object in the "Axis or external encoder that is to be assigned".
6. In the "Name" input field, adapt the name to your requirements.
7. To change the suggested data block number, select the "Manual" option.
8. To add your own information about the technology object, click "Additional information".
9. To open the configuration after adding the technology object, select the "Add new and open" check box.
10. When adding a "Interpreter" technology object, you can select a "Select kinematics" technology object from the drop-down list under "Kinematics (optional)". Upon addition, the Interpreter technology object is assigned to the selected "Kinematics" technology object.
11. To add the technology object, click "OK".

Result

The new technology object was created and created in the "Technology objects" folder in the project tree.

If the "MC_Servo" and "MC_Interpolator" organization blocks were not yet present, they were added.

When adding the "Kinematics" or "Interpreter" technology objects, the "MC_LookAhead" organization block is also added.

6.1.2 Copy technology object (S7-1500, S7-1500T)

You can copy a technology object in the following ways:

- Copying a technology object within a CPU
- Copying a technology object from a CPU S7-1500 to a CPU S7-1500T
Additional parameters for the extended functions are preset with default values. You must configure them appropriately.
- Copying a technology object from a CPU S7-1500T to a CPU S7-1500
Additional parameters, which are not supported by the CPU S7-1500, are reset to the default values.
- Copying a Kinematics technology object with an assigned Interpreter technology object
 - When you copy a Kinematics technology object with an assigned Interpreter technology object, the link to the Interpreter technology object is also copied. Note that an interpreter technology object can be assigned to only one kinematics technology object. In one of the Kinematics technology objects, remove the assignment to the Interpreter technology object.
 - When you copy a Kinematics technology object with an assigned Interpreter technology object to a different S7-1500T CPU, the link to the Interpreter technology object is also copied. If there is a technology object with the same name, Interpreter, in the other CPU, the link points to it. Otherwise, the link is without a target.

When you copy a technology object that has lower-level technology objects such as output cams, cam tracks or measuring inputs, the lower-level technology object are also copied.

The following describes how to copy a technology object within a CPU. The procedure also applies accordingly to the other copying methods mentioned.

Requirement

- You have created an technology object in the project.

Procedure

To copy a technology object, follow these steps:

1. Open the CPU's folder in the project navigator.
2. Open the "Technology Objects" folder.
3. If necessary, open the higher-level technology object.
4. Select the technology object to be copied.
5. To also copy the technology objects connected to a technology object, such as axes on a Kinematics technology object, select them as well. To select multiple axes, press and hold down the <Ctrl> key.
6. Select "Copy" in the shortcut menu.
7. Select the "Technology objects" folder or the higher-level technology object.
8. Select "Paste" in the shortcut menu.

Result

The selected technology object, including lower-level technology objects, has been copied and created in the "Technology objects" folder of the project tree.

6.1.3 Delete technology object (S7-1500, S7-1500T)

You can delete technology objects in the project tree.

When you delete a technology object that has lower-level technology objects such as output cams, cam tracks or measuring inputs, the lower-level technology object are also deleted.

Requirement

- You have created an technology object in the project.

Procedure

To delete a technology object, follow these steps:

1. Open the CPU's folder in the project navigator.
2. Open the "Technology objects" folder.
3. If necessary, open the higher-level technology object.
4. Select the technology object to be deleted.
5. Select the "Delete" command in the shortcut menu.
The "Confirm delete" dialog is opened.
6. To delete the technology object, click "Yes".

Result

The selected technology object has been deleted.

Lower-level technology objects of the deleted technology object are deleted.

Technology objects connected with the technology object remain intact after deletion, for example, linked axes to the Kinematics technology object.

Links to a technology object remain as links without a target after deletion of the technology object.

6.1.4 Group technology objects (S7-1500, S7-1500T)

You can create subfolders (groups) within the "Technology objects" folder. You use these subfolders to structure your technology objects. Only the technology objects in the subfolders are relevant to the process, not the folders themselves. Groups of technology objects behave in the same way as groups of program blocks.

When copying or deleting a group, all technology objects contained in it are copied or deleted.

The online/offline comparison only shows differences at the technology object, but not that the technology objects are in different groups.

6.1.5 Working with the configuration editor (S7-1500, S7-1500T)




You configure the properties of a technology object in the configuration window. To open the configuration window of the technology object in the project view, follow these steps:

1. Open the device "Technology objects" group in the project navigator.
2. Select the technology object and double-click on "Configuration".

The configuration is divided into categories which depend on the object type, for example, basic parameters, hardware interface, extended parameters.





Configuration editor icons

Icons in the area navigation of the configuration show additional details about the status of the configuration:


Symbol	Description
	The configuration contains default values and is complete. The configuration contains only default values. With these default values you can use the technology object without additional changes.
	The configuration contains user-defined or automatically adapted values and is complete. All input fields of the configuration contain valid values and at least one preset value has been changed.
	The configuration is incomplete or incorrect. At least one input field or drop-down list contains an invalid value. The corresponding field or the drop-down list is displayed on a red background. Click the field shows you the roll-out error message that indicates the cause of error.

6.1.6 Toolbar of the configuration (S7-1500, S7-1500T)

The following functions are available in the toolbar of the function view:

Symbol	Function	Explanation
	Show online values	Displays the current values read back from the CPU.
	Couples the function view and parameter view for the objects selected in the navigation	Enables the targeted toggling between the parameter view and function-based view.
	Collapse/expand all nodes and objects	Collapses or expands all nodes and objects of the navigation or the data structure in the currently active view.
	Collapse/expand the nodes below the marked node	Collapses or expands the marked nodes and objects of the navigation or the data structure in the currently active view.

6.2 Compare values (S7-1500, S7-1500T)

If an online connection to the CPU is available, the "Monitor all" function  appears in the configuration of the technology object.

The "Monitor all" function provides the following options:

- Comparison of configured start values of the project with the start values in the CPU and the actual values
- Direct editing of actual values and the start values of the project
- Immediate detection and display of input errors with suggested corrections
- Backup of actual values in the project
- Transfer of start values of the project to the CPU as actual values




NOTE




Differences between online and offline values

By adding or deleting technology objects that have a connection to other technology objects, such as cams, cam tracks, measuring inputs or synchronous axes, differences can occur when online and offline values are compared. These differences can be eliminated by re-compiling the project and then uploading it to the CPU.

Icons and operator controls

If there is an online connection to the CPU, the actual values are displayed at the parameters. In addition to the actual values of the parameters, the following symbols appear:

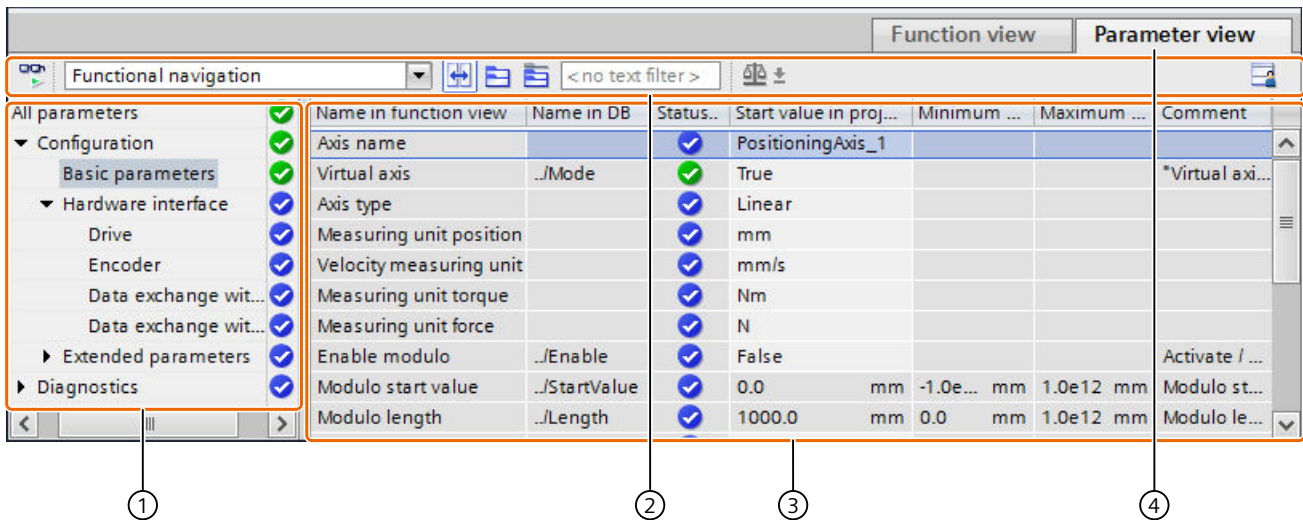
Symbol	Description
	Start value in CPU matches the configured start value in project
	Start value in CPU does not match the configured start value in project
	Software difference in lower-level component: The online and offline versions differ in at least one lower-level software component.

Symbol	Description
	The comparison of the Start value in CPU with the configured start value in project cannot be performed.
	Comparison of the online and offline values is not advisable.
	Use this button to show the start value of the CPU and the start value of the project for the respective parameter.

You can change the start value of the CPU directly and then download it to the CPU. With directly changeable parameters, the actual value can also be changed and the change is transferred immediately to the CPU.

6.3 Working with the parameter view (S7-1500, S7-1500T)

The Parameter view provides you with a general overview of all relevant parameters of a technology object. You obtain an overview of the parameter settings and can easily change them in offline and online mode.



- ① Navigation (Page 58)
- ② Toolbar (Page 57)
- ③ Parameter table (Page 58)
- ④ "Parameter view" tab

Function scope

The following functions are available for analyzing the parameters of the technology objects and for enabling targeted monitoring and modification.

Display functions:

- Display of parameter values in offline and online mode
- Display of status information of the parameters
- Display of value deviations and option for direct correction
- Display of configuration errors
- Display of value changes as a result of parameter dependencies

- Display of all memory values of a parameter: Start value PLC, Start value in project, Monitor value
- Display of the parameter comparison of the memory values of a parameter


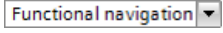






Operator control functions:

- Navigation for quickly changing between the parameters and parameter structures
- Text filter for faster searches for particular parameters
- Sorting function for customizing the order of parameters and parameter groups to requirements
- Memory function for backing up structural settings of the Parameter view
- Monitoring and modifying of parameter values online
- Saving a snapshot of parameter values of the CPU in order to capture momentary situations and to respond to them
- Applying a snapshot of parameter values as start values
- Download of modified start values to the CPU
- Comparison functions for comparing parameter values with one another

6.3.1 Structure of the parameter view (S7-1500, S7-1500T)

6.3.1.1 Toolbar (S7-1500, S7-1500T)

The following functions can be selected in the toolbar of the parameter view.

Symbol	Function	Explanation
	Monitor all	Starts monitoring of the visible tags in the active table.
	Select navigation structure	Toggle between function-based navigation and the view of the data structure of the technology data block.
	Couples the function view and parameter view for the objects selected in the navigation	Enables the targeted toggling between the parameter view and function-based view.
	Collapse/expand all nodes and objects	Collapses or expands all nodes and objects of the navigation or the data structure in the currently active view.
	Collapse/expand the nodes below the marked nodes	Collapses or expands the marked nodes and objects of the navigation or the data structure in the currently active view.
	Text filter...	After entry of a character string: Display of all parameters containing the entered string in one of the currently visible columns.
	Selection of compare values	Selection of parameter values that are to be compared with one another in online mode (Start value in project, Start value PLC) Only in online mode.
	Save window settings	Saves your display settings for the Parameter view (e.g. selected navigation structure, activated table columns, etc.)

6.3.1.2 Navigation (S7-1500, S7-1500T)

Within the "Parameter view" tab, the following alternative navigation structures can be selected.

Navigation		Explanation
Functional navigation	<ul style="list-style-type: none"> ▼ All parameters <ul style="list-style-type: none"> ▼ Configuration <ul style="list-style-type: none"> Basic parameters ▼ Hardware interface <ul style="list-style-type: none"> Drive Encoder 	In the functional navigation, the structure of the parameters is based on the structure in the configuration window ("Function view" tab), commissioning window and diagnostics window.
Data structure	<ul style="list-style-type: none"> Velocity Acceleration ActualSpeed VelocitySetpoint ▼ Actor <ul style="list-style-type: none"> Type 	In the "Data structure" navigation, the structure of the parameters is based on the structure of the technology data block.

You can use the "Select navigation structure" drop-down list to toggle the navigation structure.

6.3.1.3 Parameter table (S7-1500, S7-1500T)

The table below shows the meaning of the individual columns of the parameter table. You can show or hide the columns as required.

Column	Explanation	Offline	Online
Name in function view	Name of the parameter in the function view. The display field is empty for parameters that are not configured via the technology object.	✓	✓
Name in DB	Name of the parameter in the technology data block. If the parameter is part of a structure or UDT, the prefix "../" is added. The display field is empty for parameters that are not contained in the technology data block.	✓	✓
Full name in DB	Complete path of the parameter in the instance data block. The display field is empty for parameters that are not contained in the technology data block.	✓	✓
Status of configuration	Display of the completeness of the configuration using status symbols.	✓	-
Compare result	Result of the "Compare values" function. This column is displayed when there is an online connection.	-	✓
Start value in project	Configured start value in project. Error indication if entered values have a syntax or process-related error.	✓	✓
Default value	Value that is pre-assigned to the parameter. The display field is empty for parameters that are not contained in the technology data block.	✓	✓

✓ This function is visible in offline/online mode.

- This function is not visible in offline/online mode.

Column	Explanation	Offline	Online
Start value PLC	Start value in the CPU. This column is displayed when there is an online connection.	-	✓
Monitor value	Current value in the CPU. This column is displayed when there is an online connection.	-	✓
Modify value	Value that is to be used to change the monitor value. This column is displayed when there is an online connection.	-	✓
Minimum value	Minimum process-related value of the parameter. If the minimum value is dependent on other parameters, it is defined: <ul style="list-style-type: none"> Offline: by the start value in the project. Online: by the monitor values. 	✓	✓
Maximum value	Maximum process-related value of the parameter. If the maximum value is dependent on other parameters, it is defined: <ul style="list-style-type: none"> Offline: by the start value in the project. Online: by the monitor values. 	✓	✓
Setpoint	Designates the parameter as a setpoint. These parameters can be initialized online.	✓	✓
Data type	Data type of the parameter. The display field is empty for parameters that are not contained in the technology data block.	✓	✓
Retain	Designates the value as a retentive value. The values of retentive parameters are retained even after the voltage supply is switched off.	✓	✓
Accessible from HMI	Indicates whether the HMI can access this parameter during runtime.	✓	✓
Visible in HMI	Indicates whether the parameter is visible in the selection list of the HMI by default.	✓	✓
Comment	Brief description of the parameter.	✓	✓

✓ This function is visible in offline/online mode.

- This function is not visible in offline/online mode.

6.3.2 Opening the parameter view (S7-1500, S7-1500T)

Requirement

The technology object was added in the project navigator.

Procedure

1. Open the "Technology objects" folder in the project tree.
2. Open the technology object in the project tree.
3. Double-click the "Configuration" object.
4. Select the "Parameter view" tab in the top right corner.

Result

The Parameter view opens. Each displayed parameter is represented by one row in the parameter table.

The displayable parameter properties (table columns) vary depending on whether you are working with the Parameter view in offline or online mode.

In addition, you can selectively display and hide individual table columns.

6.3.3 Working with the parameter view (S7-1500, S7-1500T)

The following table provides an overview of the functions of the Parameter view in online and offline mode described below:

Function/action	Offline	Online
Filtering the parameter table (Page 60)	✓	✓
Sorting the parameter table (Page 61)	✓	✓
Indicating errors (Page 62)	✓	✓
Editing start values in the project (Page 62)	✓	✓
Monitoring values online in the parameter view (Page 64)	-	✓
Modifying values (Page 64)	-	✓
Comparing values (Page 65)	-	✓

✓ This function is possible in offline operation/online operation.

- This function is not possible in offline/online mode.

6.3.3.1 Filtering the parameter table (S7-1500, S7-1500T)

You can filter the parameters in the parameter table in the following ways:

- With the text filter
- With the subgroups of the navigation

Both filter methods can be used simultaneously.

With the text filter

The text filter can only be use on texts in displayed parameter lines and displayed columns.

To work with the text filter, follow these steps:

1. To filter a desired character string, enter it in the "Text filter..." input box.

The parameter table displays only the parameters containing the character string.

To reset text filtering, the following options are available:

- Select another parameter group in the navigation.
- Switch between data navigation and functional navigation, or vice versa.

With the subgroups of the navigation

To filter the navigation with the subgroups, follow these steps:

1. Click the desired parameter group in the navigation, e.g., "Static".
The parameter table now only shows the "static" parameters. You can select further subgroups for some groups of the navigation.
2. If all parameters are to be shown again, click "All parameters" in the navigation.

6.3.3.2 Sorting the parameter table (S7-1500, S7-1500T)

The values of the parameters are arranged in rows. The parameter table can be sorted by any displayed column.

- In columns containing numerical values, sorting is based on the magnitude of the numerical value.
- In text columns, sorting is alphabetical.

Sort column-by-column

1. Position the cursor in the header cell of the desired column.
The background of this cell turns blue.
2. Click the column header.

Result

The entire parameter table is sorted by the selected column. A triangle with tip facing up appears in the column header.

Clicking the column header again changes the sorting as follows:

Symbol	Description
▲	Parameter table is sorted in ascending order.
▼	Parameter table is sorted in descending order.
No symbol	The sorting is removed again. The parameter table shows the default display.

The "../" prefix in the "Name in DB" column is ignored when sorting.

6.3.3.3 Indicating errors (S7-1500, S7-1500T)

Error display

Parameter assignment errors that result in compilation errors (e.g. limit violation) are indicated in the Parameter view.

Every time a value is entered in the Parameter view, a check is made for process-related and syntax errors and displayed with the following indicators:

- Red error symbol in the "Status of configuration" (offline mode) or "Compare result" (online mode, depending on the selected comparison type) columns
- Table field with red background

If you click the bad field, a roll-out error message appears with information of the permissible value range or the required syntax (format)

Compilation error

From the error message of the compiler, you can directly open the Parameter view (functional navigation) containing the parameter causing the error in situations where the parameter is not displayed in the configuration window.

6.3.3.4 Editing start values in the project (S7-1500, S7-1500T)

With the Parameter view, you can edit the start values in the project in offline mode and online mode.

- You make value changes in the "Project start value" column of the parameter table.
- In the "Status of configuration" column of the parameter table, the progress of the configuration is indicated by the familiar status symbols from the configuration window of the technology object.

Boundary conditions

- If other parameters depend on the parameter whose start value was changed, the start value of the dependent parameters are also adapted.
- If a parameter of a technology object is not editable, it is also not editable in the parameter view. The ability to edit a parameter can also depend on the values of other parameters.

Defining new start values

To define start values for parameters in the Parameter view, follow these steps:

1. Open the Parameter view of the technology object.
2. Enter the desired start values in the "Project start value" column. The value must match the data type of the parameter and must not exceed the value range of the parameter. The limits of the value range can be seen in the "Maximum value" and "Minimum value" columns.

The "Status of configuration" column indicates the progress of the configuration with colored symbols.

Following adaptation of the start values and downloading of the technology object to the CPU, the parameters take the defined value at startup if they are not declared as retentive ("Retain" column).

Error display

When a start value is input, a check is made for process-related and syntax errors and the result is indicated.

Bad start values are indicated by:

- Red error symbol in the "Status of configuration" (offline mode) or "Compare result" (online mode, depending on the selected comparison type) columns and/or
- Red background in the "Project start value" field
If you click on the faulty field: a roll-out error message appears with information of the permissible value range or the necessary syntax (format)

Correcting bad start values

1. Correct bad start values using information from the roll-out error message. Red error symbol, red field background, and roll-out error message are no longer displayed.

The project cannot be successfully compiled unless the start values are error-free.

6.3.3.5 Monitoring values online in the parameter view (S7-1500, S7-1500T)

You can monitor the values currently taken by the parameters of the technology object in the CPU (Actual values) directly in the Parameter view.

Requirements

- There is an online connection.
- The technology object is downloaded to the CPU.
- The Parameter view of the technology object is open.

Procedure



As soon as the Parameter view is online, the following columns are additionally displayed:

- Compare result
- PLC start value
- Actual value

The "Actual value" column shows the current parameter values on the CPU.

Display

All columns that are only available online are displayed in color as follows:


Color	Description
	The values are modifiable.
	These values cannot be changed.

See also

[Parameter table \(Page 58\)](#)

6.3.3.6 Modifying values (S7-1500, S7-1500T)

With the Parameter view, you can modify values of the technology object in the CPU. You can assign values to a parameter once (control value) and modify them immediately. The modify request is executed as quickly as possible without reference to any particular point in the user program.

<p> DANGER</p> <p>Danger when modifying Changing the parameter values while the plant is operating may result in severe damage to property and personal injury in the event of malfunctions or program errors. Make sure that dangerous states cannot occur before you use the "Modify" function.</p>
--

Requirements

- There is an online connection.
- The technology object is downloaded to the CPU.
- The Parameter view of the technology object is open.
- The parameter is modifiable. (Associated field in the "Actual value" column has a corresponding background color).

Procedure

To modify values immediately, follow these steps:

1. Enter the desired control value in the "Actual value" column of the parameter table.
The parameter is modified once and immediately with the specified value.

Error display

When a control value is input, a check is made immediately for process-related and syntax errors and the result is indicated.

You can recognize incorrect control values as follows:

- The field "Actual value" is displayed with red background color.
- If you click in the incorrect field, a roll-out error message appears with information on the permissible value range or the necessary syntax.

Bad control values

- Control values with process-related errors can be transmitted.
- Control values with syntax errors cannot be transmitted.

6.3.3.7 Comparing values (S7-1500, S7-1500T)

You can use comparison functions to compare the following memory values of a parameter:


- Start value in project
- Start value PLC
- Actual value in PLC

Requirements





- There is an online connection.
- The technology object is downloaded to the CPU.
- The Parameter view of the technology object is open.

Procedure

To compare the start values on the various target systems, follow these steps:

1. In the icon  , click on the arrow "Select comparison values".
A selection list containing the comparison options opens:
 - Start value in project / Start value in PLC (default setting)
 - Start value in project / Actual value in PLC
 - Start value in PLC / Actual value in PLC
2. Select the desired comparison option.
The selected comparison option is carried out immediately:
 - Symbols are used in the "Compare result" column to indicate the result of the comparison of the selected columns.

Symbol in "Compare result" column

Symbol	Meaning
	The compare values are equal and error-free.
	The compare values are not equal and error-free.
	At least one of the two compare values has a process-related or syntax error.
	The comparison cannot be performed. At least one of the two comparison values is not available (e.g. snapshot).

Symbol in the navigation

The symbols are shown in the same way in the navigation if the compare result applies to at least one of the parameters below the displayed navigation structure.

6.4 Configuring technological modules and onboard I/O for Motion Control (S7-1500, S7-1500T)

Use with Motion Control

You can use technology functions of technology modules and the onboard I/Os of the CPUs listed below with Motion Control technology objects. To use the technology functions for Motion Control, specific parameters of the I/O configuration and parameters of the configuration of the technology object must be configured accordingly. Which parameters are relevant for the function is described below. You can set additional parameters that are not listed here. The description of the parameters is found in the documentation of the respective technology module or the respective CPU.

The following technology modules support Motion Control functionalities:

S7-1500/ET 200MP	ET 200 SP	Possible applications
TM Count 2x24V (Page 68) ¹⁾	TM Count 1x24V (Page 68) ¹⁾	<ul style="list-style-type: none"> Position detection for homing with zero mark via PROFIdrive telegram Position detection with measuring input
TM PosInput 2 (Page 70) ¹⁾	TM PosInput 1 (Page 70) ¹⁾	<ul style="list-style-type: none"> Position detection for homing with zero mark via PROFIdrive telegram Position detection with measuring input
TM Timer DIDQ 16x24V (Page 72) ²⁾	TM Timer DIDQ 10x24V (Page 72) ²⁾	<ul style="list-style-type: none"> Output of output cam and cam track for time-controlled switching ²⁾ Time-based position detection via measuring input (time stamp recording) ²⁾
–	TM Pulse 2x24V (Page 74) ¹⁾	<ul style="list-style-type: none"> Drive connection using PWM (pulse width modulation)
TM PTO 4 (Page 77) ³⁾	TM PTO 2x24V (Page 75) ³⁾	<ul style="list-style-type: none"> Drive connection via PTO (Pulse Train Output) Position detection with measuring input via PROFIdrive telegram ²⁾ Position detection for homing with zero mark via PROFIdrive telegram

¹⁾ Automatic data exchange for encoder values is supported

²⁾ Isochronous mode required

³⁾ Automatic data exchange for drive and encoder values is supported

The following CPUs support Motion Control functionalities through their onboard I/O:

CPU	Possible applications
CPU 1511C-1 PN/CPU 1512C-1 PN (Page 79) ¹⁾	<ul style="list-style-type: none"> Drive connection via PTO (Pulse Train Output) Drive connection using PWM (pulse width modulation) Encoder connection via HSC (High-speed counter) Position detection with measuring input via PROFIdrive telegram ²⁾
SIMATIC Drive Controller (Page 85)	<ul style="list-style-type: none"> Output of output cam and cam track for time-controlled switching ²⁾ Time-based position detection via measuring input (time stamp recording) ²⁾

¹⁾ Automatic data exchange for drive and encoder values is supported

²⁾ Isochronous mode required

Isochronous mode

Isochronous mode is required for use with a measuring input, output cam or cam track.

Technology modules can be used centrally or distributed in the system. Isochronous mode is supported in central operation as well as distributed operation with suitable PROFINET interface modules.

The onboard technology I/Os (X142) of the SIMATIC Drive Controller support clock synchronization.

Automatic data exchange for drive and/or encoder values

By selecting the check box for automatic data exchange, the drive and encoder parameters are automatically applied in the CPU.





Alternatively, you can manually match, by drive and encoder type, the parameters described and identified in the following table.

The following types are available for automatic data exchange:

- **Offline**
Select the check box if you want to transfer the offline values of the drive or encoder to the configuration of the technology object in the project.
- **Online**
Select the check box if you want to transfer the effective values online in the drive or encoder to the CPU during runtime. The drive and encoder parameters are transferred from the bus after the (re)initialization of the technology object and (re)start of the drive, encoder or the CPU.

6.4.1 TM Count 1x24V/TM Count 2x24V (S7-1500, S7-1500T)


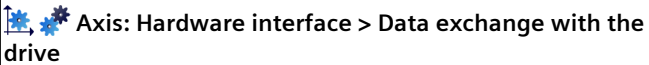
For use with Motion Control, the following parameters must be configured:

Configuration	
Technology module TM Count 1x24V/TM Count 2x24V	Technology object  Axis and  external encoder
TM Count 1x24V/TM Count 2x24V > Channel 0/1 > Operating mode	–
Select "Position detection for Motion Control technology object" operating mode	
TM Count 1x24V/TM Count 2x24V > Channel 0/1 > Module parameters	Hardware interface > Encoder
–	Select "Encoder" data connection and the channel configured for Motion Control on the technology module as encoder
Signal type • Incremental encoder	Select encoder type corresponding to configuration for technology module • Incremental ¹⁾
–	 Axis: Hardware interface > Data exchange with encoder  External encoder: Hardware interface > Data exchange Telegram "Standard telegram 83" is automatically selected after the selection of the encoder. Clear the check box "Automatic data exchange for encoder values (online)"

¹⁾ Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected

–" No configuration for technology module/technology object is required for these parameters


6.4 Configuring technological modules and onboard I/O for Motion Control (S7-1500, S7-1500T)

Configuration	
Technology module TM Count 1x24V/TM Count 2x24V	Technology object  Axis and external encoder
–	Select the check box "Automatic data exchange for encoder values (offline)" If the check box is cleared, you can manually match the parameters described and identified in this table. Select rotary or linear measuring system type ¹⁾
Signal evaluation • Single • Double • Quadruple	Select fine resolution corresponding to configuration on the technology module ¹⁾ • 0 = Single • 1 = Double • 2 = Quadruple
• Rotary type: Enter increments per revolution • Linear type: Configuration not relevant	• Rotary type: Enter increments per revolution corresponding to configuration at technology module (1:1) ¹⁾ • Linear type: Enter distance between increments ¹⁾
–	 Axis: Hardware interface > Data exchange with the drive
• Rotary type: Enter reference speed corresponding to configuration for technology object (1:1) • Linear type: Configuration not relevant	Enter reference speed
–	Homing
Select the homing signal for homing mark 0: • Signal N of the incremental encoder • DIO	Use the homing mode "Use zero mark via PROFIdrive telegram". TM Count 1x24V as of V2.0 The states of the digital inputs are displayed in the operating mode "Position detection for Motion Control technology object" in the process image. The following bits are used for this purpose: • STS_DIO (Status of DIO): ZSW2_ENC.Reserved_Bit11 • STS_DI1 (Status of DI1): ZSW2_ENC.Reserved_Bit10 • STS_DI2 (Status of DI2): ZSW2_ENC.Reserved_Bit8 To select one of the digital inputs use, for example, a PLC tag of the data type "PD_TEL83_IN" with the input start address of the desired channel of the module. The status word "ZSW2_ENC" and the named bits can be found within the created tag structure.
TM Count 2x24V > I/O addresses	–
The organization block ("MC_Servo") and the process image ("PIP OB Servo") are selected automatically for the input and output addresses by selecting the channel in the encoder configuration at the technology object.	
Process image: PIP OB servo	

¹⁾ Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected





–" No configuration for technology module/technology object is required for these parameters

Additional configuration for use with the technology object measuring input

Configuration	
Technology module TM Count 1x24V/TM Count 2x24V	Technology object  Measuring input
TM Count 1x24V/TM Count 2x24V > Channel 0/1 > Operating mode	Hardware interface > Measuring input
Select "Position detection for Motion Control technology object" operating mode	Measuring using PROFIdrive telegram (drive or external encoder)
TM Count 1x24V/TM Count 2x24V > Channel 0/1 > Module parameters You set the parameters for the encoder signals of the channel under "Module parameters" in the operating mode "Position input for Motion Control". The parameters must be set depending on the encoder used.	In the "Number of the measuring input" selection box, select "1" (measuring input 1).

6.4.2 TM PosInput 1/TM PosInput 2 (S7-1500, S7-1500T)


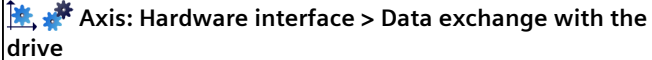
For use with Motion Control, the following parameters must be configured:

Configuration	
Technology module TM PosInput 1 / TM PosInput 2	Technology object  Axis and  external encoder
TM PosInput 1/2 > Channel 0/1 > Operating mode	–
Select "Position detection for Motion Control technology object" mode	
TM PosInput 1/2 > Channel 0/1 > Module parameters You set the parameters for the encoder signals of the channel under "Module parameters" in the "Position input for Motion Control" mode. The parameters must be set depending on the encoder used. The configuration of the encoder is required for use with an SSI absolute encoder. Information on the configuration is available in the documentation for the respective technology module.	Hardware interface > Encoder
–	Select "Encoder" data connection and the channel activated and configured as encoder on the technology module
Signal type • Incremental encoder • Absolute encoder	Select encoder type corresponding to configuration for technology module: • Incremental ¹⁾ • Absolute/Cyclic absolute
–	 Axis: Hardware interface > Data exchange with encoder  External encoder: Hardware interface > Data exchange Telegram "Standard telegram 83" is automatically selected after the selection of the encoder.

¹⁾ Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected

–" No configuration for technology module/technology object is required for these parameters


6.4 Configuring technological modules and onboard I/O for Motion Control (S7-1500, S7-1500T)

Configuration	
Technology module TM PosInput 1 / TM PosInput 2	Technology object  Axis and external encoder
–	<p>Clear the check box "Automatic data exchange for encoder values (online)"</p> <p>Select the check box "Automatic data exchange for encoder values (offline)"</p> <p>If the check box is cleared, you can manually match the parameters described and identified in this table.</p> <p>Select rotary or linear measuring system type ¹⁾</p>
<p>Signal evaluation</p> <ul style="list-style-type: none"> • Single • Double • Quadruple 	<p>Select fine resolution corresponding to configuration on the technology module ¹⁾</p> <ul style="list-style-type: none"> • Incremental encoder: <ul style="list-style-type: none"> – 0 = Single – 1 = Double – 2 = Quadruple • Absolute encoder: <ul style="list-style-type: none"> – 0 = Single
<ul style="list-style-type: none"> • Rotary type: Enter increments per revolution • Linear type: Configuration not relevant 	<ul style="list-style-type: none"> • Rotary type: Enter increments per revolution corresponding to configuration at technology module (1:1) ¹⁾ • Linear type: Enter distance between increments ¹⁾
–	 Axis: Hardware interface > Data exchange with the drive
<ul style="list-style-type: none"> • Rotary type: Enter reference speed corresponding to configuration for technology object (1:1) • Linear type: Configuration not relevant 	Enter reference speed
–	Homing
<p>Select the homing signal for homing mark 0:</p> <ul style="list-style-type: none"> • Signal N of the incremental encoder • DIO 	Use the homing mode "Use zero mark via PROFIdrive telegram".
TM PosInput 1/2 > I/O addresses	–
<p>The organization block ("MC_Servo") and the process image ("PIP OB Servo") are selected automatically for the input and output addresses by selecting the channel in the encoder configuration at the technology object.</p>	

¹⁾ Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected

– No configuration for technology module/technology object is required for these parameters

Additional configuration for use with the technology object measuring input

Configuration	
Technology module TM PosInput 1 / TM PosInput 2	Technology object  Measuring input
TM PosInput 1/2 > Channel 0/1 > Operating mode	Hardware interface > Measuring input
Select operating mode "Position detection for Motion Control technology object"	Measuring using PROFIdrive telegram (drive or external encoder)
TM PosInput 1/2 > Channel 0/1 > Module parameters You set the parameters for the encoder signals of the channel under "Module parameters" in the operating mode "Position input for Motion Control". The parameters must be set depending on the encoder used. The configuration of the encoder is required for use with an SSI absolute encoder. Please note the following information in this regard. Information on the configuration is available in the documentation for the respective technology module.	In the "Number of the measuring input" selection box, select "1" (measuring input 1).

NOTE

When a single-turn absolute encoder is used and two edges are to be measured ("MC_MeasuringInput.Mode" = 2, 3 or 4), the distance between the measured edges of the Measurement input must be < 1 encoder revolution. Otherwise, use a multi-turn absolute encoder.

6.4.3 TM Timer DIDQ 10x24V/TM Timer DIDQ 16x24V (S7-1500, S7-1500T)

You can operate the TM Timer DIDQ technology module centrally on an S7-1500 CPU or decentrally on a distributed I/O. For use with a measuring input, output cam or cam track, the technology module must be used decentrally and with isochronous mode.

NOTICE



Inaccurate time synchronization

When using the technology module TM Timer DIDQ with the technology object Measuring input, Output Cam, or Cam Track, at the same time as the "TIO_Synch" function, the time synchronization of the technology module becomes inaccurate.

Do not use the "TIO_Synch" function if you are using the technology module TM Timer DIDQ with the technology object Measuring input, Output Cam, or Cam Track.



For use with Motion Control, the following parameters must be configured:

Use with output cam/cam track technology object

Configuration	
Technology module TM Timer DIDQ 10x24V/TM Timer DIDQ 16x24V	Technology object  Output cam /  Cam track
Basic parameters	–
Select desired number of outputs under channel configuration (ET 200MP TM Timer DIDQ 16x24V only)	


"–" No configuration for technology module/technology object is required for these parameters

6.4 Configuring technological modules and onboard I/O for Motion Control (S7-1500, S7-1500T)

Configuration	
Technology module TM Timer DIDQ 10x24V/TM Timer DIDQ 16x24V	Technology object  Output cam /  Cam track
Channel parameters	Hardware interface > Output cam output/output cam track
–	Activate output
	Output via Timer DQ
Select "Timer DQ" mode for the respective output	Select output cam output
I/O addresses	–
Select "Isochronous mode"	
The organization block ("MC_Servo") and the process image ("PIP OB Servo") are updated automatically for the input and output addresses by selecting the channel in the output configuration at the technology object.	

"–" No configuration for technology module/technology object is required for these parameters

Use with technology object measuring input


Configuration	
Technology module TM Timer DIDQ 10x24V/TM Timer DIDQ 16x24V	Technology object  Measuring input
Basic parameters	–
Select desired number of inputs under channel configuration	
Channel parameters	Hardware interface > Measuring input
Configuration of DI group: Use inputs individually	–
Select "Timer DI" mode for the respective input	Select measuring via timer DI
–	Select measuring input
Select application-dependent input delay	–
I/O addresses	
Select "Isochronous mode"	
The organization block ("MC_Servo") and the process image ("PIP OB Servo") are updated automatically for the input and output addresses by selecting the channel in the input configuration at the technology object.	

"–" No configuration for technology module/technology object is required for these parameters

6.4.4 TM Pulse 2x24V (S7-1500, S7-1500T)

For use with Motion Control, the parameters described below must each be configured:


Drive connection using PWM (pulse width modulation)

Configuration	
TM Pulse 2x24V	Technology object
	 Axis
TM Pulse 2x24V > Channel configuration	-
Select if you want to use 1 or 2 channels.	
TM Pulse 2x24V > Channel > Operating mode	
Select "Pulse width modulation PWM" or "PWM with DC motor" operating mode	
TM Pulse 2x24V > Channel > Parameters	Hardware interface > Drive
Select "S7 analog output" output format	Select analog drive connection For the selection of the analog output, create a PLC tag of the "Int" type with corresponding address. The offset for the PLC tag to the start address is 2. To activate the output of the PWM signal, set the following two bits of the control interface of the PWM channel in the user program: <ul style="list-style-type: none"> • SW_ENABLE (= Bit 0 in Byte 9) • TM_CTRL_DQ (= Bit 1 in Byte 9) The offset for byte 9 to the start address of the PWM channel is 9.
TM Pulse 2x24V > Channel > I/O addresses	-
Select the organization block "MC_Servo" for the input and output addresses. The "TPA OB Servo" process image is selected automatically for the input and output addresses by selecting the organization block.	

"-" No configuration for technology object is required for these parameters

6.4.5 TM PTO 2x24V (S7-1500, S7-1500T)


For use with Motion Control, the following parameters must be configured.

Configuration	
Technology module TM PTO 2x24V	Technology object  Axis
TM PTO 2 > Channel 0...1 > Operating mode	–
Select signal type: <ul style="list-style-type: none"> PTO (pulse (P) and direction (D)) PTO (count up (A) and count down (B)) PTO (A, B phase-shifted) PTO (A, B phase-shifted - quadruple) 	
Configure the pulse pause for direction reversal	
–	Hardware interface > Drive Select drive type "PROFIdrive" and "Drive" data connection Select the channel at the technology module as drive
	Hardware interface > Encoder Select encoder type "PROFIdrive" and data connection Select the simulated encoder of the PTO 2 channel as encoder
TM PTO 2 > Channel 0...1 > Diagnostic interrupts	–
When the "Enable diagnostic interrupts" check box is selected, diagnostic interrupts are activated if: <ul style="list-style-type: none"> No supply voltage Errors occur at digital outputs The detected error is displayed for the respective channel with feedback bit "Fault_Present" and "Sensor_Error".	
TM PTO 2 > Channel 0...1 > Axis parameters	Data exchange with the drive
–	Telegram "Standard telegram 1" is automatically selected after selection of the drive
	Clear check box "Automatic data exchange for drive values (online)"
	Select check box "Automatic data exchange for drive values (offline)" If the check box is cleared, you can manually match the parameters described and identified in this table.
Enter reference speed corresponding to configuration for technology object (1:1)	Enter reference speed of the drive ¹⁾
Enter maximum speed corresponding to configuration for technology object (1:1)	Enter maximum speed of the drive ¹⁾ If the maximum speed is exceeded, technology alarm 102 is triggered and displayed.
–	Data exchange with encoder
	Telegram "Standard telegram 81" is automatically selected after selection of the encoder

1) Parameters are automatically applied when "Automatic data exchange for drive values (offline)" is selected

2) Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected


– No configuration required for these parameters at the technology module/technology object

Configuration	
Technology module TM PTO 2x24V	Technology object  Axis
–	Clear the check box "Automatic data exchange for encoder values (online)" Select the check box "Automatic data exchange for encoder values (offline)" If the check box is cleared, you can manually match the parameters described and identified in this table. Select rotary measuring system type ²⁾
Enter increments per revolution	Enter increments per revolution corresponding to configuration at technology module (1:1) ²⁾
-	Enter fine resolution ²⁾ • 0 = Single
Configure stop behavior • Quick stop time • Ramp stop time	–
TM PTO 2 > Channel 0...1 > Hardware inputs/outputs	–
If you want to use a hardware output to enable the drive, select the "Use ED as drive enable" check box.	No setting required at the technology object. The output is automatically controlled by the "MC_Power".
	Homing
For the hardware input (DIO), select the "Use RS as reference switch" check box.	Use the homing mode "Use zero mark via PROFIdrive telegram".
Select the edge of the hardware input for triggering the reference output cam function.	
	"Measuring input > Configuration > Hardware interface" technology object
When using a measuring input, select the "Use MI as measuring input" check box.	Select the measuring input type "Measuring via PROFIdrive telegram (drive or external encoder)". Select the measuring input "1" under hardware connection.
When using Drive ready, select the "Use Rdy as ready signal" check box.	No setting required at the technology object. When the input is used, "MC_Power" waits until the input signal is present before it sets the drive enable.
Configuring input delay	–
TM PTO 2 > Channel 0...1 > Sign-of-life error	
Configure tolerated number of sign-of-life errors	
TM PTO 2 > I/O addresses	
The organization block ("MC_Servo") and the process image ("PIP OB Servo") are selected automatically for the input and output addresses by selecting the PTO channel at the technology object.	

1) Parameters are automatically applied when "Automatic data exchange for drive values (offline)" is selected
 2) Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected
 "-" No configuration required for these parameters at the technology module/technology object

6.4.6 TM PTO 4 (S7-1500, S7-1500T)


For use with Motion Control, the following parameters must be configured.

Configuration	
Technology module TM PTO 4	Technology object  Axis
TM PTO 4 > Channel configuration	-
Configure the number of channels (1 to 4) you want to use.	
TM PTO 4 > Channel 0...3 > Operating mode	
Select signal type: <ul style="list-style-type: none"> • PTO (pulse (P) and direction (D)) • PTO (count up (A) and count down (B)) • PTO (A, B phase-shifted) • PTO (A, B phase-shifted - quadruple) 	
Select signal interface: <ul style="list-style-type: none"> • RS422, symmetrical/TTL (5V), asymmetrical • 24V asymmetric 	
Configure the interpulse pause for direction reversal.	Hardware interface > Drive Select drive type "PROFIdrive" and "Drive" data connection. Select the pulse output configured at the technology module as drive. Hardware interface > Encoder The encoder of the actuator telegram (simulated encoder) is automatically selected. Alternatively, an existing encoder interface can be selected.
-	
-	
TM PTO 4 > Channel 0...3 > Diagnostic interrupts	-
When the "Enable diagnostic interrupts" check box is selected, diagnostic interrupts are activated if: <ul style="list-style-type: none"> • No supply voltage • Errors occur at digital outputs The detected error is displayed for the respective channel with feedback bit "Fault_Present" and "Sensor_Error".	
TM PTO 4 > Channel 0...3 > Axis parameters	Data exchange with the drive
-	Telegram "Standard telegram 3" is automatically selected after the selection of the drive.
	Clear check box "Automatic data exchange for drive values (online)"
	Select check box "Automatic data exchange for drive values (offline)" If the check box is cleared, you can manually match the parameters described and identified in this table.
Enter reference speed corresponding to configuration for technology object (1:1)	Enter reference speed of the drive ¹⁾

1) Parameters are automatically applied when "Automatic data exchange for drive values (offline)" is selected


2) Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected

"-" No configuration required for these parameters at the technology module/technology object

Configuration	
Technology module TM PTO 4	Technology object  Axis
Enter maximum speed corresponding to configuration for technology object (1:1)	Enter maximum speed of the drive ¹⁾ If the maximum speed is exceeded, technology alarm 102 is triggered and displayed.
–	Data exchange with encoder Telegram "Standard telegram 83" is automatically selected after the selection of the encoder. Clear the check box "Automatic data exchange for encoder values (online)" Select the check box "Automatic data exchange for encoder values (offline)" If the check box is cleared, you can manually match the parameters described and identified in this table. Select rotary measuring system type ²⁾
Enter increments per revolution	Enter increments per revolution corresponding to configuration at technology module (1:1) ²⁾
-	Enter fine resolution ²⁾ • 0 = Single
Configure stop behavior • Quick stop time • Ramp stop time	–
TM PTO 4 > Channel 0...3 > Hardware inputs/outputs	–
If you want to use a hardware output to enable the drive, select the "Use drive enable" check box. Next select one of the two hardware outputs DQ0 or DIQ2.	No setting required at the technology object. The output is automatically controlled by the "MC_Power".
	Homing
Activate the hardware input (DIO) for the reference cam.	Use the homing mode "Use zero mark via PROFIdrive telegram".
Select the edge of the hardware input for triggering the reference output cam function.	
When using a measuring input, select the "Use DI1 check box as measuring input".	"Measuring input > Configuration > Hardware interface" technology object Select the measuring input type "Measuring via PROFIdrive telegram (drive or external encoder)". Select the measuring input "1" under hardware connection.
Select the "Use "drive ready"" check box. In the ""Drive ready" input", select the hardware input that is to be used to display whether the drive is ready.	No setting required at the technology object. When the input is used, "MC_Power" waits until the input signal is present before it sets the drive enable.
Configuring input delay	–
TM PTO 4 > Channel 0...3 > Sign-of-life error	

1) Parameters are automatically applied when "Automatic data exchange for drive values (offline)" is selected
 2) Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected
 "–" No configuration required for these parameters at the technology module/technology object

6.4 Configuring technological modules and onboard I/O for Motion Control (S7-1500, S7-1500T)


Configuration	
Technology module TM PTO 4	Technology object  Axis
Configure tolerated number of sign-of-life errors	–
TM PTO 4 > I/O addresses	
The organization block ("MC_Servo") and the process image ("PIP OB Servo") are selected automatically for the input and output addresses by selecting the PTO channel at the technology object.	

- 1) Parameters are automatically applied when "Automatic data exchange for drive values (offline)" is selected
 - 2) Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected
- " No configuration required for these parameters at the technology module/technology object


6.4.7 CPU 1511C-1 PN/CPU 1512C-1 PN (onboard I/O devices) (S7-1500, S7-1500T)

For use with Motion Control, the parameters described below must be configured.

Drive connection via PTO (Pulse Train Output)

Configuration	
CPU 1511C-1 PN/CPU 1512C-1 PN	Technology object  Axis
Pulse generators (PTO/PWM) > PTO1...4/PWM1...4 > General	–
To activate a channel for PTO mode, select one of the following operating modes: <ul style="list-style-type: none"> • PTO (pulse (A) and direction (B)) • PTO (count up (A), count down (B)) • PTO (A,B phase shifted) • PTO (A,B phase shifted, quadruple) 	
–	Hardware interface > Drive Select drive type "PROFIdrive" and "Drive" data connection. Select the pulse generator of the CPU configured for PTO mode as drive.
	Hardware interface > Encoder The encoder of the actuator telegram (simulated encoder) is automatically selected. Alternatively, an existing encoder interface can be selected.
Pulse generators (PTO/PWM) > PTO1...4/PWM1...4 > Axis parameters	Hardware interface > Data exchange with the drive
–	Telegram "Standard telegram 3" is automatically selected after the selection of the drive.

- 1) Parameters are automatically applied when "Automatic data exchange for drive values (offline)" is selected
 - 2) Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected
- " No configuration for CPU/technology object is required for these parameters


Configuration	
CPU 1511C-1 PN/CPU 1512C-1 PN	Technology object 
–	Clear check box "Automatic data exchange for drive values (online)"
	Select check box "Automatic data exchange for drive values (offline)" If the check box is cleared, you can manually match the parameters described and identified in this table.
Enter reference speed corresponding to configuration for technology object (1:1)	Enter reference speed of the drive ¹⁾
Enter reference speed corresponding to configuration for technology object (1:1)	Enter maximum speed of the drive ¹⁾ If the maximum speed is exceeded, technology alarm 102 is triggered and displayed.
–	Hardware interface > Data exchange with encoder
	Telegram "Standard telegram 3" is automatically selected after the selection of the encoder.
	Clear the check box "Automatic data exchange for encoder values (online)"
	Select the check box "Automatic data exchange for encoder values (offline)" If the check box is cleared, you can manually match the parameters described and identified in this table.
	Select rotary measuring system type ²⁾
Enter increments per revolution	Enter increments per revolution corresponding to configuration for CPU (1:1) ²⁾
The fine resolution has the fixed value "0 bit" (= single) and cannot be changed.	Enter the fine resolution corresponding to the configuration on the CPU ²⁾ Bits in incr. actual value (G1_XIST1): 0 (= single)
Pulse generators (PTO/PWM) > PTO1...4/PWM1...4 > Hardware inputs/outputs	Homing
Select the hardware input for the reference switch In addition, configure the input delay for the selected hardware input. You configure the input delay in the device configuration at the corresponding DI channel (DI 16/DQ 16 > Inputs > Channel > Input parameters > Input delay).	Use the homing mode "Use zero mark via PROFIdrive telegram" for drive connection via PTO.
Select the edge of the hardware input for triggering the reference output cam function.	
When using a measuring input, select the hardware input of the measuring input. The following table includes the configuration description.	–

¹⁾ Parameters are automatically applied when "Automatic data exchange for drive values (offline)" is selected

²⁾ Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected


"–" No configuration for CPU/technology object is required for these parameters

6.4 Configuring technological modules and onboard I/O for Motion Control (S7-1500, S7-1500T)

Configuration	
CPU 1511C-1 PN/CPU 1512C-1 PN	Technology object
	 Axis
Select the hardware input that is used to display whether the drive is ready. In addition, configure the input delay for the selected hardware input. You configure the input delay in the device configuration at the corresponding DI channel (DI 16/DQ 16 > Inputs > Channel > Input parameters > Input delay).	–
With selected "PTO (pulse (A) and direction (B))" mode, the hardware output for the PTO signal A ("Pulse output (A)") is automatically selected through the device configuration and cannot be changed. For PTO signal B ("Direction output (B)") select one of the hardware outputs offered in the selection box. The hardware outputs for the PTO signals are selected through the device configuration for the following operating modes and cannot be changed: <ul style="list-style-type: none"> • PTO (count up (A), count down (B)) • PTO (A,B phase shifted) • PTO (A,B phase shifted, quadruple) 	
Pulse generators (PTO/PWM) > PTO1...4/PWM1...4 > I/O addresses	
The organization block ("MC_Servo") and the process image ("PIP OB Servo") are selected automatically for the input and output addresses by selecting the PTO channel at the technology object.	


- 1) Parameters are automatically applied when "Automatic data exchange for drive values (offline)" is selected
 - 2) Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected
- No configuration for CPU/technology object is required for these parameters

Additional configuration for use with the technology object measuring input

Configuration	
Technology module	Technology object
CPU 1511C-1 PN/CPU 1512C-1 PN	 Measuring input
Pulse generators (PTO/PWM) > PTO1...4/PWM1...4 > Hardware inputs/outputs	Hardware interface > Measuring input
Select the hardware input of the measuring input. In addition, configure the input delay for the selected hardware input. You configure the input delay in the device configuration at the corresponding DI channel (DI 16/DQ 16 > Inputs > Channel > Input parameters > Input delay).	Measuring using PROFIdrive telegram (drive or external encoder) In the "Number of the measuring input" selection box, select "1" (measuring input 1).



Drive connection using PWM (pulse width modulation)

Note that only travel in the positive direction is possible with a drive connection using the integrated PWM function of the compact CPU.

Configuration	
CPU 1511C-1 PN/CPU 1512C-1 PN	Technology object
	 Speed axis
Pulse generators (PTO/PWM) > PTO1...4/PWM1...4 > General	-
Select "Pulse width modulation PWM" mode	
Pulse generators (PTO/PWM) > PTO1...4/PWM1...4 > Hardware inputs/outputs	
Select the hardware output to be used for pulse output. Select whether the set hardware output is to work as a fast push-pull switch or as P switch.	
Pulse generators (PTO/PWM) > PTO1...4/PWM1...4 > Parameters	Hardware interface > Drive
Select "S7 analog output" output format	Select analog drive connection For the selection of the analog output, create a PLC tag of the "Int" type with corresponding address. The offset for the PLC tag of the control interface of the PWM channel is 2. To activate the output of the PWM signal, set the following two bits of the control interface of the PWM channel in the user program: <ul style="list-style-type: none"> • SW_ENABLE (= bit 0 in byte 9) • TM_CTRL_DQ (= Bit 1 in Byte 9) The offset for byte 9 to the start address of the PWM channel is 9.
Select minimum pulse width of 0 µs	-
Select required period duration (e.g. 100 µs)	
Pulse generators (PTO/PWM) > PTO1...4/PWM1...4 > I/O addresses	
Select the organization block "MC_Servo" for the input and output addresses. The "TPA OB Servo" process image is selected automatically for the input and output addresses by selecting the organization block.	



"-" No configuration for technology object is required for these parameters

Encoder connection via HSC (High Speed Counter)

Configuration		
CPU 1511C-1 PN/CPU 1512C-1 PN	Technology object	
	 Axis	 External encoder
High-speed counter (HSC) > HSC 1...6 > General > Enable	–	–
Enable high-speed counter		
High-speed counter (HSC) > HSC 1...6 > Basic parameters > Operating mode		
Select "Position input for Motion Control" mode		
High-speed counter (HSC) > HSC 1...6 > Basic parameters > Module parameters	Hardware interface > Encoder	Hardware interface > Encoder
–	Select "Encoder" data connection and the high-speed counter activated and configured as encoder on the CPU	Select "Encoder" data connection and the high-speed counter activated and configured as encoder on the CPU
Signal type • Incremental encoder	Select encoder type according to the device configuration of the CPU ¹⁾ • Incremental	Select encoder type according to the device configuration of the CPU ¹⁾ • Incremental
–	Hardware interface > Data exchange with encoder	Hardware interface > Data exchange
	Telegram "Standard telegram 83" is automatically selected after the selection of the encoder.	Telegram "Standard telegram 83" is automatically selected after the selection of the encoder.
	Clear the check box "Automatic data exchange for encoder values (online)"	Clear the check box "Automatic data exchange for encoder values (online)"
	Select the check box "Automatic data exchange for encoder values (offline)" If the check box is cleared, you can manually match the parameters described and identified in this table.	Select the check box "Automatic data exchange for encoder values (offline)" If the check box is cleared, you can manually match the parameters described and identified in this table.
	Select rotary measuring system type ¹⁾	Select rotary measuring system type ¹⁾
Signal evaluation • Single • Double • Quadruple	Enter fine resolution according to the configured signal evaluation set for the high-speed counter (HSC) ¹⁾ • 0 = Single • 1 = Double • 2 = Quadruple	Enter fine resolution according to the configured signal evaluation set for the high-speed counter (HSC) ¹⁾ • 0 = Single • 1 = Double • 2 = Quadruple
Enter increments per revolution	Enter increments per revolution corresponding to device configuration for CPU (1:1) ¹⁾	Enter increments per revolution corresponding to device configuration for CPU (1:1) ¹⁾
–	Hardware interface > Data exchange with the drive	–
Enter reference speed corresponding to configuration for technology object (1:1)	Enter reference speed	

¹⁾ Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected


–" No configuration for CPU/technology object is required for these parameters

Configuration		
CPU 1511C-1 PN/CPU 1512C-1 PN	Technology object	
	 Axis	 External encoder
–	Homing	Homing
Select the homing signal for homing mark 0: <ul style="list-style-type: none"> Signal N of the incremental encoder DIO (can be set with the hardware inputs/outputs) In addition, configure the input delay for the selected hardware input. You configure the input delay in the device configuration at the corresponding DI channel (DI 16/DQ 16 > Inputs > Channel > Input parameters > Input delay).	Use the homing mode "Use zero mark via PROFIdrive telegram".	Use the homing mode "Use zero mark via PROFIdrive telegram".
High-speed counter (HSC) > HSC 1...6 > I/O addresses	–	–
The organization block ("MC_Servo") and the process image ("PIP OB Servo") are selected automatically for the input and output addresses by selecting the HSC channel at the technology object.		

1) Parameters are automatically applied when "Automatic data exchange for encoder values (offline)" is selected

"–" No configuration for CPU/technology object is required for these parameters

Additional configuration for use with the technology object measuring input

Configuration	
Technology module CPU 1511C-1 PN/CPU 1512C-1 PN	Technology object
	 Measuring input
High-speed counter (HSC) > HSC 1...6 > Hardware inputs/outputs	Hardware interface > Measuring input
Select the hardware input of the measuring input. In addition, configure the input delay for the selected hardware input. You configure the input delay in the device configuration at the corresponding DI channel (DI 16/DQ 16 > Inputs > Channel > Input parameters > Input delay).	Measuring using PROFIdrive telegram (drive or external encoder) In the "Number of the measuring input" selection box, select "1" (measuring input 1).



6.4.8 SIMATIC Drive Controller (onboard I/O) (S7-1500, S7-1500T)

You can use the inputs and outputs of interface X142 of a SIMATIC Drive Controller as measuring input for the measuring input technology object, as well as for the output cam/cam track technology object.

Isochronous mode is required for use with a measuring input, output cam or cam track.


For use with Motion Control, the following parameters must be configured:

Use with output cam/cam track technology object

Configuration	
SIMATIC Drive Controller	Technology object
	 Output cam /  Cam track
Channel parameters > Channel 0 to 7	Hardware interface > Output cam output/output cam track
–	Activate output
–	Output via Timer DQ
Select desired channel and select operating mode "Timer DQ".	Select output cam output
I/O addresses	–
Select "Isochronous mode"	
The organization block ("MC_Servo") and the process image ("PIP OB Servo") are updated automatically for the input and output addresses by selecting the channel in the output configuration at the technology object.	

"–" No configuration at the SIMATIC Drive Controller/technology object is required for these parameters

Use with measuring input technology object

Configuration	
SIMATIC Drive Controller	Technology object
	 Measuring input
Channel parameters > Channel 0 to 7	Hardware interface > Measuring input
–	Select measuring via timer DI
Select desired channel and select operating mode "Timer DI"	Select measuring input
I/O addresses	–
Select "Isochronous mode"	
The organization block ("MC_Servo") and the process image ("PIP OB Servo") are updated automatically for the input and output addresses by selecting the channel in the input configuration at the technology object.	

"–" No configuration at the SIMATIC Drive Controller/technology object is required for these parameters

Programming (S7-1500, S7-1500T)

User program

The "Programming" section contains general information on supplying and evaluating the Motion Control instructions and on the technology data block.

You can use Motion Control instructions in the user program to assign jobs to the technology object. You define the job using the input parameters of the Motion Control instructions. You can track the current job status using the job parameters if you use a separate instance for each Motion Control instruction per technology object.

In a typical programming, you can use one or more instances for each Motion Control instruction for each technology object.

The use of a separate instance per technology object is always necessary for Motion Control instructions without parameter "DONE" and for the Motion Control instruction "MC_MoveJog".

Only one instance per technology object may be active in the program flow for the Motion Control instruction "MC_Power". Disable the technology object with the same instance you used to enable the technology object, otherwise an error with error ID 16#800C will occur. The technology data block is available to you as an additional interface to the technology object.

Interpreter programs

You can find information on the Interpreter program in the documentation S7-1500 Interpreter functions ([Page 8](#))

7.1 Process response (S7-1500, S7-1500T)

7.1.1 Organization blocks for Motion Control (S7-1500, S7-1500T)

When you create a technology object, organization blocks MC_Servo and MC_Interpolator are automatically created for processing the technology objects. The organization block MC_LookAhead is also created for the kinematics technology object as of technology version 5.0. The technology objects are processed in the motion control application cycle. The application cycle consists of required and optional organization blocks (OBs).

In the user program, call the appropriate motion control instruction and start a motion control job for a technology object. In the organization block Main, call the user program cyclically.

Optionally, programmable motion control OBs are available, which you must insert manually. These organization blocks take into account special requirements with regard to time-critical events or the time sequence of function calls. This makes it possible, for example, to start motions immediately in the event of time-critical events.

The following table shows the organization blocks for motion control:

Organization block	Description	Priority ¹⁾
MC_PreServo [OB67] (optional)	For For example: Preparation of the telegram contents from the drive system. Is called immediately before the MC_Servo.	Corresponds to MC_Servo
MC_Servo [OB91] (know-how-protected)	Calculation of the position controller System performance, no user program possible.	17 to 26 Default 26
MC_PostServo [OB95] (optional)	For For example: Preparation of the setpoints for the drive system. Is called immediately after the MC_Servo.	Corresponds to MC_Servo
MC_Transformation [OB98] (optional)	Programming of the transformation of Cartesian coordinates and axis-specific setpoints for user-defined kinematics	17 to 25 Default 25
MC_PreInterpolator [OB68] (optional)	For For example: MotionIn instructions, and instructions for measuring inputs, output cams and cam tracks The MC_PreInterpolator is required for iposynchronous processing of Motion Control instructions. Is called immediately before the MC_Interpolator.	Corresponds to MC_Interpolator
MC_Interpolator [OB92] (know-how-protected)	Evaluation of the motion control instructions, generation of setpoints and monitoring functionality System performance, no user program possible.	16 to 25 Default 24
MC_LookAhead [OB97] (know-how-protected)	Calculation of the motion processing of the kinematics technology object Applies only to a technology object kinematics V5.0 or higher. System performance, no user program possible.	15 to 16 Default 15

¹⁾ 26 corresponds to highest priority.

The clock ratio of the two organization blocks MC_Servo and MC_Interpolator to each other is always 1:1. You can scale the ratio of the bus clock to the application cycle.

You can set the application cycle and the priority of the organization blocks according to your requirements for control quality and system load.

You can check the runtime of the respective organization block (except MC_LookAhead) with the instruction "RT_INFO". The current application cycle (in μ s) of the organization blocks MC_PreServo, MC_PostServo and MC_PreInterpolator can be read using the start information.

Application cycle

In the properties of the organization block MC_Servo, you can set the application cycle in which the organization blocks for MC_Servo, MC_Interpolator and their optional OBs are called:

- **Synchronous to the bus (recommended setting for optimum control quality)**

The application cycle is synchronous to the selected source of the send clock and the corresponding reduction ratio. The following clock sources are available for selection:

- PROFINET IO
- PROFIBUS DP
- Local bus system (as of firmware version \geq 2.6)
- PROFIdrive system for SINAMICS Integrated of SIMATIC Drive Controller

A bus system that is connected to the CPU via a communications processor/communication module (CP/CM) cannot be used synchronously.

- **Cyclic**
The application cycle is called at the specified time interval. Processing is asynchronous to the bus clock/send clock.

Ways to influence the process response

The system load is primarily determined by the quantity structure (number of technology objects), the communication load and the user program. The processing time in the application cycle increases with the number of technology objects (MC_Servo). Simultaneous starting of Motion Control instructions results in additional processing times in the application cycle (MC_Interpolator and MC_LookAhead) at short notice. Optional motion control OBs additionally affect the processing time of the application cycle.

You define the available processing time using the set application cycle. The time available at the end of the application cycle is used for processing the low-priority OBs with the other user program.

The cycle time of the Main can increase significantly due to longer processing times in the application cycle.

System overloads are indicated by timeouts or overflows (Page 90) of Main, MC_Servo and MC_Interpolator.

You have the following options for influencing the system load and the processing times of the user program:

- Adjusting the application cycle:
 - Adjusting the send clock of the bus system
 - Reduce clock
- Adjusting the percentage cycle load of the MC_LookAhead (adjustment range 1% to 40%, default setting 20%)
- Adjusting the percentage communication load of the CPU
- To relieve the load on MC_Interpolator and MC_LookAhead, avoid simultaneous starting of Motion Control instructions
- Improving the system performance for the cam interpolation in the MC_Interpolator and for the motion preparation of the kinematics technology object in the MC_LookAhead using the properties under "General > Multi-core processor" (S7-1500T FW V3.0 and higher)
If both the options are activated, a cam interpolation can interrupt the motion preparation of the kinematics technology object.

If necessary, use one or more options to optimize the system and the process response.

Reduction ratio

You can reduce the application cycle of the MC_Servo relative to the send clock of the selected isochronous bus system. You can set an integer multiple of the send clock as the factor. A maximum cycle time of 32 ms is possible for the application cycle.

Note the following settings when you call an isochronous mode interrupt OB (OB6x) and the MC_Servo.

- Call both organization blocks with the same source of the send clock (same bus system).
- Use the same application cycle both organization blocks (in ms).

Priority

You can configure the priority of the organization blocks as needed in their properties under "General > Properties > Priority".

When setting the priorities, make sure that the MC_Servo is always set before the MC_PreInterpolator and the MC_Interpolator. The priority of MC_Servo must be at least one level higher than the priority of MC_Interpolator. The priority of the MC_LookAhead must be higher than the priority of cyclic interrupts.

7.1.2 Process image partition "OB Servo PIP" (S7-1500, S7-1500T)

The process image partition "PIP OB Servo" is made available in isochronous mode for Motion Control when MC_Servo is called. All drives and encoders used by Motion Control are assigned to this process image partition.

Because the organization blocks MC_PreServo and MC_PostServo are called automatically by the MC_Servo, the process image partition is also available automatically. If you use a MC_PreServo, the data is read in when the MC_PreServo starts. If you use a MC_PostServo, the data is output after the MC_PostServo.

Additionally, you should assign all I/O modules used by Motion Control to this process image partition (e.g. hardware limit switches). The assignment results in chronologically synchronous processing with the technology object.

The input process image partition is also updated in STOP mode.

Process image partition in the user program

As of firmware version V1.5, you can access the process image partition "OB Servo PIP" in your user program. This makes it possible to evaluate the process image partition using the trace function.

Process image partition in watch/force table

Access to data of the process image partition "PIP OB Servo" via watch and force tables is not possible.

7.1.3 Operational Sequence and Timeouts (S7-1500, S7-1500T)

When processing the Motion Control functionality, the Motion Control organization blocks MC_Servo and MC_Interpolator, including the optional organization blocks, are called and processed in each application cycle. Your user program is processed during the remaining time until the next application cycle.

For error-free program execution, keep to the following rules:

- In each application cycle, MC_Servo must be started and executed completely.
- In each application cycle, the relevant MC_Interpolator must at least be started.
- For drives connected via PROFINET, the MC_PreServo, the MC_Servo, and the MC_PostServo must be completely processed in the first PROFINET cycle.

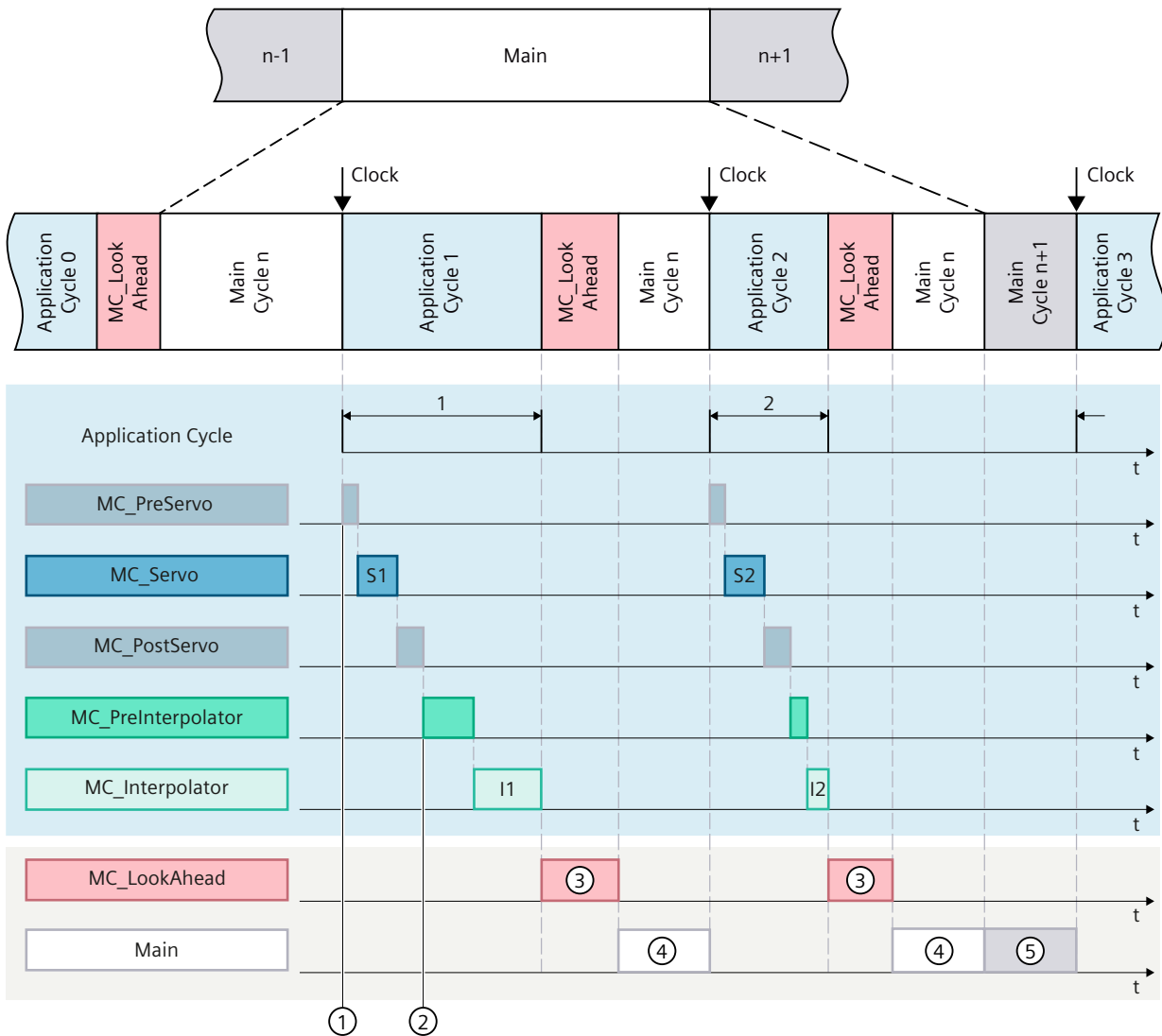
Example

The figure below illustrates the time sequence of the cyclic user program and the application cycle:

- The upper section shows the processing of the Main without interruption of the application cycle by Motion Control OBs with higher priority.
- The central section shows the processing of Main with interruption Motion control OBs with higher priority are executed in the application cycle.

The Main is interrupted in the cycle of the application cycle. The cycle time of the user program becomes correspondingly longer.

- The lower section shows a detailed view of the error-free process response of the individual organization blocks.



- ① "TPA OB Servo" input process image partition
- ② "TPA OB Servo" output process image partition
- ③ MC_LookAhead cycle
- ④ Main cycle n
- ⑤ Main cycle n+1

The Motion Control OBs and the Main are processed one after the other in application cycle 1. The process image partition "PIP OB Servo" ① is read before processing the MC_PreServo. The MC_Servo is displayed as "S1" in the first application cycle. After the processing of the MC_PostServo, the process image partition "PIP OB Servo" ② is updated.

The MC_PreInterpolator and the MC_Interpolator are then processed. The MC_Interpolator is displayed as "I1" in the first application cycle. Its processing time varies according to the evaluation of the motion control instructions as well as the monitoring and setpoint generation for all technology objects configured on the CPU for motion control.

Number ③ represents the processing of the MC_LookAhead . The processing of the MC_LookAhead is interrupted in the first application cycle and continued in the second application cycle.

The Main ④ is further processed only after all Motion Control OBs have been processed. In the second application cycle, the processing time for the MC_Interpolator "I2" and MC_LookAhead cycle ③ is shorter than in the first application cycle. The MC_LookAhead ③ is processed further in the second application cycle. The Main cycle n ④ is finished before the third application cycle. The Main cycle n+1 ⑤ is already processed in the time still remaining until the third application cycle. This means that parts of two Main cycles can be processed between two application cycles.

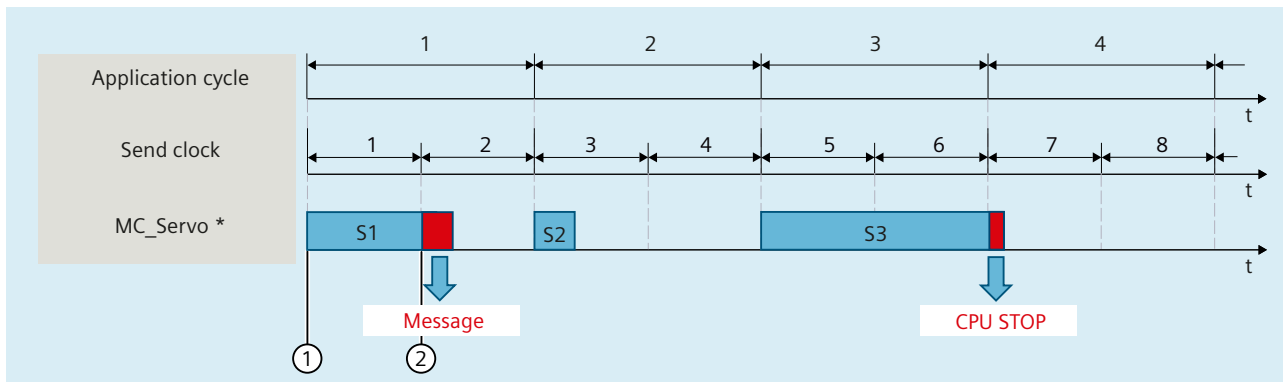
Overflows

Overflows can occur if the configured application cycle is not adhered to, for example because additional technology objects or programs are added in the MC_PreServo or MC_PostServo. The application cycle must be adapted in this case. The MC_Servo must be completed before the next send clock, irrespective of the permissible duration of the application cycle.

If the processing time of the MC_Servo (including MC_PreServo and/or MC_PostServo, if available) exceeds the duration of a send clock, the message "Overflow" is displayed in the diagnostic buffer of CPU. The controller no longer runs isochronously.

If the processing time exceeds the duration of an application cycle, the CPU switches to STOP operating state.

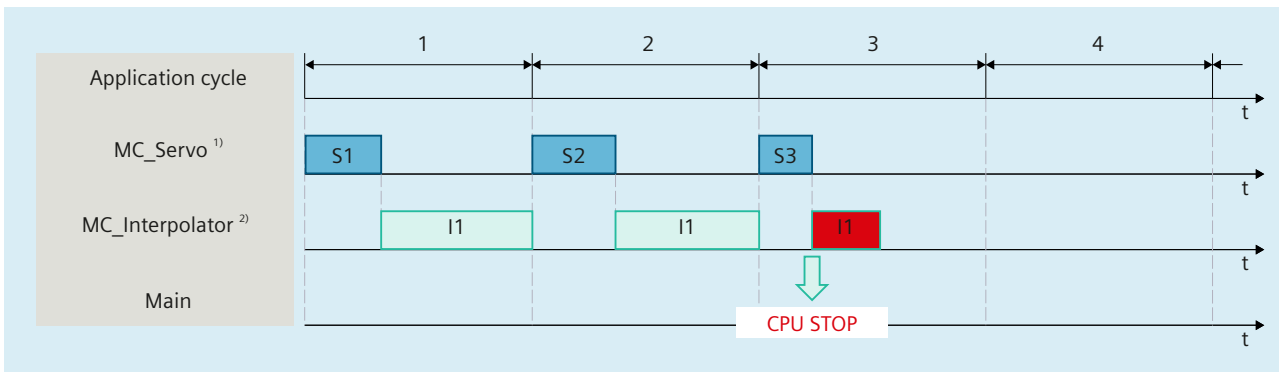
The following figure shows the process response in the case of overflow of MC_Servo in the application cycle and in the send clock with a reduction ratio of 2:



- ① Start of the processing of MC_Servo
- ② Overflow (message)
- * Including MC_PreServo and/or MC_PostServo, if present

The execution of an MC_Interpolator may only be interrupted by an MC_Servo call. If more interruptions occur, the CPU switches to STOP mode.

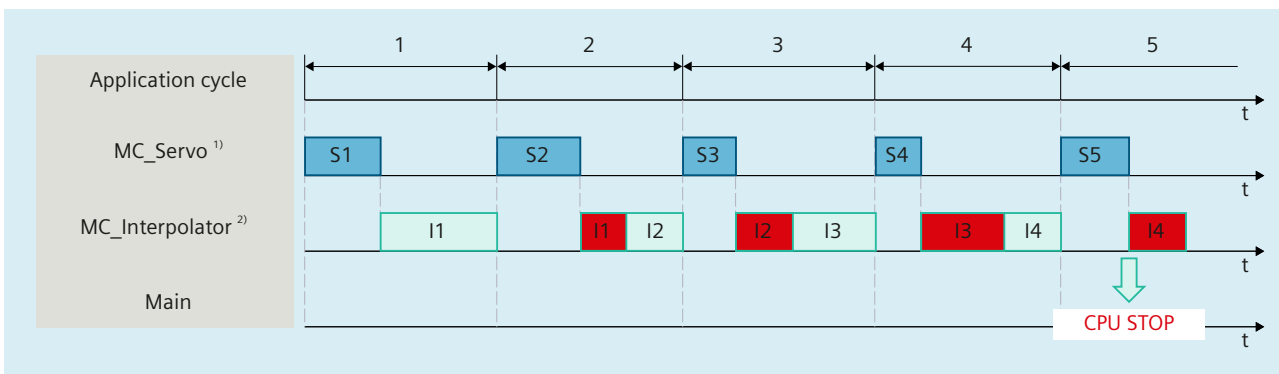
The following figure shows the process response when an MC_Interpolator is interrupted over two time slices:



- 1) Including MC_PreServo and/or MC_PostServo, if present
- 2) Including MC_PreInterpolator, if present

The CPU tolerates a maximum of three consecutive MC_Interpolator overflows. If more overflows occur, the CPU switches to STOP mode.

The following figure shows the process response if there are four consecutive individual overflows of MC_Interpolator:



- 1) Including MC_PreServo and/or MC_PostServo, if present
- 2) Including MC_PreInterpolator, if present

7.1.4 Operating modes (S7-1500, S7-1500T)

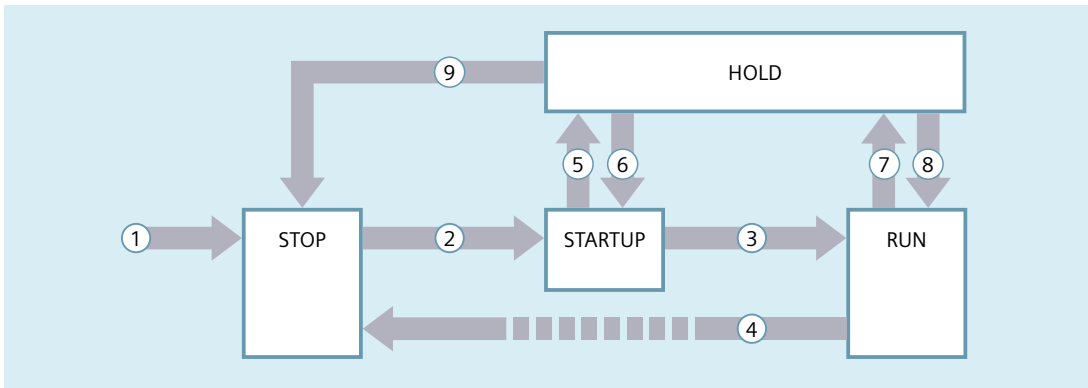
This section examines the behavior of the Motion Control in the respective operating modes and in the transitions between operating modes. A general description of the operating modes can be found in the S7-1500 System Manual

Operating modes and transitions

The CPU has the following operating modes:

- STOP
- STARTUP
- RUN
- HOLD

The following figure shows the operating modes and the operating mode transitions:



Operating mode transitions

The following table shows the behavior of the Motion Control in the transitions between the operating modes:

No.	Operating mode transition	Behavior
①	POWER ON → STOP	The CPU performs a restart of the technology objects. The technology objects are reinitialized with the values from the load memory.
②	STOP → STARTUP	Not relevant for Motion Control.
③	STARTUP → RUN	The process outputs are enabled.
④	RUN → STOP	When the CPU changes from RUN mode to STOP mode, all technology objects are disabled in accordance with the alarm response "remove enable". Active Motion Control jobs are aborted. If restart-relevant data has been changed for technology objects in RUN, the CPU performs a restart of the corresponding technology objects.
⑤	STARTUP → HOLD	Breakpoint in the start-up routine reached.
⑥	HOLD → STARTUP	Not possible when using technology objects

No.	Operating mode transition	Behavior
⑦	RUN → HOLD	Breakpoint reached
⑧	HOLD → RUN	Not possible when using technology objects
⑨	HOLD → STOP	By operation of switch/display or by setting to STOP from programming device.

STOP mode

In STOP mode the user program is not processed and all process outputs are disabled. Thus no Motion Control jobs are executed.

The technology data blocks are updated.

STARTUP mode

Before the CPU starts processing of the cyclical user program, the startup OBs are processed one time.

In STARTUP mode, the process outputs are disabled. Motion Control jobs are rejected.

The technology data blocks are updated.

RUN mode

The user program is processed in RUN mode.

In RUN mode, the programmed Motion Control jobs are cyclically called and processed.

The technology data blocks are updated.

HOLD operating state

Working with breakpoints is not supported when technology objects are used. An overflow of the MC_Servo hereby occurs. This leads to an immediate switch to STOP mode.

In HOLD operating state, events are not initiated and the user program is not executed.

The outputs in the HOLD operating state behave as parameterized. Outputs either supply a configured substitute value or keep the last value output and bring the controlled process to a safe operating state.

When you reach a breakpoint, the CPU executes an implicit restart of the technology object.

Homing the technology once again.

7.2 Technology data block (S7-1500, S7-1500T)

The properties of real objects (e.g. axes) are configured by means of the technology objects and saved in a technology data block. The technology data block contains all configuration data, setpoint and actual values, and status information of the technology object. The TIA Portal automatically creates the technology data block when the technology object is created. You access the data of the technology data block (read/write access) with your user program. You can find a list and description at the tags of the corresponding technology object.

7.2.1 Evaluating the technology data block (S7-1500, S7-1500T)

Access to data in the technology data block occurs in accordance with the access to standard data blocks. Only tags with elementary data types can be accessed in the technology data block. Access to a complete data structure (e.g. STRUCT, ARRAY) is not possible.

Reading values from the technology data block

In your user program you can read actual values (e.g. current position) and status information, or detect error messages in the technology object. When you program a query in your user program (e.g. current velocity), the value is directly read from the technology object.

Reading values from the technology data block takes longer than for other data blocks. If you use these tags several times in a single cycle of your user program, it is recommended to copy the tag values to local tags, and use the local tags in your program.

Writing values to the technology data block

The configuration of the technology object in the TIA Portal is used to write the corresponding data to the technology data block. After they have been loaded into the CPU, these data are stored in the CPU on the SIMATIC Memory Card (load memory).

In the following cases, it may be necessary for the user program to write values to the technology data block:

- Adaptation of the configuration of the technology object (e.g. dynamic limits, software limit switches)
- Use of overrides
- Adaptation of position control (e.g. "Kv" parameter)

Changes to values in the technology data block by the user program can take effect at various points in time. The respective properties of the individual tags are described in the tags of the corresponding technology object:

Effectiveness of changes	Description
Direct (DIR)	You write changes using direct assignments. The changes are applied only at the start of the next MC_Servo. The changes are retained until the next POWER OFF of the CPU or restart of the technology object.

Effectiveness of changes	Description	
Direct (DIR) (e.g. <TO>.Override.Velocity)	LREAL (e.g. <TO>.Override.Velocity)	The technology object performs a range check on the written value, and immediately starts using the new value. If range limits are violated when writing, the technology object automatically corrects the values. If the value is below the range, then the value is set to the low limit of the range; if the range is exceeded, then the value is set to the high limit of the range.
	DINT/BOOL (e.g. <TO>.PositionLimits_SW.Active)	Changes are only permitted in the defined value range. Value changes outside the value range are not applied. If you enter invalid values, the programming error OB (OB 121) is started.
When Motion Control instruction is called (CAL) (e.g. <TO>.Sensor[1..4].ActiveHoming.HomePositionOffset)	You write changes using direct assignments. The changes are applied at the start of the next MC_Servo after the call of the corresponding Motion Control instruction in the user program. The changes are retained until the next POWER OFF of the CPU or restart of the technology object.	
Restart (RES) (e.g. <TO>.Homing.AutoReversal)	Since restart-relevant tags have dependencies on other tags, value changes cannot be applied at any arbitrary time. The changes are only used after reinitialization (restart) of the technology object. During a restart the technology object is reinitialized with the data in load memory. You therefore write changes to the start value in the load memory with the extended instruction "WRIT_DBL" (write to data block in load memory). You trigger the restart in your user program using the Motion Control instruction "MC_Reset" with parameter "Restart" = TRUE. Additional information regarding the restart can be found in the section Restarting technology objects (Page 123).	
Read only (RON) (e.g. <TO>.Position)	The tag cannot and must not be changed during runtime of the user program.	

NOTE**Save changes with "WRIT_DBL"**

Changes to tags immediately in effect are lost on POWER OFF of the CPU, or restart of the technology object.

If changes in the technology data block should also be retained after POWER OFF of the CPU, or restart of the technology object, you must write the changes to the start value in the load memory with the extended instruction "WRIT_DBL".

NOTE**Using the "READ_DBL" and "WRIT_DBL" data block functions**

The "READ_DBL" and "WRIT_DBL" data block functions may only be used on individual tags in conjunction with the tags of the technology object. The "READ_DBL" and "WRIT_DBL" data block functions must not be applied to data structures of the technology object.

Isochronous evaluation of data

If you want to process data of the technology data block in isochronous mode from a Motion Control application cycle, there is the option of evaluating this data in the MC_PreServo /MC_PostServo as of technology version V3.0. As of technology version V5.0, you also have the option of evaluating this in MC_PreInterpolator.

7.2.2 Evaluate StatusWord, ErrorWord and WarningWord (S7-1500, S7-1500T)

To be able to symbolically use individual status and error information from the "StatusWord", "ErrorWord" and "WarningWord" data double words, you can evaluate them as described below. For consistent evaluation, you should avoid using bit addressing to access these data double words in the technology data block. Access to an individual bit in the technology data block only lasts as long as the access to the entire data word.

When required, copy the required data double word to a tag of a data structure and query the individual bits of the tag.

The allocation of the individual bits in the data double words can be found in the description of the tags of the corresponding technology object.

Requirements

The technology object has been created.

Procedure

To evaluate the individual bits in the data word "StatusWord", follow these steps:

1. Create a global data structure. Name the data structure, e.g. as "Status".
2. Create a double word (DWORD) in the data structure "Status". Name the double word, e.g. as "Temp".
3. Create 32 Boolean tags in the "Status" data structure. You can obtain a clearer overview by giving the individual Boolean tags identical names as the bits in the technology DB (e.g. name the fifth Boolean tag "HomingDone").
4. If needed, copy the tag "<TO>.StatusWord" from the technology data block to the double word "Temp" in your data structure.
5. Copy the individual bits of double word "Temp" to the corresponding Boolean tags with bit accesses.
6. Use the Boolean tags to query the status bits.

Evaluate the data words "ErrorWord" and "WarningWord" as specified in steps 1 to 6.

Example

The following example shows how you can read out and save the fifth bit "HomingDone" of the data word "StatusWord":

SCL	Explanation
#Status.Temp := "TO".StatusWord;	//Copy status word
#Status.HomingDone := #Status.Temp.%X5;	//Copy individual bits per bit access

STL	Explanation
L "TO".StatusWord	//Copy status word
T #Status.Temp	
U #Status.Temp.%X5	//Copy individual bits per bit access
= #Status.HomingDone	

7.2.3 Change restart-relevant data (S7-1500, S7-1500T)

In order to change restart-relevant data in the technology data block, write to the starting values of the tags in load memory using the extended instruction "WRIT_DBL". In order for the changes to be applied, a restart of the technology object must be performed.

You can see from the description of the tag of the corresponding technology object whether value changes of a tag are restart-relevant.

Requirement

The technology object has been created.

Procedure

To change restart-relevant data, proceed as follows:

1. Create a data block and fill it with the restart-relevant values, that you want to change in the technology data block. In doing so, the data types must match the tags to be changed.
2. Write the values from your data block to the starting values of the tags of the technology data block in load memory, using the extended instruction "WRIT_DBL".
If restart-relevant data were changed, this will be indicated in the "<TO>.StatusWord.X3 (OnlineStartValuesChanged)" tag of the technology object.
3. Perform a restart of the technology object using the Motion Control instruction "MC_Reset" with parameter "Restart" = TRUE.

After the restart of the technology object, the new value is transferred into the technology data block in work memory, and is effective.

Changing multiple variables with the "WRIT_DBL" instruction

The tag "<TO>.StatusWord.X3 (OnlineStartValuesChanged)" indicates that the value of a tag has been changed.

If you change several variables with "WRIT_DBL", you cannot use "<TO>.StatusWord.X3 (OnlineStartValuesChanged)" to determine which tags have already been written.

To capture the relevant value changes of several variables, build your user program with "WRIT_DBL" similar to the following programming example in a sequencer:

SCL programming example

```
//*****
#FB_STATE_LIMITS_MAX_JERK: //write limits max jerk
#tempRetVal := WRIT_DBL (REQ := TRUE,
SRCBLK := #statDynamicLimits.MaxJerk,
BUSY => #statBusyWriting,
DSTBLK => #axis.DynamicLimits.MaxJerk);
IF #statBusyWriting THEN
; //wait until writing completed
ELSIF #tempRetVal = 16#0 AND NOT #statBusyWriting
THEN //done without error
//go to next state:
//restart technology object, if all data are
//written to load memory
//Alternatively: Write more variables in
//the next state via WRIT_DBL and
//execute restart later
```

```

#statFBState := #FB_STATE_RESTART_TO;
ELSE //error Write DBL
//go to error state
#statStatus := INT_TO_WORD (#tempRetVal);
END_IF;
//*****
#FB_STATE_RESTART_TO: //Restart TO
#instFBReset (Axis := #axis,
Execute := TRUE,
Restart := TRUE);

```

7.3 Motion Control instructions (S7-1500, S7-1500T)

7.3.1 Motion Control instruction parameters (S7-1500, S7-1500T)

The individual Motion Control instructions are described in detail in the section "S7-1500 Motion Control V7".

When creating your user program, take the following explanations of the Motion Control instruction parameters into account.

Reference to the technology object

The technology object is specified for the Motion Control instruction as follows:

- **Parameter "Axis"**

The "Axis" input parameter of a Motion Control instruction contains a reference to the technology object that is to execute the corresponding job.

The corresponding technology object is also referenced in the following parameters:

- Parameter "Master"
- Parameter "Slave"
- Parameter "Cam"
- Parameter "MeasuringInput"
- Parameter "OutputCam"
- Parameter "CamTrack"
- "AxesGroup" parameter

As of technology version V3.0, the reference to the technology object can be specified, also in limited manner, via the data type "DB_ANY".

As of firmware version V3.0, you can temporarily home the technology object with "DB_ANY".

For more information on programming with "DB_ANY", refer to the section "Parameter transfer for function blocks ([Page 105](#))".

Job start and transfer of input parameters of a Motion Control instruction

For the start of jobs and the transfer of modified parameter values, a distinction is made between the following Motion Control instructions:

- **Motion Control instructions with "Execute" parameter**
 With a positive edge at the "Execute" parameter, the job is started and the existing values for the input parameters are transferred.
 Subsequently changed parameter values are not transferred until the next job start.
 Resetting the "Execute" parameter does not end the job, but it does affect the display duration of the job status. As long as "Execute" is set to "TRUE", the output parameters will be updated. If "Execute" is reset before the completion of a job, the parameters "Done", "Error" and "CommandAborted" are correspondingly set for only one call cycle.
- **Motion Control instructions with "Enable" parameter**
 The job is started by setting the "Enable" parameter.
 As long as "Enable" = TRUE, the job remains active and changed parameter values will be transferred each time the instruction is called in the user program.
 The job is ended by resetting the "Enable" parameter.
 The input parameters "JogForward" and "JogBackward" of the Motion Control instruction "MC_MoveJog" correspond in their behavior to the "Enable" parameter.

Job status

The following output parameters indicate the status of the job execution:

- **Motion Control instructions with "Done" parameter**
 The normal completion of a job is indicated with parameter "Done" = TRUE.
- **Motion Control instructions without "Done" parameter**
 The achievement of the job objective is indicated by other parameters (e.g. "Status", "InVelocity"). For more information, refer to the "Tracking running jobs ([Page 110](#))" section.
- **Parameter "Busy"**
 As long as a job is in progress, the "Busy" parameter shows the value "TRUE". If a job was ended or canceled, "Busy" shows the value "FALSE".
- **Parameter "Active"**
 If a job is active in Motion Control, the parameter "Active" shows the value "TRUE". As long as a job is in the job sequence, "Active" shows the value "FALSE".
- **Parameter "CommandAborted"**
 If a job was canceled by another job, the "CommandAborted" parameter shows the value "TRUE".
- **Parameter "Error"**
 If a Motion Control instruction error occurs, the "Error" parameter shows the value "TRUE". The "ErrorID" parameter indicates the cause of the error.

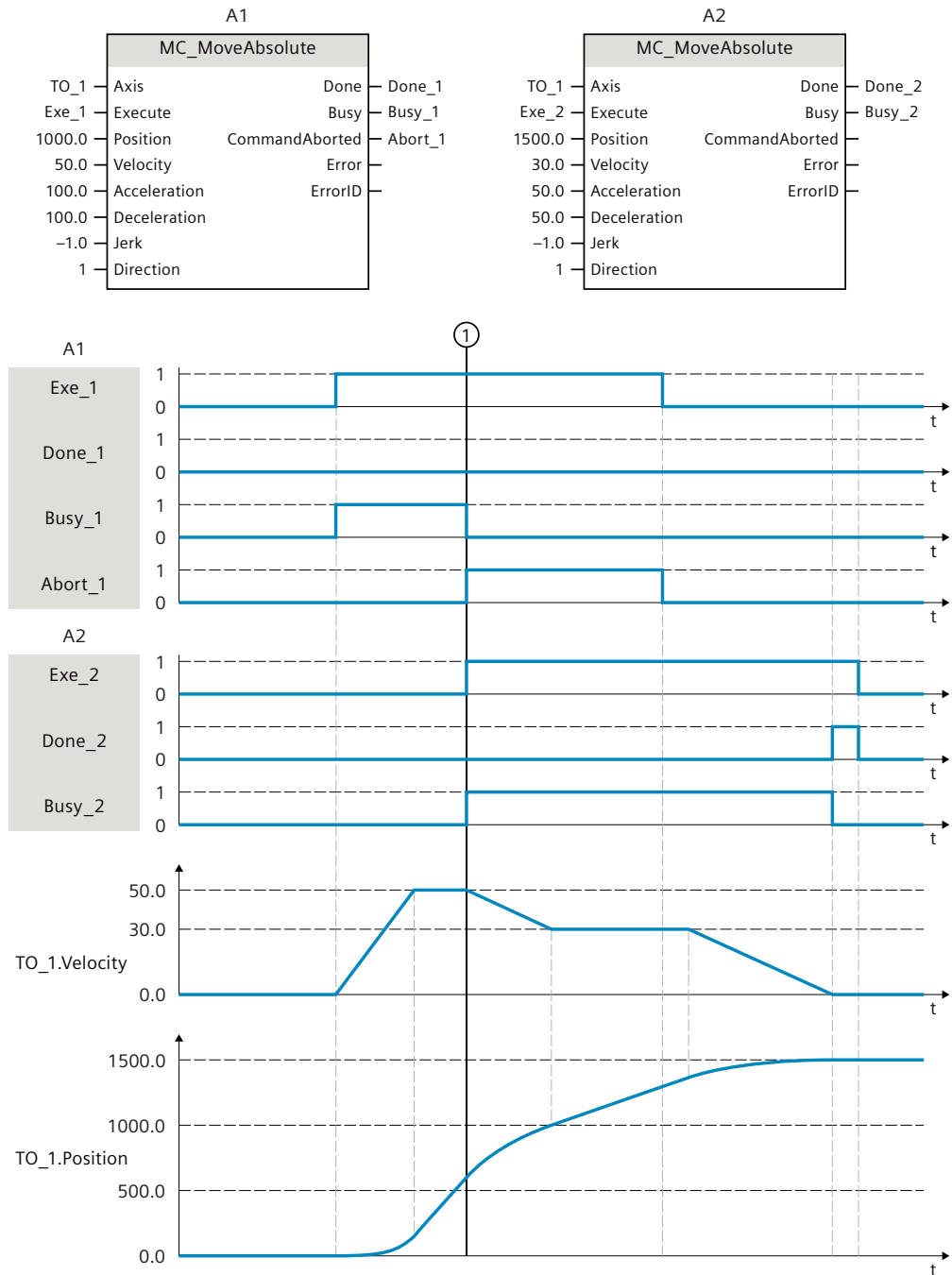
As long as the "Execute" or "Enable" parameter is set to "TRUE", the output parameters will be updated. Otherwise, the parameters "Done", "Error" and "CommandAborted" are correspondingly set for only one cycle.

Cancelation of running jobs

An active Motion Control job is canceled by the start of a new Motion Control job. In the process, the current dynamic setpoints (acceleration, deceleration, jerk, velocity) are set to the values of the overriding job.

Example of parameter behavior

The behavior of the parameters of Motion Control instructions is shown in the following chart using the example of two "MC_MoveAbsolute" jobs.



Using "Exe_1", an "MC_MoveAbsolute" job (A1) with target position 1000.0 is initiated. "Busy_1" is set to "TRUE". The axis is accelerated to the specified velocity and moved to the target position (see TO_1.Velocity and TO_1.Position). Before the target position is reached, the job is overridden at time ① by another "MC_MoveAbsolute" job (A2). The termination is signaled via "Abort_1", and "Busy_1" is set to "FALSE". The axis is braked to the velocity specified in A2 and moved to the new target position 1500.0. When the axis reaches the target position, this is signaled via "Done_2".

Non-position-controlled operation

The position control of the axis can be deactivated with the following parameters:

- MC_Power.StartMode = 0
- MC_MoveVelocity.PositionControlled = FALSE
- MC_MoveJog.PositionControlled = FALSE

For more information, refer to the section "Non-position-controlled operation" of the "S7-1500/S7-1500T Axis functions" ([Page 8](#)) documentation.

7.3.2 Add Motion Control instructions (S7-1500, S7-1500T)

You add Motion Control instructions to a program block in the same way as other instructions. You control all available functions of the technology object using the Motion Control instructions. The instructions can be called at all execution levels.

Requirements

The technology object was created.

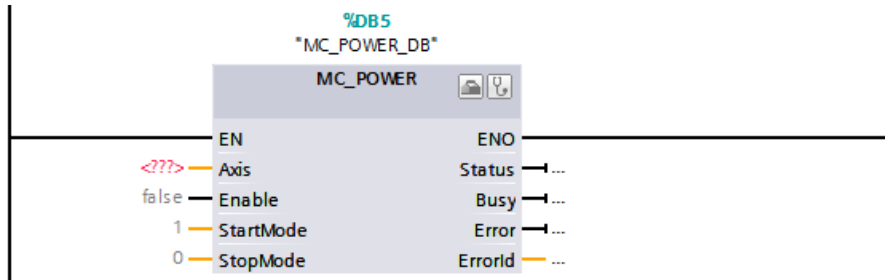
Procedure

To add Motion Control instructions in your user program, proceed as follows:

1. Double click your program block in the project tree (the program block must be called in the cyclical program).
The program block is opened in the programming editor, and the available instructions are displayed.
2. In the "Instructions" task card, open the "Technology > Motion Control" folder.
3. Using drag-and-drop, move the Motion Control instruction, e.g. "MC_Power", to the desired segment of the program block.
The "Call options" dialog opens.
4. In the dialog, specify a name and a number for the instance data block of the Motion Control instruction.

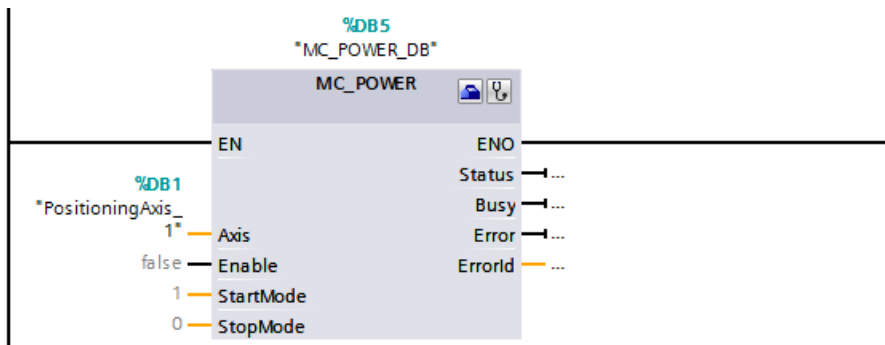
- Click "OK".

The Motion Control instruction "MC_Power" is inserted into the network.





The instance data block is automatically created under "Program Blocks > System Blocks > Program Resources".

- Input parameters without a default value (e.g. "Axis"), must be assigned. Select the technology object in the project tree and move the technology object onto <...> at the "Axis" parameter using drag-and-drop.



Once the technology object is specified in the "Axis" parameter, the following buttons are available to you:

	To open the configuration of the technology object, click on the toolbox icon.
	To open the diagnostics of the technology object, click on the stethoscope icon.

- Add additional Motion Control instructions in accordance with steps 3 through 6.

NOTE

Multi-instance DBs

If you use multi-instances of the Motion Control instructions "MC_Power" or "MC_TorqueLimiting", create the multi-instances in a separate instance data block. This allows you to download program blocks from other sections of your user program without switching off the axes, including in "RUN" mode.

See also

[Tracking active jobs \(Page 110\)](#)

7.3.3 Parameter transfer for function blocks (S7-1500, S7-1500T)

If you want to reuse a function block with Motion Control instructions for different technology objects, create an input parameter of the data type of the respective technology object in the block interface of the calling function block. You assign the data type in the block interface with a direct input. The parameter is then transferred as reference to the technology object to the "Axis" parameter of the Motion Control instructions. The data types of technology objects correspond to the structure of the associated technology data block.

In contrast to standard data types, the data types for technology objects are always passed on as pointers to the function block (Call by reference). This remains true if you declare the data types of the technology objects in "Input" area of the block interface. Write access to function blocks always leads directly to modification of the referenced technology object.

Example 1: Tag transfer with specific data type

By specifying the data type, you can address the tags of the technology object in the function block (<parameters of the block interface>.<tag of the technology object>). The data types for the reference to the technology objects are available in the section "appendix [\(Page 124\)](#)".

The following table shows the declaration of the tags used:

Tag	Declaration	Data type	Description
axis	Input	TO_PositioningAxis	Reference to the technology object
on	Input	BOOL	Signal to enable the axis
actPosition	Output	LReal	Query of the actual position from the technology data block
instMC_POWER	Static	MC_POWER	Multi-instance of the Motion Control instruction "MC_Power"

The following SCL program shows the tag transfer with a specific data type:

SCL	Explanation
#instMC_POWER(Axis := #axis, Enable := #on);	//Call of the Motion Control instruction "MC_Power" with enable of the axis
#actPosition := #axis.ActualPosition;	//Query of the actual position from the technology data block

Example 2: Tag transfer with "DB_ANY" for axes

A more flexible option for the transferring technology objects to blocks is provided by the data type "DB_ANY" in combination with references to technology objects. These references can be assigned during runtime.

You can store all types of technology objects in a "DB_ANY" data type ARRAY. An ARRAY of the data type "DB_ANY" can thus represent a list of axes, for example. This allows technology objects to be handled more flexibly in a user program.

This example shows how to enable and disable different axis types with one function block. The technology objects are homed temporarily.

The following technology objects are used:

Number	Data type	Name
1	TO_SpeedAxis	SpeedAxis_1
1	TO_PositioningAxis	PositioningAxis_1
1	TO_SynchronousAxis	SynchronousAxis_1

To enable and disable all three axes with one function block, follow these steps:

1. Add a global data block, in the example "AxesDB".
2. Define the one constant as the last index of the array, e.g. LAST_INDEX_AXES. The constant has the value 2 in this example because 3 axes are used and the array starts at 0.
3. Add an "Array[0.. LAST_INDEX_AXES] of DB_ANY" to the created "AxesDB" data block:

Tag	Declaration	Data type
axes	Static	Array[0.. LAST_INDEX_AXES] of DB_ANY

The unique assignment of the technology objects to the elements of the array must be carried out once in the user program, for example, when the CPU is started in OB100 (Startup).

4. Assign the created technology objects to the elements of the array from the data block in OB100 (Startup).

The following SCL program shows the tag transfer with "DB_ANY":

```
//assign technology objects to ARRAY of DB_ANY
"Data".axes[0] := "SpeedAxis_1";
"Data".axes[1] := "PositioningAxis_1";
"Data".axes[2] := "SynchronousAxis_1";
```

5. Create a function block, in the example "EnableAxes".
6. Declare the tags as follows:

Tag	Declaration	Data type	Description
enableAxes	Input	BOOL	Enable/disable all axes
axisEnabled	Output	ARRAY [0.. LAST_INDEX_AXES] of BOOL	Status axis enable
axis	InOut	ARRAY[0.. LAST_INDEX_AXES] of DB_ANY	List of the axes

Tag	Declaration	Data type	Description
instMC_Power	Static	ARRAY [0.. LAST_INDEX_AXES] of MC_POWER	Array of multi-instances of the MC_POWER
tempRefSpeedAxis	Temp	REF_TO TO_SpeedAxis	Temporary reference of the technology object type TO_SpeedAxis
tempAxesCounter	Temp	Dint	Counter tag for axes

In the program code, reference the three axis types to the temporary reference of the TO_SpeedAxis technology object. Since the TO_SpeedAxis is a component of the TO_PositioningAxis and the TO_SynchronousAxis, referencing of all three axes is possible.

7. Program the "EnableAxis" FB as follows:

```
// enable all axes
FOR #tempAxesCounter := 0 TO LAST_INDEX_AXES DO
  // build temporary reference
  #tempRefSpeedAxis := #axes[#tempAxesCounter];
  // check valid reference
  IF #tempRefSpeedAxis <> NULL THEN
  // call multi instance
    #instMC_Power[#tempAxesCounter] (Axis :=
#tempRefSpeedAxis^,
                                     Enable := #enableAxes);
  // set output states, if axis is enabled
  #axisEnabled[#tempAxesCounter] :=
#tempRefSpeedAxis^.StatusWord.%X0;
  ELSE
    ; // implement error handling in case of NULL
  END_IF;
END_FOR;
```

Example 3: Tag transfer with "DB_ANY" for cyclic cams

This example shows how to temporarily reference the different technology object types for cyclic cams for cam interpolation. You can also implement the switching of cyclic cams during camming with temporary references.

The following technology objects are used:

Number	Data type	Name
2	TO_Cam	TO_Cam_1, TO_Cam_2
2	TO_Cam_10k	TO_Cam_3_10k_1, TO_Cam_4_10k

1. Proceed as in the first five steps of example 2. Cyclic cams are used instead of axes.
2. Declare the tags of the "InterpolateCams" function block as follows:

Tag	Declaration	Data type	Description
executeCamInterpolation	Input	BOOL	Start cam interpolation
selectedCamIndex	Input	USInt	Cyclic cam index
done	Output	BOOL	Cyclic cam was interpolated

Tag	Declaration	Data type	Description
busy	Output	BOOL	Cyclic cam is being interpolated
error	Output	BOOL	Error on the block
cams	InOut	ARRAY[0.. LAST_INDEX_CAMS] of DB_ANY	List of cyclic cams
instInterpolateCam	Static	MC_INTERPOLATECAM	Multi-instance of the MC_INTERPOLATECAM
tempRefCam	Temp	REF_TO TO_CamBase	Temporary reference of the technology object type TO_CamBase
tempAxesCounter	Temp	Dint	Counter tag for axes

In the program code, reference the cyclic cams to the temporary reference of the TO_CamBase technology object. A reference is possible because the TO_CamBase is a component of the TO_Cam and the TO_Cam_10k.

You can use the "selectedCam" input to access the index of the array and thus select the cyclic cam that is to be interpolated. Start the cam interpolation via the "executeCamming" input.

3. Program the "InterpolateCams" FB as follows.

```
//execute cam interpolation for TO_Cam and TO_Cam_10k
#tempRef_Cam := #cams[#selectedCamIndex];
// check valid reference
IF #tempRefCam <> NULL THEN
    // call multi instance for cam interpolation
    #instInterpolateCam(Cam := #tempRefCam^,
        Execute:=#executeCamInterpolation,
        Done=>#done,
        Busy=>#busy,
        Error=>#error);
ELSE
    // implement error handling here,
    // DB_ANY does not have a type TO_Cam or TO_Cam_10k
    #busy := FALSE;
    #done := FALSE;
    #error := TRUE;
END_IF;
```

7.4 Starting Motion Control jobs (S7-1500, S7-1500T)

Motion Control jobs are started by setting the "Execute" or "Enable" parameter of the Motion Control instruction. The call of the Motion Control instructions for a technology object should occur in an execution level.

When executing Motion Control jobs, you should also take note of the status of the technology object.

Starting Motion Control jobs should be performed in the following steps:

1. Query the status of the technology object.
2. Initiate new job for the technology object.
3. Check job status.

These steps are explained using the example of a job for absolute positioning.

1. Query the status of the technology object

Make sure that the technology object is in the appropriate status to perform the desired job:

- **Has the technology object been released?**

To execute motion jobs, the technology object must be enabled.

Enabling is performed using the Motion Control instruction "MC_Power".

The "MC_Power.Status" parameter (<TO>.StatusWord.X0 (Enable)) must show the value "TRUE".

- **Is a technology alarm pending?**

To perform motion commands, no technology alarms or alarm responses may be pending.

The "<TO>.ErrorDetail.Number" and "<TO>.ErrorDetail.Reaction" tags of the technology object must show the value zero. After resolving the error, acknowledge any pending alarms using the Motion Control instruction "MC_Reset".

You can find a list of the technology alarms and alarm responses in the "Technology alarms" section of the "S7-1500/S7-1500T Motion Control alarms and error IDs" documentation ([Page 8](#)).

- **Has the technology object been homed?**

In order to perform a job for absolute positioning, the positioning axis/synchronous axis technology object must be homed. The referencing occurs via the Motion Control instruction "MC_Home". The "<TO>.StatusWord.X5 (HomingDone)" tag of the technology object must show the value "TRUE".

2. Initiate new command for the technology object

At the "Position" parameter of a Motion Control instruction, e.g. "MC_MoveAbsolute", specify the position to which the axis should be moved. Start the job with a positive edge at the "Execute" parameter.

3. Check command status

Parameter "Done" of the Motion Control instruction indicates successful completion of a job (target reached, in this case).

If an error is detected, the "Error" parameter of the Motion Control instruction is set to "TRUE", and the job is rejected.

You can program an error handling routine for the Motion Control job. For this purpose, evaluate the error indicated in the "Error" parameter. The cause of the error is indicated in the ErrorID parameter. After resolving the cause of the error, restart the job.

If "Error" = TRUE and "ErrorID" = 16#8001 is indicated during job execution, a technology alarm has occurred.

You can find a list of the ErrorIDs in the "Error IDs on Motion Control instructions" section of the "S7-1500/S7-1500T Motion Control alarms and error IDs" documentation ([Page 8](#)).

More information

An option for evaluating the individual status-, error- and warning bits find can be found in section "Evaluate StatusWord, ErrorWord and WarningWord ([Page 98](#))".

7.5 Tracking active jobs (S7-1500, S7-1500T)

The current status of the job processing is made available via the output parameters of the Motion Control instruction. These parameters are updated with each call of the Motion Control instruction.

When tracking jobs, a distinction is made between three groups:

- Motion Control instructions with "Done" parameter ([Page 110](#))
- Motion Control instructions without "Done" parameter ([Page 115](#))
- Motion Control instruction "MC_MoveJog" ([Page 119](#))

7.5.1 Motion Control instructions with "Done" parameter (S7-1500, S7-1500T)

Jobs of Motion Control instructions with the "Done" parameter are started with a positive edge at the "Execute" parameter. If the job was completed without errors or interruption by another job (e.g. "MC_MoveAbsolute": Target position reached), the "Done" parameter shows the value "TRUE". The CPU processes the Motion Control statements and the technology object in different execution levels. The status bits on the Motion Control statements, such as the "Done" parameter, are updated with a delay.

In positioning instructions, the parameter "Done" is delayed by the set minimum dwell time (<TO>.PositioningMonitoring.MinDwellTime).

The following Motion Control instructions have a "Done" parameter for the S7-1500 CPU:

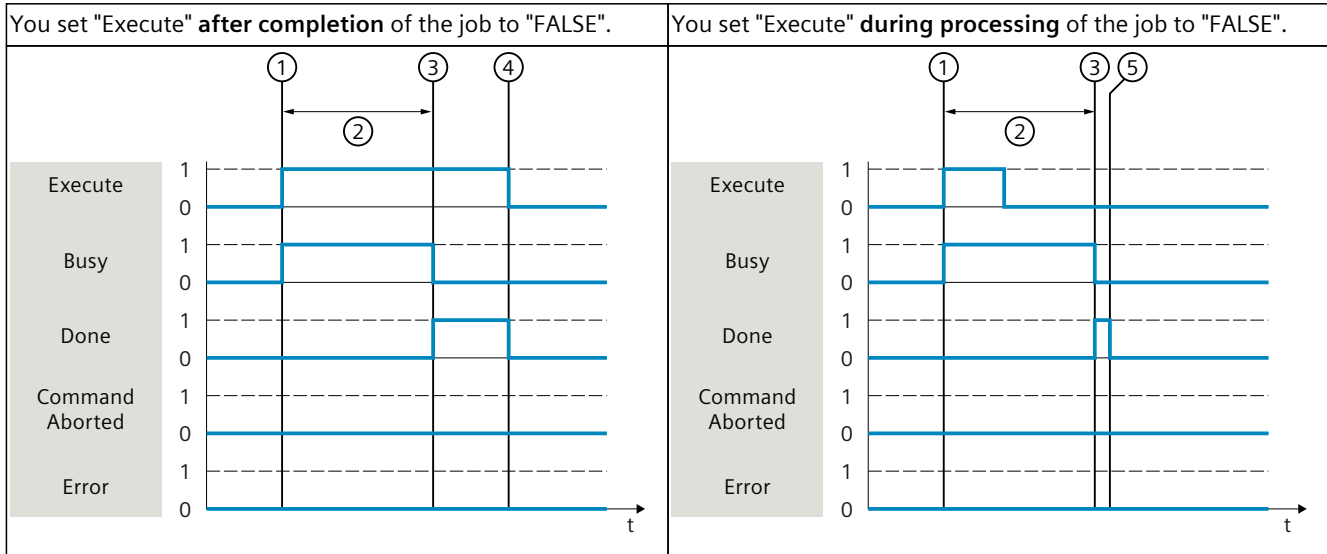
- MC_Reset
- MC_Home
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveSuperimposed
- MC_HaltSuperimposed
- MC_SetSensor (S7-1500T)

- MC_Stop
- MC_SetAxisSTW
- MC_WriteParameter
- MC_SaveAbsoluteEncoderData
- MC_MeasuringInput
- MC_AbortMeasuringInput
- MC_PhasingRelative (S7-1500T)
- MC_PhasingAbsolute (S7-1500T)
- MC_OffsetRelative (S7-1500T)
- MC_OffsetAbsolute (S7-1500T)
- MC_GearOut (S7-1500T)
- MC_CamOut (S7-1500T)
- MC_InterpolateCam (S7-1500T)
- MC_GetCamLeadingValue (S7-1500T)
- MC_GetCamFollowingValue (S7-1500T)
- MC_CopyCamData (S7-1500T)
- MC_GroupInterrupt (S7-1500T)
- MC_GroupContinue (S7-1500T)
- MC_GroupStop (S7-1500T)
- MC_MoveLinearAbsolute (S7-1500T)
- MC_MoveLinearRelative (S7-1500T)
- MC_MoveCircularAbsolute (S7-1500T)
- MC_MoveCircularRelative (S7-1500T)
- MC_MoveDirectAbsolute (S7-1500T)
- MC_MoveDirectRelative (S7-1500T)
- MC_TrackConveyorBelt (S7-1500T)
- MC_KinematicsMotionSimulation (S7-1500T)
- MC_DefineWorkspaceZone (S7-1500T)
- MC_DefineKinematicsZone (S7-1500T)
- MC_SetWorkspaceZoneActive (S7-1500T)
- MC_SetWorkspaceZoneInactive (S7-1500T)
- MC_SetKinematicsZoneActive (S7-1500T)
- MC_SetKinematicsZoneInactive (S7-1500T)
- MC_DefineTool (S7-1500T)
- MC_SetTool (S7-1500T)
- MC_SetOcsFrame (S7-1500T)
- MC_LoadProgram (S7-1500T)
- MC_RunProgram (S7-1500T)
- MC_StopProgram (S7-1500T)

The behavior of the parameters is shown below by way of example for various situations.

Complete execution of the job

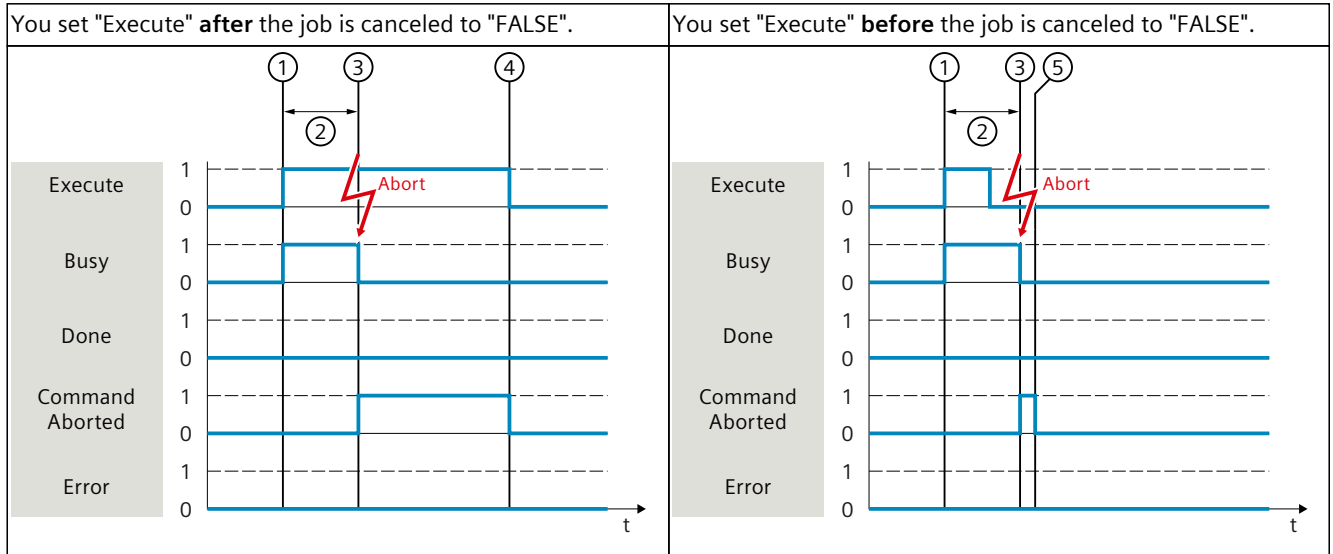
If the Motion Control job has been completely executed all the way to the end, this is indicated with parameter "Done" = "TRUE". The signal state of the "Execute" parameter influences the display duration for the "Done" parameter:



- | | |
|---|---|
| ① | The job is started with a positive edge at the "Execute" parameter. Depending on the programming, "Execute" can be reset to the value "FALSE" during the job or the value "TRUE" can be retained until after completion of the job. |
| ② | While the job is being executed, the "Busy" parameter shows the value "TRUE". |
| ③ | At the completion of the job (for example, with Motion Control instruction "MC_MoveAbsolute": Target position reached), the "Busy" parameter changes to "FALSE" and the "Done" parameter to "TRUE". |
| ④ | As long as the "Execute" parameter retains the value "TRUE" after completion of the job, the "Done" parameter also retains the value "TRUE". |
| ⑤ | If the "Execute" parameter was already set to "FALSE" before completion of the job, the "Done" parameter shows the value "TRUE" for only one execution cycle. |

Job cancellation

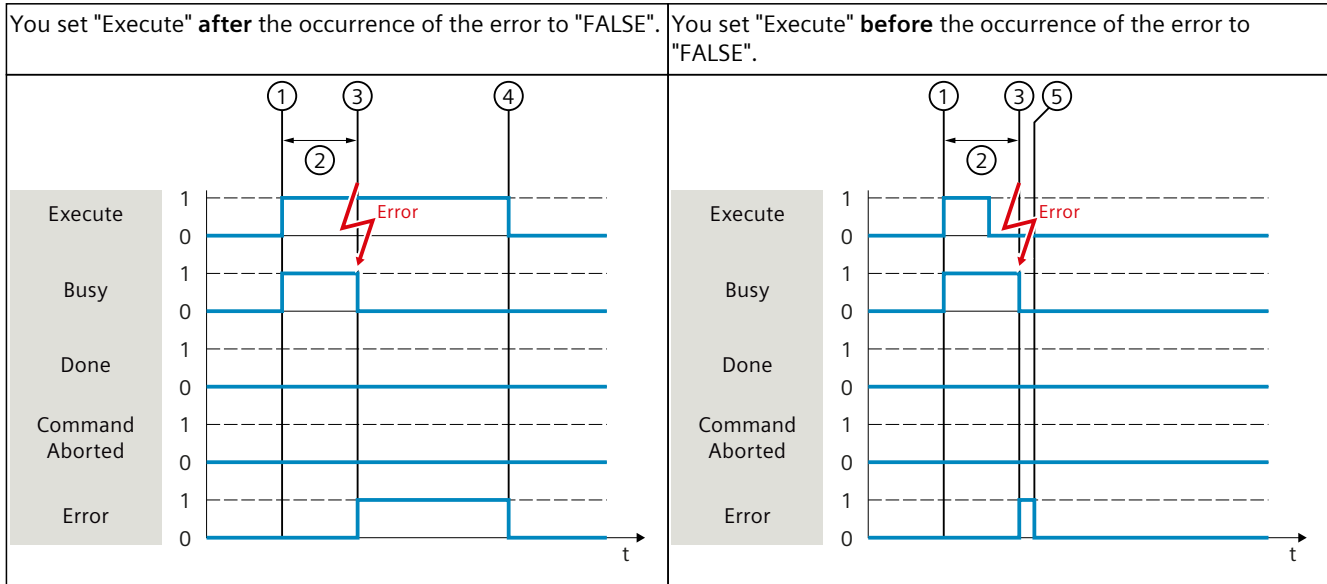
If the Motion Control job is canceled during processing by another job, this is indicated in the "CommandAborted" parameter with the value "TRUE". The signal state of the "Execute" parameter influences the display duration for the "CommandAborted" parameter:



- | | |
|---|---|
| ① | The job is started with a positive edge at the "Execute" parameter. Depending on the programming, "Execute" can be reset to the value "FALSE" during the job or the value "TRUE" can be retained until after completion of the job. |
| ② | While the job is being executed, the parameter "Busy" shows the value "TRUE". |
| ③ | During job execution, the job is canceled by another Motion Control job. When the job is canceled, the "Busy" parameter changes to "FALSE" and "CommandAborted" changes to "TRUE". |
| ④ | As long as the "Execute" parameter retains the value "TRUE" after completion of the job, the "CommandAborted" parameter also retains the value "TRUE". |
| ⑤ | If the "Execute" parameter was already set to "FALSE" before the job is canceled, the "CommandAborted" parameter shows the value "TRUE" for only one execution cycle. |

Error during job execution

If an error occurs during execution of the Motion Control job, this is indicated with parameter "Error" = "TRUE". The signal state of the "Execute" parameter influences the display duration for the "Error" parameter:



- | | |
|---|---|
| ① | The job is started with a positive edge at the "Execute" parameter. Depending on the programming, "Execute" can be reset to the value "FALSE" during the job or the value "TRUE" can be retained until after completion of the job. |
| ② | While the job is being executed, the "Busy" parameter shows the value "TRUE". |
| ③ | An error occurs during the execution of the job. When the error occurs, the "Busy" parameter changes to "FALSE" and the "Error" parameter to "TRUE". |
| ④ | As long as the "Execute" parameter retains the value "TRUE" after the occurrence of the error, the "Error" parameter also retains the value "TRUE". |
| ⑤ | If the "Execute" parameter was already set to "FALSE" before the occurrence of the error, the "Error" parameter shows the value "TRUE" for only one execution cycle. |

Motion Control instructions

You can find more information on the Motion Control instructions of the Kinematics technology object in the "S7-1500T Kinematics functions" documentation ([Page 8](#)).

7.5.2 Motion Control instructions without "Done" parameter (S7-1500, S7-1500T)

Motion control instructions without the "Done" parameter use a special parameter to indicate that the job objective (e.g. "InVelocity", "InGear") has been achieved. The target state or motion is held until the job is aborted or an error occurs.

The following motion control instructions have a special parameter for indicating the job status:

Motion control instruction	Parameter	S7-1500	S7-1500T
MC_Power	Status	✓	✓
MC_MoveVelocity	InVelocity	✓	✓
MC_MoveJog	InVelocity	✓	✓
MC_GearIn	InGear	✓	✓
MC_GearInPos	InSync	-	✓
MC_GearInVelocity	InSync	-	✓
MC_CamIn	InSync	-	✓
MC_SynchronizedMotionSimulation	InSimulation	-	✓
MC_LeadingValueAdditive	Busy	-	✓
MC_MotionInVelocity	Busy	-	✓
MC_MotionInPosition	Busy	-	✓
MC_MotionInSuperimposed	Busy	-	✓
MC_TorqueLimiting	InClamping and InLimitation	✓	✓
MC_KinematicsTransformation	Busy and Valid	-	✓
MC_InverseKinematicsTransformation	Busy and Valid	-	✓

The following motion control instructions do not have any special parameter for indicating the job status. Feedback is provided via the following tags:

Motion control instruction	Parameter	Description
MC_MeasuringInputCyclic	Busy	The execution of a measuring job is indicated with parameter "Busy" = "TRUE". Completed measuring events are indicated in the corresponding event counters "<TO>.MeasuredValues.MeasuredValue1Counter" and "<TO>.MeasuredValues.MeasuredValue2Counter" of the technology data block.
MC_OutputCam	Busy	The execution of a job is indicated with parameter "Busy" = "TRUE". The CamOutput tag in the associated technology data block indicates the switching state of the output cam.
MC_CamTrack	Busy	The execution of a job is indicated with parameter "Busy" = "TRUE". The TrackOutput tag in the associated technology data block indicates the switching state of the cam track.

The behavior of the parameter is shown for various situations using the motion control instruction "MC_MoveVelocity" as an example:

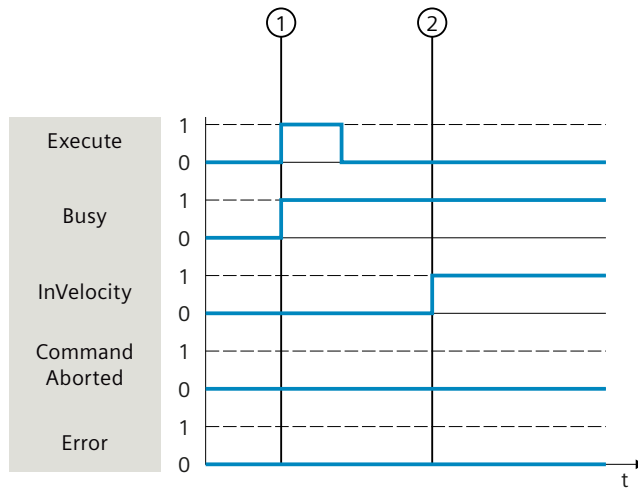
Example "MC_MoveVelocity"

An "MC_MoveVelocity" job is started with a positive edge at the "Execute" parameter. The job objective is fulfilled when the assigned velocity is reached and the axis travels at constant velocity. When the assigned velocity is reached and maintained, this is indicated in the "InVelocity" parameter with the value "TRUE".

The motion of the axis can, for example, be stopped with an "MC_Halt" job.

The assigned velocity is reached and maintained

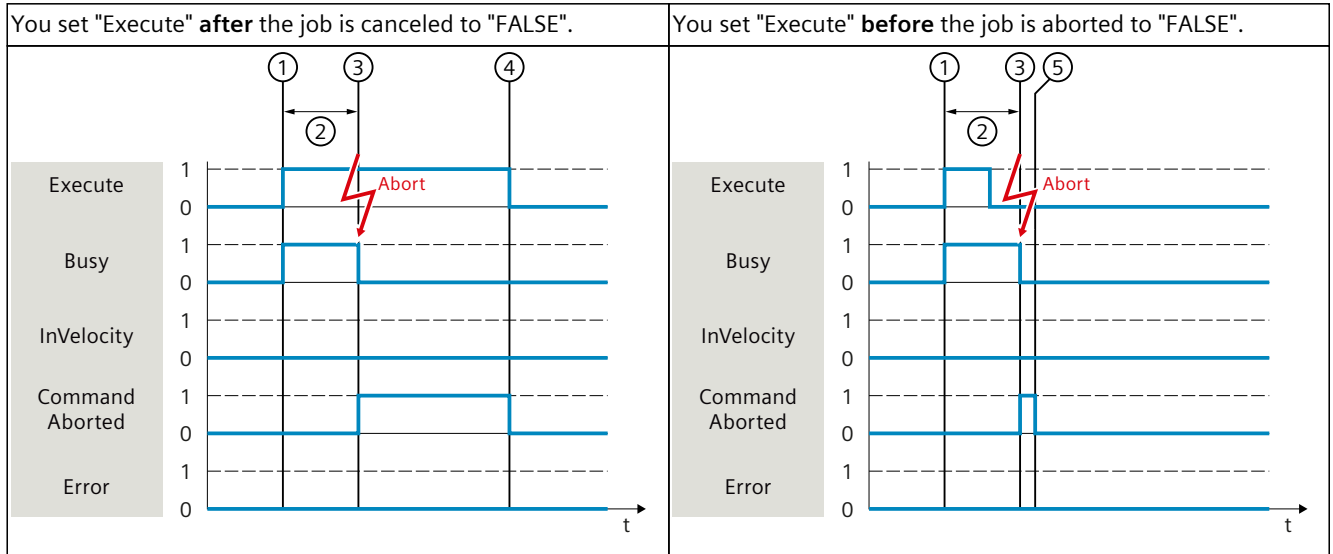
Reaching of the assigned velocity is indicated with parameter "InVelocity" = "TRUE". The "Execute" parameter has no effect on the display duration for the "InVelocity" parameter.



①	The job is started with a positive edge at the "Execute" parameter. Depending on the programming, "Execute" can be reset to the "FALSE" value before or after the parameterized velocity has been reached. While the job is being executed, the parameter "Busy" shows the value "TRUE".
②	When the assigned velocity is reached, the "InVelocity" parameter changes to "TRUE". The "Busy" and "InVelocity" parameters retain the value "TRUE" until another motion control job overrides the "MC_MoveVelocity" job.

The job is aborted before the assigned velocity is reached

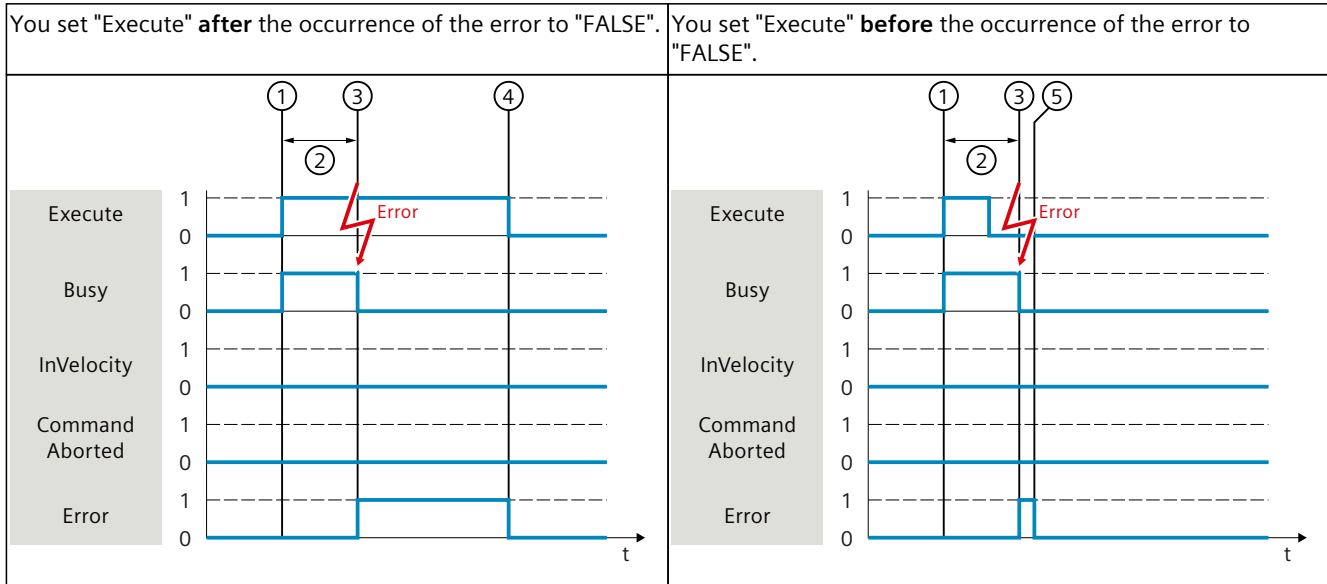
If the motion control job is canceled by another job before the assigned velocity is reached, this is indicated with parameter "CommandAborted" = "TRUE". The signal state of the "Execute" parameter influences the display duration for the "CommandAborted" parameter.



- | | |
|---|--|
| ① | The job is started with a positive edge at the "Execute" parameter. Depending on the programming, "Execute" can be reset to the value "FALSE" during the job or the value "TRUE" can be retained until after the job is aborted. |
| ② | While the job is being executed, the "Busy" parameter shows the value "TRUE". |
| ③ | During job execution, the job is aborted by another motion control job. When the job is aborted, the "Busy" parameter changes to "FALSE" and "CommandAborted" changes to "TRUE". |
| ④ | As long as the "Execute" parameter retains the value "TRUE" after completion of the job, the "CommandAborted" parameter also retains the value "TRUE". |
| ⑤ | If the "Execute" parameter was already set to "FALSE" before the job is aborted, the "CommandAborted" parameter shows the value "TRUE" for only one execution cycle. |

An error has occurred prior to reaching the assigned velocity

If an error occurs during execution of the motion control job before the assigned velocity has been reached, this is indicated with parameter "Error" = "TRUE". The signal state of the "Execute" parameter influences the display duration for the "Error" parameter.



- | | |
|---|--|
| ① | The job is started with a positive edge at the "Execute" parameter. Depending on the programming, "Execute" can be reset to the value "FALSE" during the job or the value "TRUE" can be retained until after the error has occurred. |
| ② | While the job is being executed, the "Busy" parameter shows the value "TRUE". |
| ③ | An error occurs during the execution of the job. When the error occurs, the "Busy" parameter changes to "FALSE", and the "Error" parameter to "TRUE". |
| ④ | As long as the "Execute" parameter retains the value "TRUE" after completion of the job, the "Error" parameter also retains the value "TRUE". |
| ⑤ | If the "Execute" parameter was already set to "FALSE" before the job is aborted, the "Error" parameter shows the value "TRUE" for only one execution cycle. |

Motion control instructions

You can find more information on the motion control instructions of the Kinematics technology object in the "S7-1500T Kinematics functions" documentation ([Page 8](#)).

7.5.3 Motion Control instruction "MC_MoveJog" (S7-1500, S7-1500T)

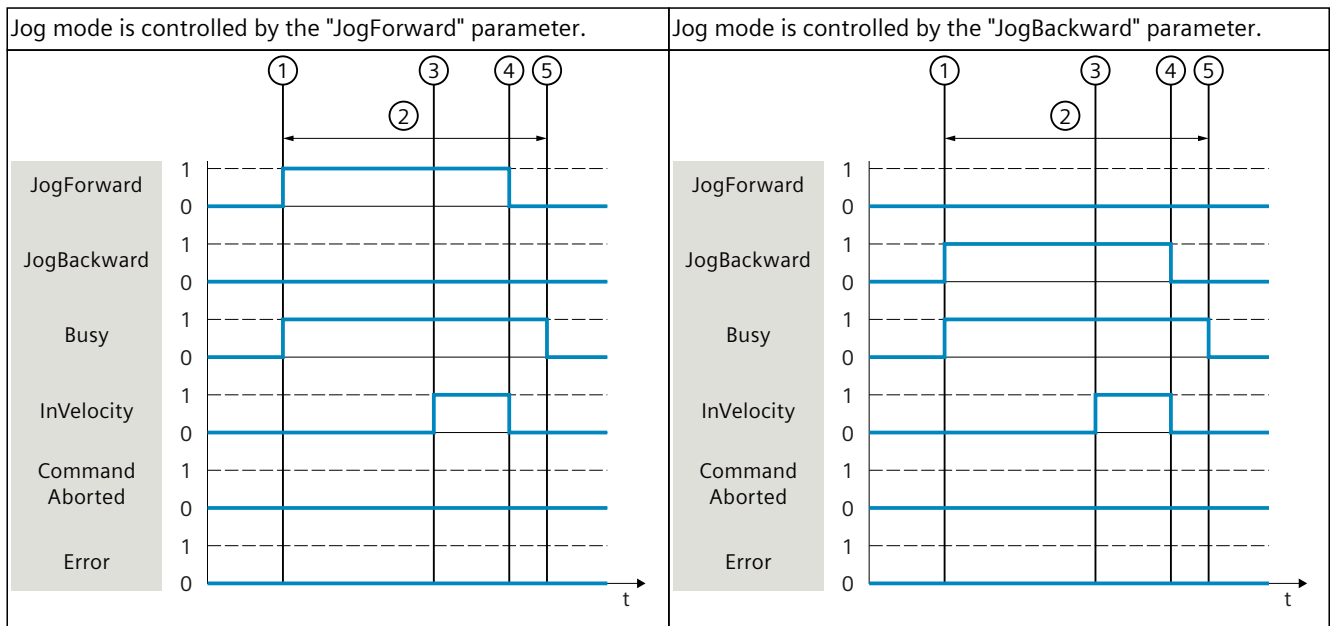
An "MC_MoveJog" job is started by setting the "JogForward" or "JogBackward" parameter. The job objective is fulfilled when the assigned velocity is reached and the axis travels at constant velocity. When the assigned velocity is reached and maintained, this is indicated in the "InVelocity" parameter with the value "TRUE".

The job is complete when the "JogForward" or "JogBackward" parameter has been set to the value "FALSE" and the axis has come to a standstill.

The behavior of the parameters in various situations is shown below by way of example.

The assigned velocity is reached and maintained

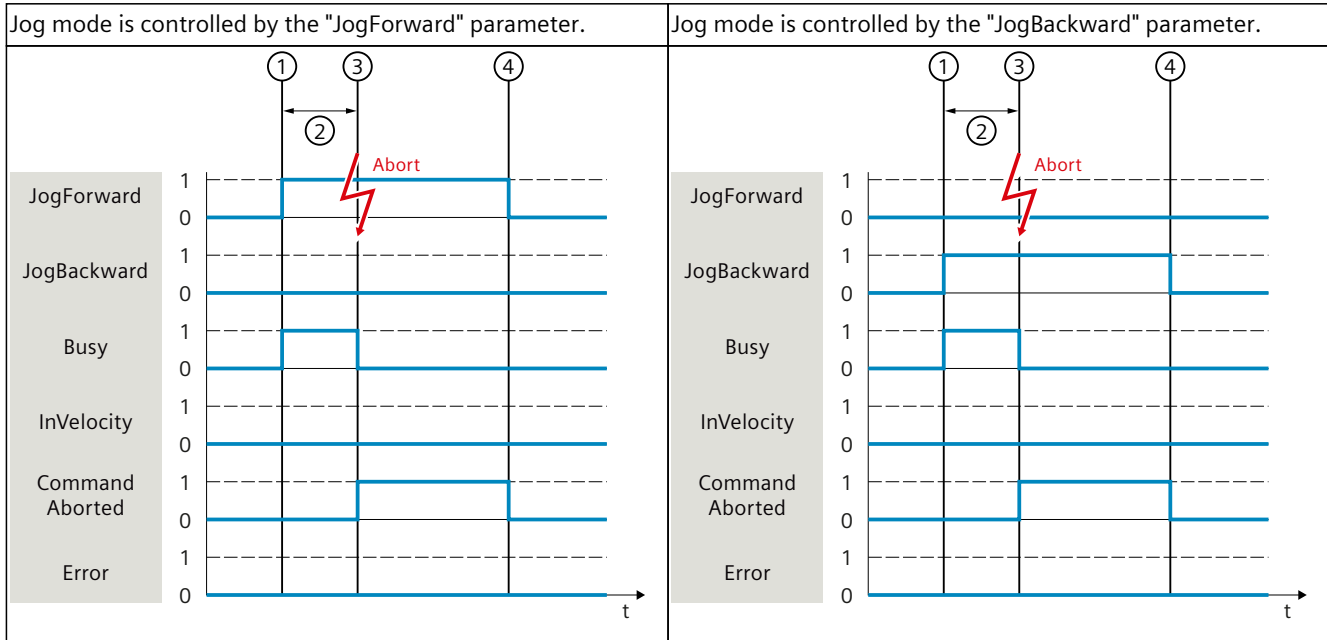
If the Motion Control job has been performed up to the point of reaching the assigned velocity, then this is indicated in the "InVelocity" parameter with the value "TRUE".



①	The job is started by setting the "JogForward" or "JogBackward" parameter.
②	While the job is being executed, the "Busy" parameter shows the value "TRUE".
③	When the assigned velocity is reached, the "InVelocity" parameter changes to "TRUE".
④	When the "JogForward" or "JogBackward" parameter is reset, the motion of the axis ends. The axis decelerates. The "InVelocity" parameter changes to "FALSE".
⑤	If the axis has come to a standstill, then the Motion Control job is complete and the "Busy" parameter changes to "FALSE".

The job is aborted during execution

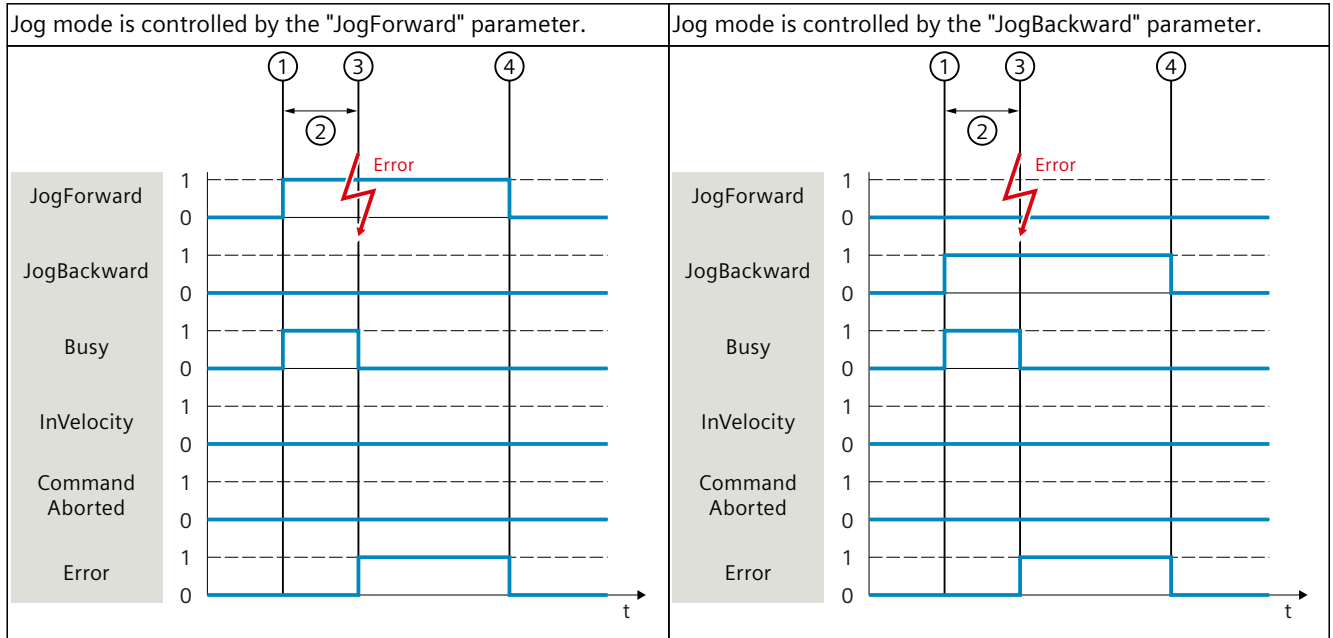
If the Motion Control job is canceled during processing by another job, this is indicated in the "CommandAborted" parameter with the value "TRUE". The behavior of the "CommandAborted" parameter is independent of reaching the assigned velocity.



①	The job is started by setting the "JogForward" or "JogBackward" parameter.
②	While the job is processing, the "Busy" parameter shows the value "TRUE".
③	During job execution, the job is aborted by another Motion Control job. When the job is aborted, the "Busy" parameter changes to "FALSE" and "CommandAborted" changes to "TRUE".
④	When the "JogForward" or "JogBackward" parameter is reset, the "CommandAborted" parameter likewise changes to "FALSE".

An error occurs during the execution of the job

If an error occurs during execution of the Motion Control job, this is indicated in the "Error" parameter with the value "TRUE". The behavior of the "Error" parameter is independent of reaching the assigned velocity.



- | | |
|---|---|
| ① | The job is started by setting the "JogForward" or "JogBackward" parameter. |
| ② | While the job is being executed, the "Busy" parameter shows the value "TRUE". |
| ③ | An error occurs during the execution of the job. When the error occurs, the "Busy" parameter changes to "FALSE", and "Error" changes to "TRUE". |
| ④ | When the "JogForward" or "JogBackward" is reset to the value "FALSE", the "Error" parameter likewise changes to "FALSE". |

7.6 Ending Motion Control jobs (S7-1500, S7-1500T)

When a job is ended, a distinction is made between error-free completion of the job and a motion cancelation.

Completion of job

The completion of a Motion Control job is indicated as described in the "Tracking running jobs (Page 110)" section.

Job termination

The termination and the substitution behavior are described in the section "Substitution behavior of Motion Control jobs". Special waiting jobs can be canceled with an "MC_Power" job.

Motion cancelation

If a motion must be canceled, you can perform the following measures:

- Execute "MC_Halt" or "MC_Stop"
To cancel a motion and stop the axis, you can use an "MC_Halt" or "MC_Stop" job.
- Execute "MC_GroupStop" (S7-1500T)
With the Motion Control instruction "MC_GroupStop", you stop and abort an active motion on the kinematics technology object.
- Disable "MC_Power"
In an emergency, you can stop the axis via an emergency stop ramp. To do this, set the "Enable" parameter of the "MC_Power" job to "FALSE". The axis is decelerated according to the selected "StopMode" and all jobs of the technology object are canceled.
- Execute "MC_GearOut" or "MC_CamOut" (S7-1500T)
To terminate gearing, you can desynchronize the following axis with a "MC_GearOut" job.
To end camming, you can desynchronize the following axis with an "MC_CamOut" job.
- Execute "MC_StopProgram" (S7-1500T)
To stop a motion controlled by the Interpreter technology object, stop the execution of the Interpreter program with an "MC_StopProgram" job.

Measuring job cancelation

With an "MC_AbortMeasuringInput" job you can cancel an active one-time or cyclic measuring job.

Cancellation of an active output cam/cam track

- "MC_OutputCam"
With "MC_OutputCam.Enable" = FALSE you can disable an output cam.
- "MC_CamTrack"
With "MC_CamTrack.Enable" = FALSE you can disable an active cam track.

7.7 Restart of technology objects (S7-1500, S7-1500T)

After the CPU is switched on, or after technology objects are downloaded into the CPU, the system automatically initializes the technology objects with the start values from the technology data block. If restart-relevant changes are detected during a reload into the CPU, a restart of the technology object is automatically performed.

If restart-relevant data have been changed in RUN mode by the user program, then the technology object must be reinitialized by the user in order for the changes to be used. At a RUN → STOP transition, the CPU automatically performs a restart of technology objects with restart-relevant changes.

If changes in the technology data block should also be retained after the restart of the technology object, then you must write the changes to the start value in load memory using the extended instruction "WRIT_DBL".

Restart required

If a restart of the technology object is required, this is indicated at "Technology object > Diagnostics > Status and error bits > Axis status or Encoder status > Online start value changed", as well as in the tag "<TO>.StatusWord.X3 (OnlineStartValuesChanged)" of the technology object.

Restarting a technology object

A restart of the technology object is triggered by the user by means of the "MC_Reset" Motion Control instruction, with parameter "Restart" = TRUE.

During a restart, all configuration data of the technology object are loaded from load memory into work memory. In the process, the actual values in the technology data block are overwritten.

Note the following during a restart of the technology object:

- A restart resets the "Homed" status of a technology object with incremental actual values ("<TO>.StatusWord.X5 (HomingDone)").
- While a restart is being performed, the technology object cannot perform any jobs. An active restart will be indicated under "Technology object > Diagnostics > Status and error bits > Axis status or Encoder status > Restart active", and in the "<TO>.StatusWord.X2 (RestartActive)" tag of the technology object.
- Motion Control jobs are rejected during a restart with the "Error" = TRUE and "ErrorID" = 16#800D parameters (job not executable, because a restart is active).
- Active Motion Control jobs to the technology objects output cam, cam track, measuring input and kinematics are canceled by a restart with "ErrorID" = 16#800D.
- While a restart is being executed, you cannot access the technology data block.

See also

[Change restart-relevant data \(Page 99\)](#)

7.8 Data types (S7-1500, S7-1500T)

Data types for the use of technology

The table below contains the data types for reference to the respective technology object:

Data type	Description
TO_Object ¹⁾	Basis of technology objects
TO_Axis ¹⁾	Basis of axes
TO_SpeedAxis ¹⁾	Speed axis
TO_PositioningAxis ¹⁾	Positioning axis
TO_SynchronousAxis ¹⁾	Synchronous axis
TO_ExternalEncoder ¹⁾	External encoder
TO_OutputCam	Output cam
TO_CamTrack	Cam track
TO_MeasuringInput	Measuring input
TO_CamBase ¹⁾	Basic information on cams (S7-1500T)
TO_Cam ¹⁾	Cam (S7-1500T)
TO_Cam_10k ¹⁾	
TO_Kinematics	Kinematics (S7-1500T)
TO_LeadingAxisProxy ¹⁾	Leading axis proxy (S7-1500T)
TO_Interpreter	Interpreter
TO_InterpreterProgram	Interpreter program
TO_InterpreterMapping	Interpreter mapping
PD_TELx	Telegram no. "x"
DX_TEL_SyncOp	Leading value telegram
PD_STW1_611Umode	Control word 1 (STW1)
PD_STW2_611Umode	Control word 2 (STW2)
PD_ZSW1_611Umode	Status word 1 (ZSW1)
PD_ZSW2_611Umode	Status word 2 (ZSW2)

¹⁾ Cascading technology objects

Cascading technology objects

The structure of the technology objects is structured as follows:

- "TO_Object" is the basis of all technology objects and a component of the "TO_Axis" and "TO_CamBase".
- "TO_Axis" is part of "TO_SpeedAxis", "TO_ExternalEncoder" and "TO_LeadingAxisProxy".
- "TO_SpeedAxis" is part of "TO_PositioningAxis".
- "TO_PositioningAxis" is part of "TO_SynchronousAxis".
- "TO_CamBase" is part of "TO_Cam" and "TO_Cam_10k".

Downloading to CPU (S7-1500, S7-1500T)

When downloading to the CPU S7-1500, it is always verified that the project files are consistent online and offline after the download. The data of the technology objects is stored in technology data blocks. The conditions for downloading blocks thus apply when downloading new or modified technology objects.

Downloading in "RUN" mode

When downloading in "RUN" mode of the CPU, a check is made to determine whether a download without restart of the technology object is possible. If restart-relevant configuration values were changed, then a restart of the technology object is automatically performed after the load into the CPU.

Loading a technology object is only possible if the technology object is disabled.

Downloading a function block with multi-instances of Motion Control instructions "MC_Power" is only possible under the following preconditions:

- "MC_Power.Enable" = FALSE
- "MC_Power.Status" = FALSE
- "<TO>.StatusWord.X0" = FALSE

Downloading a function block with multi-instances of Motion Control instructions "MC_TorqueLimiting" is only possible under the following preconditions:

- "MC_TorqueLimiting.Enable" = FALSE
- "MC_TorqueLimiting.Busy" = FALSE
- "<TO>.StatusWord.X26" = FALSE

The following changes **cannot** be downloaded to the CPU in "RUN" mode:

- Changes to the cycle clocks of MC_Servo
- Changes to the hardware interface of the technology object in "Technology object > Configuration > Hardware interface"

NOTE

Multi-instance DBs

If you use multi-instances of the Motion Control instructions "MC_Power" or "MC_TorqueLimiting", create the multi-instances in a separate instance data block. This allows you to download program blocks from other sections of your user program without switching off the axes, including in "RUN" mode.

Diagnostics (S7-1500, S7-1500T)

The description of Motion Control diagnostics is limited to the diagnostics view of the technology objects in TIA Portal, the technology alarms and the error IDs on Motion Control instructions.

The following descriptions can be found in the "S7-1500/S7-1500T Motion Control alarms and error IDs" documentation ([Page 8](#)):

- Diagnostics concept
- Technology alarms
- Error IDs in Motion Control instructions

A comprehensive description of the system diagnostics of the S7-1500 CPU can be found in the "Diagnostics" function manual

(<https://support.industry.siemens.com/cs/ww/en/view/59192926>).

Glossary

(S7-1500, S7-1500T)

Absolute synchronous operation

Function corresponds to the Motion Control instruction MC_GearInPos or MC_CamIn.

Absolute value encoder

Position encoder which outputs the position in the form of a digital numerical value. This numerical value is unique within the entire measuring range of the absolute value encoder.

Axis control panel

The axis control panel allows you to move the axis in manual mode, optimize the axis settings, and test the operation of the axis in your system.

Axis type

The axis type differs depending on the unit of measurement according to which the axis is positioned.

Depending on the execution of the mechanics, an axis is implemented as a linear axis or rotary axis:

- For linear axes, the position of the axis is specified as a linear measure, e.g. millimeters (mm).
- For rotary axes, the position of the axis is specified as an angular measure, e.g. degrees (°).

Communication processor (CP)

Module for expanded communications tasks covering special applications, for example in the area of security.

Communications module (CM)

Module for communications tasks which is used as an interface expansion of the CPU (for example PROFIBUS) or provides additional communications options (e.g. PtP) in an automation system.

Drive

The combination of motor (electric or hydraulic), actuator (converter, valve), control system, measuring system and supply (infeed, accumulator).

Dynamic Servo Control (DSC)

In drives that support DSC, you can optionally use the position controller in the drive. The position controller in the drive is usually implemented with a rapid speed-control cycle. This improves the control performance for digitally coupled drives.

Following error

The following error is the difference between the position setpoint and the actual position value. The transmission times of the setpoint to the drive, and of the actual position value to the controller, are taken into account in the calculation of the following error.

GSD file

As a Generic Station Description, this file contains all properties of a PROFINET or PROFIBUS device that are necessary for its configuration.

Hardware limit switch

Mechanical limit position switch that limits the maximum permissible traversing range of the axis.

Homing

With homing, you create the relationship between the position in the technology object and the mechanical position of the axis. The position value in the technology object is assigned to a homing mark at the same time. This homing mark represents a known mechanical position.

Incremental encoder

Position encoder which outputs the position change incrementally in the form of a digital numerical value.

Kv factor

Gain factor of the position controller

Master value

Input value for synchronous operation

Motion Control instruction

Use the Motion Control instructions to start Motion Control jobs at technology objects in your user program and thus execute the desired functionality at the technology objects. You track the status of running jobs with the output parameters of the Motion Control instructions.

Override

Percentage correction of the velocity/speed

Processing cycle clock

The processing of a technology object in the servo cycle clock.

PROFIdrive

PROFIdrive is a profile specified by the PNO (PROFIBUS user organization) for PROFIBUS DP and PROFINET IO for speed- and position-controlled drives.

PROFIdrive frame

Frame for communication according to PROFIdrive.

Relative gearing

Function corresponds to the Motion Control instruction MC_GearIn.

Restart

The technology object is reinitialized with the current configuration parameters.

Safe Stop 1 (SS1)

The Safe Stop 1 (SS1) safety function brings a drive to standstill quickly and safely via an internal rapid stop ramp. Safe Torque Off (STO) is activated after standstill. STO ensures that no more torque generating energy acts on a drive. This prevents unintended startup of the drive.

You can use the SS1 safety function when a fast stop of the drive with a subsequent transition to STO is required. SS1 is used, for example, to quickly stop high inertia loads or to brake drives quickly and safely at high speeds

Safe Stop 2 (SS2)

The Safe Stop 2 (SS2) safety function brings a drive to standstill quickly and safely via an internal rapid stop ramp. After standstill is reached, the standstill position is monitored on the drive side. The drive can deliver full torque to maintain the standstill.

SS2, for example, is used for processing machines and machine tools.

Safe Torque Off (STO)

The Safe Torque Off (STO) safety function is the most commonly used and most basic drive-internal safety function. STO ensures that no more torque generating energy acts on a drive. This prevents unintended startup of the drive. The pulses of the drive are eliminated. The drive is reliably torque-free. This state is monitored internally in the drive.

You can use STO when the drive comes to a standstill in a sufficiently short time on its own due to the load torque or due to friction. Other areas of use are where "coasting" of the drive has no relevance for safety.

Software limit switch

A programmable position which limits the traversing range of an axis.

Synchronization

The phase of the following axis to reach synchronous movement.

Synchronous operation

Defined synchronous movement after synchronization of a following axis to a leading axis.

Technology alarm

If an error occurs at a technology object (e.g. approaching a hardware limit switch), a technology alarm is triggered and indicated.

The impact of a technology alarm on the technology object is specified by the alarm reaction (e.g. remove enable). The alarm reaction is specified by the system.

Technology data block

The technology data block represents the technology object and contains all configuration data, setpoint and actual values, and status information of the technology object.

Technology module (TM)

Module for technological tasks, e.g. counting, measuring and positioning.

Zero mark

Position reference for the movement of rotary and linear incremental encoders. The zero mark of an incremental encoder is used as a homing mark, for example.

Index

A

Actuator, [21](#)

D

Data type

DB_ANY, [106](#)

DB_ANY, [107](#)

DB_ANY, [106](#), [107](#)

E

Extended Motion Control Resources, [22](#)

External encoder

Adding, [51](#)

Moving, [53](#)

Deleting, [53](#)

I

Interpolator OB, [86](#), [90](#)

M

MC_Interpolator OB, [86](#), [90](#)

MC_Servo OB, [86](#), [90](#)

Motion Control resources, [22](#)

P

Positioning axis

Adding, [51](#)

Moving, [53](#)

Deleting, [53](#)

Process image partition "OB Servo PIP", [89](#)

R

Reduction ratio, [86](#)

Reinitialization of technology objects, [123](#)

Restart of technology objects, [123](#)

S

S7-1500 Motion Control

How it works, [18](#)

Introduction, [18](#)

Technology object, [18](#)

Technology object data block, [19](#)

Motion Control instruction, [19](#)

Technology object, [19](#)

Drive and encoder connection, [21](#)

Configuration limits, [22](#)

Versions, [40](#)

Versions, [44](#)

Technology object, [51](#)

Configuration, [51](#)

Configuration, [53](#)

Technology object, [53](#)

Configuration, [54](#)

Configuration, [55](#)

Process response, [86](#)

Process response, [89](#)

Process response, [90](#)

Process response, [94](#)

Technology data block, [96](#)

Technology data block, [96](#)

Programming, [96](#)

Technology data block, [98](#)

Technology data block, [99](#)

Programming, [100](#)

Motion Control instruction, [100](#)

Motion Control instruction, [103](#)

Programming, [109](#)

Motion Control instruction, [109](#)

Technology object, [123](#)

Programming, [123](#)

Downloading to CPU, [125](#)

S7-1500 Motion Control

Technology object, [53](#)

Configuration, [53](#)

S7-1500 Motion Control drive connection, [21](#)

S7-1500 Motion Control encoder connection, [21](#)

- S7-1500 Motion Control instruction, [19](#)
 - Parameters, [100](#)
 - Inserting, [103](#)
 - Starting Motion Control job, [109](#)
 - Tracking Motion Control job, [110](#)
 - Ending a Motion Control job, [121](#)
- S7-1500 Motion Control operating mode, [94](#)
- Sensor, [21](#)
- Servo OB, [86](#), [90](#)
- Speed axis
 - Adding, [51](#)
 - Moving, [53](#)
 - Deleting, [53](#)
- Synchronous axis
 - Adding, [51](#)
 - Moving, [53](#)
 - Deleting, [53](#)

T

- Technology data block
 - Basics, [96](#)
 - Analyzing, [96](#)
 - Evaluating StatusWord, ErrorWord and WarningWord, [98](#)
 - Change restart-relevant data, [99](#)
- Technology object
 - External encoder, [51](#)
 - Synchronous axis, [51](#)
 - Positioning axis, [51](#)
 - Speed axis, [51](#)
 - External encoder, [53](#)
 - Synchronous axis, [53](#)
 - Positioning axis, [53](#)
 - Speed axis, [53](#)
 - Speed axis, [53](#)
 - Positioning axis, [53](#)
 - Synchronous axis, [53](#)
 - External encoder, [53](#)
 - Data types, [105](#)
- Technology object data block
 - Basics, [19](#)
- Technology objects, [18](#), [19](#)