SIEMENS

SIMOTION

Terms and Abbreviations

Glossary

05/2009

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

with a safety alert symbol, indicates that minor personal injury can result if proper precautions are not taken.

CAUTION

without a safety alert symbol, indicates that property damage can result if proper precautions are not taken.

NOTICE

indicates that an unintended result or situation can occur if the corresponding information is not taken into account.

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 device/system may only be set up and used in conjunction with this documentation. Commissioning and operation of a device/system may only be performed by **qualified personnel**. Within the context of the safety notes in this documentation qualified persons are defined as persons who are authorized to commission, ground and label devices, systems and circuits in accordance with established safety practices and standards.

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 adhered to. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of the 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.

Siemens AG
Industry Sector
Postfach 48 48
90026 Nürnberg
GERMANY

Preface

This **document** is part of the **SIMOTION Engineering System Handling** documentation package.

Chapters in this manual

The glossary defines important terms relating to the SIMOTION Engineering System.

The glossary is arranged alphabetically and replaces the glossaries in the individual manuals

SIMOTION Documentation

An overview of the SIMOTION documentation can be found in a separate list of references.

This documentation is included as electronic documentation with the supplied SIMOTION SCOUT.

The SIMOTION documentation consists of 9 documentation packages containing approximately 80 SIMOTION documents and documents on related systems (e.g. SINAMICS).

The following documentation packages are available for SIMOTION V4.1 SP4.

- SIMOTION Engineering System
- SIMOTION System and Function Descriptions
- SIMOTION Service and Diagnostics
- SIMOTION Programming
- SIMOTION Programming - References
- SIMOTION C
- SIMOTION P350
- SIMOTION D4xx
- SIMOTION Supplementary Documentation

Hotline and Internet addresses

Siemens Internet address

The latest information about SIMOTION products, product support, and FAQs can be found on the Internet at:

- General information:
  - **http://www.siemens.de/simotion** (German)
  - **http://www.siemens.com/simotion** (international)
- Downloading documentation
  Further links for downloading files from Service & Support.
  http://support.automation.siemens.com/WW/view/en/10805436
- Individually compiling documentation on the basis of Siemens contents with the My Documentation Manager (MDM), refer to
  http://www.siemens.com/mdm

  My Documentation Manager provides you with a range of features for creating your own documentation.
- FAQs
  You can find information on FAQs (frequently asked questions) by clicking **http://support.automation.siemens.com/WW/view/en/10805436/133000**.

Additional support

We also offer introductory courses to help you familiarize yourself with SIMOTION.

For more information, please contact your regional Training Center or the main Training Center in 90027 Nuremberg, Germany.

Information about training courses on offer can be found at:

www.sitrain.com

Technical Support

If you have any questions, please contact our hotline:

	Europe / Africa
Phone	+49 180 5050 222 (subject to charge)
Fax	+49 180 5050 223
0.14/min from German wire-line network, mobile phone prices may differ.	
Internet	http://www.siemens.com/automation/support-request

	Americas
Phone	+1 423 262 2522
Fax	+1 423 262 2200
E-mail	techsupport.sea@siemens.com

	Asia / Pacific
Phone	+86 1064 757575
Fax	+86 1064 747474
E-mail	support.asia.automation@siemens.com

Note

Country-specific telephone numbers for technical support are provided under the following Internet address:

http://www.automation.siemens.com/partner

Questions about this documentation

If you have any questions (suggestions, corrections) regarding this documentation, please fax or e-mail us at:

Fax	+49 9131-98 2176
E-mail	docu.motioncontrol@siemens.com

Glossary

A

Absolute encoder

Position or shaft encoder that supplies the axis position as an absolute actual value immediately after its power supply unit is switched on. Absolute information about the current angular or linear position of the axis is therefore always available. The absolute encoder outputs the positional information using a parallel or serial transmission method, e.g. as dual or gray code.
Shaft encoders are grouped into single-turn encoders and multiturn encoders. The uniqueness range is one revolution on single-turn encoders and multiple (typically 4,096) revolutions on multiturn encoders.

Application: An absolute encoder is used when homing is not possible or not desired.

Absolute encoder adjustment

→ Homing absolute encoder, Types of homing

Access error

An error that occurs when accessing a variable or I/O from the user program.

Active homing

A special traversing motion is executed for active homing. The following homing modes can be selected in the configuration:

- Homing with homing output cam and encoder zero mark
- Homing with external zero mark only
- Homing with encoder zero mark only

→ Types of homing, Reference cam, External zero mark, Encoder zero mark

Actual value

Actual value of a process variable, e.g. position, velocity, pressure, measured at a defined time.

Actuation time

Lead time for the switching of an output cam, in order, for example, to compensate for hardware switching time conditions; the controller determines the internal switching position depending on the current velocity and the actuation time.

Actuator interface

The actuator interface is the interface of the axis to the drive. When the drive is connected via PROFIdrive message frame, this encompasses not only the setpoint, but also the control signals of the axis to the drive as well as the status information and the actual speed value from the drive to the controller.

→ Encoder interface

Addition object

→ TO additionObject

ADI4

The ADI4 interface module (analog drive interface for 4 axes) can be used with SIMOTION to operate up to 4 drives with an analog setpoint interface on the isochronous PROFIBUS DP.

These can be electric or hydraulic drives. The ADI4 can also be used as an external encoder for SIMOTION.

Alarm

SIMOTION differentiates alarms as follows:

- Peripheral alarms are generated by the SIMOTION Kernel.
  Peripheral alarms are alarms that originate from peripheral components, such as station, diagnostic, process, and pull/plug alarms.
- Technology alarms are generated by the technology objects.

→ Process interrupt, Diagnostic interrupt, Technology alarm

Alarm_S

Messages that can be programmed by the user; they can be issued and acknowledged via system functions.

→ Process fault diagnostics

Analog drive interface for 4 axes

→ ADI4

Anti-windup

Stops the I component of the controller during limitation of the total manipulated variable.

Application project

SIMOTION project with preconfigured hardware configuration and integration into the SIMOTION execution system.

Articulated joint positioning space

If Cartesian kinematics end points can be reached via various articulated joint positions, articulated joint positioning spaces are defined for the corresponding kinematics.

→ Kinematics

ASIC

Application Specific Integrated Circuit

AS-Interface

Actor sensor interface

ASM

Interface modules are components of RFID systems and are used to connect write/read devices.

→ RFID

Asynchronous command execution

→ Cyclic programming

Asynchronous exception

Results either from time monitoring, a task, a program execution error, or a technology object fault and is triggered asynchronously to the program execution.

→ Synchronous exception

Average manipulated variable (TControl only)

In the case of a failure or fault in the actual value acquisition, the average actuating signal of the previous controller cycles can be output until the fault is corrected.

Axis

The term "axis" comprises the following elements:

- Drive (current/speed control)
- Axis control (actual value system, position control, monitoring, etc.)
- Functionality (positioning, synchronous operation, etc.), and
- Configuration (homing, mechanics, etc.)

In SIMOTION SCOUT, the axis is made available to the user in the form of a technology object in the project navigator; the user can access any of the axis components listed above via the Axis technology object. Axes can be controlled from the user program via technology commands.

→ Drive

Axis coordinate

Coordinate system in which the technological quantity of the axis is represented (e.g. in mm or °) (also known as the axis coordinate system)

Axis simulation

The calculated axis setpoints are not output to the drive or final controlling element, and the actual position values are not received from an encoder, but are calculated via a simulation of the actual values depending on the setpoints. I/O devices or drives do not have to be available.

→ Simulation mode

B

BackgroundTask

The BackgroundTask is executed in the round-robin execution level together with the MotionTasks. It is a cyclic task.

→ Execution level, Round-robin execution level, MotionTask

Backlash

The term backlash (also backlash on reversal, clearance, play, dead travel on reversing) designates the distance/angle that a spindle or motor must travel in the event of a direction reversal until the axis (cradle, machine table, tool carrier) is moved in the other direction again.

Backlash compensation

Automatic allowance for backlash in the controller when traversing or positioning the axis.

Balancing filter (axis)

A balancing filter is used to delay setpoints by the value of the process transition time before the setpoints are compared with the actual value and switched into the process again, multiplied by the position control gain.

When precontrol is active, allowance can be made in the balancing filter for the process response prior to formation of the system deviation from the position setpoint and actual position.

Basic coordinate system

Coordinate system of path interpolation. A clockwise, rectangular coordinate system in accordance with DIN 66217 is used.

Basic cycle clock

Reference cycle clock for the interpolator and position control cycle clock when the equidistant DP cycle clock is not activated.

→ Cycle clock

Blending

Specific method of connecting or lining up motions in which the transition velocity between the two motions results from the lesser of the programmed velocities. This transition velocity is reached exactly at the transition point, which is the target position of the first motion.
Otherwise, the individual motions are applied with the velocity profiles corresponding to their dynamic specifications.

BOP

Basic Operator Panel

Bus cycle

Periodic recurring process in isochronous mode with deterministic cycle time, e.g. DP cycle or send interval.
The bus cycle is the cycle synchronized to the SIMOTION and also the cycle which the processing cycle clock is generated from.

→ Bus cycle clock

Bus cycle clock

The bus cycle clock (DP cycle clock on PROFIBUS and sending cycle on PROFINET) is used as a basis for setting further tasks.

The bus cycle clock is the fastest SIMOTION processing cycle clock, which all other system cycle clocks are derived from (e.g. servo, IPO, IPO 2, etc.). Communication with drives and I/O takes place via the bus cycle clock.

The bus cycle clock is started deterministically by a clock signal at the beginning of each bus cycle.

→ Bus cycle

C

C2xx Kernel

Name for the SIMOTION Kernel of the C230-2, C240, and C240 PN hardware platforms.

Cam

→ TO Cam

CamEdit

Simple cam editor integrated in SIMOTION SCOUT for text-based input of interpolation points and polynomials.

→ SIMOTION SCOUT, Cam

Camming

Camming is a non-linear angular-locked synchronism, whose transmission ratio is specified in a cam.

CamTool

→ SIMOTION CamTool

Cam track

Cam tracks allow several output cams to be output as a track on one output.

→ TO camTrack, Technology object, Output cam

Cartesian axes

Axes X, Y, and Z of the path object

→ Basic coordinate system

CE

Controller enable

CF

CompactFlash

Circular path

2D- or 3D-path that describes a polynomial segment.

Clocking

Passing on of the products from machine station to machine station.

Clocking, central

All machine stations pass on products simultaneously (synchronously).

→ Clocking, multiple

Clocking, local

One, several or all machine stations can pass on products at different times (asynchronously).

Clocking, multiple

Variant of central clocking in which the clockings of the machine stations differ through divider ratios.

→ Clocking, central

CoC

Center of Competence

Combo box

A combo box is a selection list in SIMOTION SCOUT that can also be edited.

Command

Programming unit used for open-loop and closed-loop control of motions, assignment of variables, etc. A command is represented as a block with graphic symbols in an MCC chart. MCC commands are executed from the start node down to the end node.

→ System function

Command library

Structured command library with system functions organized by subject and including a search function.

CompactFlash Card

CompactFlash card

→ Memory Card

Configuration data

Configuration data define the basic functionality of a technology object. They are normally set during the configuration of the technology object with SIMOTION SCOUT and cannot be modified during runtime.

A majority of the configuration data can be modified even during runtime via the user program or the expert list in SIMOTION SCOUT.

→ System variable

Configuration data buffer

Buffer for configuration data. There are configuration data buffers for:

- Start values
- Current values
- New values (for online changes).

The configuration data buffer for new values can be reset.

The control zone improves the dynamic response of the control and reduces overshoots. Maximum cooling occurs above the control zone to be defined, and maximum heating takes place below the control zone. Within the control zone, the actuating signal calculated by the controller is output.

Controller

→ Servo

Controller cycle time

→ Sampling controller, Sampling time

Controller object

→ TO controllerObject, CPU

Control zone (TControl only)

The control zone is a band on either side of the current manipulated variable that is formed by a value for the upper limit and a value for the lower limit.

Cooling controller

Controller with one digital output for cooling. The controller has two operating modes:

- Cooling
- OFF

Counter cams

In the case of modulo rotary axes, it is possible to specify whether the output cam should switch in every single or only in every nth pass.

Coupling

Method of feeding products to a machine station.

Coupling, fixed

Products are coupled with the feeder in a fixed manner. Empty locations in the feeder are possible.

Coupling, loose

Products are not coupled with the feeder in a fixed manner. The products line up automatically. Empty locations in the feeder are not desirable.

CP

Communication processor

Communication processors (CP) implement serial data exchange via a point-to-point connection and the connection of AS-Interface slaves.

CPU

Central Processing Unit

CUA

Control Unit Adapter

Current data

Current technology object data in current memory (configuration data and system variables).

Cycle clock

For example, the following cycle clocks are defined for SIMOTION:

- Basic cycle clock
- Bus cycle clock
- Position control cycle clock
- Interpolator cycle clock
- Cycle time
- DCC cycle clock

Cycle time

→ Cycle clock

Cyclic mode

With a cyclic application, a function, such as a cam, is performed cyclically, i.e. it is repeated continuously.

Cyclic programming

Asynchronous command execution The technology function is transferred to the technology object and the program is resumed immediately. Feedback messages must be evaluated explicitly by querying the axis status or command status.

This method of motion control is referred to as *cyclic programming*. It is permitted in all system tasks and is intended especially for the programming of cyclic tasks.

→ Technology function, Technology object, Sequential programming

D

DAC

Digital-analog converter

Data exchange broadcast

Data exchange broadcast is a function of PROFIBUS DP-V2. It is the direct exchange of data between PROFIBUS slaves (e.g. drive converters) with no involvement from the bus master.

→ PROFIBUS DP-V2

Data set

Collection of product features and their IT map. For example, data sets contain the axis configuration data.

A data set changeover occurs when a change is made during cyclic operation from one data set to another data set of the same type but having different contents (e.g. changeover of controller parameters).

Data set structure

Structure (data types, sequence), in which the collection of product features is stored in the data set.

Data type

A data type is used to determine how the value of a variable or constant is to be saved and used.

→ Elementary data types, User-defined data types

DCB

→ Drive Control Block

DCC

Drive Control Chart

DCC enables SIMOTION and SINAMICS users to also implement and graphically configure drive-related tasks employing continuous closed-loop and open-loop control.

A set of Drive Control Blocks (DCB) is available for this purpose. These blocks can be graphically interconnected and configured in so-called "charts" via a configuration tool (DCC editor).

→ Drive Control Block

DESINA

Distributed and standardized installation technology for machine tools and manufacturing systems

Detail view

The detail view is part of the workbench and displayed by default at the bottom of the screen. It displays detailed information about the currently active, selected object. Tabs can be used to browse through this detailed information.

→ Workbench, Symbol browser

Device level

The device level contains the SIMOTION Kernel. The SIMOTION Kernel contains the basic functionality (e.g. communication) as well as the capability of being programmed in ST (IEC1131-3) with PLC functionality (combinational logic, etc.). The device level makes system functions and system variables of the device available on the user level. This enables the device functionality to be used in user programs.

→ SIMOTION RT

Device update tool

→ Upgrading

DI

Digital input

Diagnostic interrupt

→ Alarm, SystemInterrupt

DIP

Dual In Line Package

Direct homing

The axis position is set without traversing having taken place. This is also referred to as home position setting.

→ Types of homing

Distributed synchronous operation

The *distributed synchronous operation* functionality allows you to create a master value source and a following axis on different controllers.
Cross-project *distributed synchronous operation* is also possible.

DMC20

You can use a DMC20 DRIVE-CLiQ hub to extend the number of DRIVE-CLiQ interfaces and facilitate a point-to-point topology.

→ DRIVE-CLiQ

DO

Abbreviation for:

- Digital output
- Drive object

DP

Distributed I/O

→ PROFIBUS DP

DP cycle

→ Bus cycle

DP cycle clock

→ Bus cycle clock

DPRAM

Dual-port random access memory

DP-V1

→ PROFIBUS DP-V1

DP-V2

→ PROFIBUS DP-V2, Data exchange broadcast, Equidistance mode

DRAM

Dynamic random access memory

Drive

The drive includes the (electric or hydraulic) motor, the final controlling element (converter, valve), the control unit, the measuring system and the supply components (infeed, pressure accumulator).

For electric drives, a distinction is made between a converter system and an inverter system:

- On a converter system (for example MICROMASTER 4), from the user's point of view, the infeed, final controlling element, and control unit are grouped together in one device;
- while on an inverter system (for example SINAMICS S), the supply is executed via a Line Module to make a DC link, which the inverter (Motor Module s) is then connected to.
  The control unit is implemented as a separate device and connected to the other components via DRIVE-CLiQ.

Position control functionality can be configured on the drive unit (control unit + power unit + motor).

Technology functions are also available on the drive unit.

With machine or plant automation, the drive executes the setpoints generated in a higher-level automation system and returns the actual state.

This is normally a motion, but can also be a force or a torque.

→ Axis, DRIVE-CLiQ

Drive axis

→ Technology object, TO driveAxis, Drive Control Chart

DRIVE-CLiQ

Drive Component Link with IQ

DRIVE-CLiQ is used to connect the closed-loop control module with the power components, encoders and other system components, such as terminal modules. Setpoints and actual values, control commands, status messages, and type plate data for the components are transmitted via DRIVE-CLiQ.

DRIVE-CLiQ significantly simplifies commissioning and diagnostics since all connected components are identified automatically with the help of an electronic type plate.

The use of standardized cables and connectors reduces the variety of different parts, thereby lowering storage costs.

Drive Control Block

Blocks which are created in C programming language and can be used in the DCC.

→ DCC

Drive Control Chart

→ DCC

Drive object

Drive object of a SINAMICS drive unit

→ DO

DSC

→ Dynamic Servo Control enables the actual position value to be evaluated directly in the drive in a fast speed controller cycle.

Dynamic response adaptation

The setpoint circuit of the control loop contains a parameterizable, dynamic PT2 filter with time constants T1 and T2 for adapting the dynamic behavior of axes. This feature allows you to adapt the dynamic response of faster axes to the response of the slowest axis.

Dynamic Servo Control

With the Dynamic Servo Control function, the dynamically active component of the position controller in the drive is executed with the sampling time cycle of the speed loop.
It is thus possible to set a substantially greater position controller gain factor Kv in relation to the sampling times.
This increases the dynamic response for setpoint sequence and disturbance compensation in highly dynamic drives.
DSC is possible for drives which are linked via PROFIdrive message frames.

E

ECOFAST

Energy and Communication Field Installation System

Elementary data types

Elementary data types define the structure of data that cannot be broken down into smaller units. An elementary data type describes a memory range with a fixed length and stands for bit data, integers, floating-point numbers, duration, time of day, and date.

→ User-defined data types, Data type

EMC

Electromagnetic compatibility

Enable depending on conditions

To enable a process with the mode depending on conditions, e.g. switching from position control to force control on an axis depending on conditions.

Enabling signal

The connected drive is switched on using closed-loop control (drive connection, pulse enable, setpoint enable, etc.) via special enabling signals, e.g. to an Axis TO according to PROFIdrive.

Encoder interface

When the encoder is connected via PROFIdrive message frame, the encoder interface encompasses not only the actual encoder value, but also the control signals of the axis to the encoder evaluation, e.g. in the drive, as well as the status information of the encoder evaluation, e.g. in the drive, to the controller.

→ Actuator interface

Encoder zero mark

The zero mark of an incremental encoder is used as a reference mark.

→ Reference mark

Equidistance mode

Equidistance mode in relation to a bus system (PROFIBUS DP V2, PROFINET IO with IRT) indicates that the transmission of process data occurs in cycles and at equal time intervals (isochronous mode) and the application functions (tasks) in the bus nodes are synchronized to this cycle.
Isochronous communication takes place via a global control telegram, sent by the master to all bus nodes when the cycle starts. This is the basis for generating a bus cycle in each individual bus node.

Special telegram failure strategies in the bus nodes ensure that communication remains stable, even if the equidistant global control telegram fails sporadically ("bus clock PLL" for bridging clock failures).

In drive technology, isochronous communication forms the basis of drive linking according to the PROFIdrive profile. The bus system not only carries message frame traffic in an equidistant time scale, but the internal control algorithms, such as the speed and current controller in the drive or controller, are also time-synchronized in the higher-level automation system.

→ PROFIBUS DP-V2, Isochronous mode, PROFINET IO with IRT, PROFIdrive profile, Bus cycle clock

EP

Enable Pulses

ES

Engineering system

ESD

Electrostatic Sensitive Devices

ExecutionFaultTask

Fault task that is called when program execution errors occur.

→ Execution level, Fault task

Execution level

Execution levels define the chronological sequence of tasks in the execution system.
Each execution level can be assigned at least one task.
For example, the following execution levels are defined in SIMOTION:

- StartupTask
- MotionTasks
- BackgroundTask
- TimerInterruptTasks
- SynchronousTasks
- SystemInterruptTasks
- UserInterruptTasks
- ShutdownTask
- Fault tasks

→ Execution system, Task, Fault task

Execution system

The SIMOTION execution system provides a series of execution levels with various execution properties and interrupts.

→ Execution level

Extensible Markup Language

→ XML

External zero mark

An external signal (external zero mark) is used as the reference mark.

→ Reference mark

F

Fault task

Fault tasks are sequential tasks that are executed one time after start and then terminated. Examples are:

- PeripheralFaultTask
- TimeFaultTask
- TimeFaultBackgroundTask
- TechnologicalFaultTask
- SystemFaultTask
- ExecutionFaultTask

→ Execution level

FB

→ Function block

FBD

→ Function Block Diagram

FC

→ Function

Feed drive

FD

FFT

Fast Fourier Transformation

Fine interpolator (axis)

The purpose of the fine interpolator (FIPO) is to generate interim setpoints for the position setpoints when the interpolator (IPO) and the controller (servo) have different cycle clock ratios.

FIPO

→ Fine interpolator (axis)

Firmware

Firmware is also identified as a kernel in SIMOTION C, SIMOTION P, and SIMOTION D.

→ Kernel

Fixed gear

→ TO fixedGear

FM

Function module

Function modules relieve the CPU from time-intensive tasks such as counting or outputting of cams.

→ CPU

Following axis

In the synchronous operation function, the following axis is the axis that follows the motion of the leading axis. The following axis is also referred to as the slave axis.

→ TO followingAxis, Leading axis

Following error

The difference between the axis setpoint and the axis actual value in the servo.

Follow-up mode

The setpoint is corrected to the actual value in follow-up mode of the TO axis. The actual position and actual speed values will be updated.
Motion commands are not accepted or executed.

Force/pressure control

Establishing of a specifiable force/pressure in relation to a measured force/pressure signal via a force/pressure controller in the process, with an axis/drive as actuator.

→ Force/pressure controller

Force/pressure controller

→ Force/pressure control

Formula object

→ TO formulaObject

FPU

Floating-point unit

Designates a special CPU that is used for processing floating-point numbers.

FRAM

Ferroelectric Random Access Memory

Non-volatile electronic memory with ferroelectric properties.

Function

The term "function" (FC) can have two different meanings depending on the context:

1. Generally, the term is used to describe a functional system characteristic.
2. A function is defined as a program organization unit (POU) that, when executed, supplies exactly one data element (that can have multiple values) and that can be called in text languages to act as an address in an expression.

→ Program organization unit

Function block

A function block (FB) is a program organization unit that supplies one or more values when executed. Several named instances of a function block can be generated.

→ Program organization unit, Instance

Function Block Diagram

Function Block Diagram (FBD) is one of the three programming languages of STEP 5 and STEP 7 and an important component of IEC 61131.
FBD uses the logic boxes familiar from Boolean algebra to map logic. In addition, complex functions (e.g. mathematical functions) can be represented directly in conjunction with the logic boxes. Conversion to another programming language (e.g. Ladder Diagram) is possible.

→ Ladder diagram

FW

→ Firmware

G

Gearing

A linear angular-locked synchronism with constant gear ratio.

Gear ratio

Gear ratio for gearing or fixed gear.

Global device variables

Global device variables (scope: device) are created in the "GLOBAL DEVICE VARIABLES" container (project navigator) and are valid within the sources (programs, functions, function blocks, etc.) assigned to the "Programs" container. They can be viewed and controlled in the HMI/OPC.

→ Global user variable

Global measuring

Global measuring inputs can be freely assigned to the axes and add an internal timestamp to the measurement result for more precise determination of the axis positions.

The input assignment is not permanently related to the relevant hardware. Therefore, a measurement can be carried out using any encoder available in the project.

Global measuring inputs are available on the D410, D4x5, CX32, CU3xx, TM15/TM17 High Feature, or C240 (B1-B4).

The term "central measuring input" is also used within the context of drives.

→ Local measuring

Global user variable

Global user variables are classified according to their scope as follows:

- I/O variables (device scope)

  enable direct access to the I/O. After the "I/O" container is selected (in the project navigator), the I/O variables are created in the symbol browser and can be viewed and controlled in the HMI/OPC.

- Global device variables (device scope)

  are created in the "Programs" container (project navigator) and are valid within the sources (programs, functions, function blocks etc.) assigned to the "Programs" container. They can be viewed and controlled in the HMI/OPC.

- Unit variables (unit scope)

  are created within a unit in the source code and are valid for all programs, functions and function blocks within this unit.

  Depending on the section of the source code where the unit variables are defined, they are available in different ways:

  Unit variables defined in the implementation section can be viewed and controlled in the symbol browser.

  Unit variables defined in the interface section can also be viewed and controlled in the HMI/OPC and can be used by other units on the same device via the "USES <unit_name>" command.
  Global device variables can also be defined here.

- HMI variables (HMI scope)

  General term for global user variables that can be viewed and controlled in the HMI.

→ Variable, Local user variable, Symbol browser

GND

Reference point for zero potential (ground)

GSD

Device master data (device parameter list) for integration in HW Config.

H

Hardware platform

→ SIMOTION hardware platform

Heating and cooling controller (temperature channel)

Controller with two digital outputs for heating and cooling. The controller has three operating modes:

- Heating
- Off
- Cooling

Heating controller (temperature channel)

Controller with one digital output for heating. The controller has two operating modes:

- Heating
- OFF

Heating current monitoring (temperature channel)

Monitoring of heating current; for this purpose, only one heating switch is closed (100%) and a check made for heating current flow.

Heating current sensing (temperature channel)

Measurement of the heating current controlled by the heating switch.

HMI

→ Human Machine Interface

HMI acknowledge buffer

Memory area from which job execution feedback messages on the HMI can be read out.

HMI variable

→ Global user variable

Home position coordinate

The reference point coordinate is the value of the reference point where the axis is located after synchronization and traveling of the reference point offset.

Home position setting

→ Direct homing, Types of homing

Homing absolute encoder

The zero point of the absolute encoder is adjusted. (absolute encoder adjustment)

→ Types of homing

HTL

High Threshold Logic

Logic with high fault threshold

A form of encoder signal. Incremental encoders are available as TTL and HTL encoders.

Human Machine Interface

Interface and functionality for operator control and monitoring of SIMOTION. These can be SIMATIC HMI devices or open solutions based on OPC.

→ HMI

HW

Hardware

HW Config

SIMOTION module where the hardware configuration can be adjusted.

I

ID

Abbreviation of identifier, designates a unique coding of a product.

IM

Interface module

IM174

The IM174 interface module (interface module for 4 axes) can be used with SIMOTION to operate up to 4 drives with an analog setpoint interface on the isochronous PROFIBUS DP. These can be electric or hydraulic drives.

In addition, IM174 can be used to connect stepper drives with a pulse/direction interface or as an external encoder for SIMOTION.

Incremental encoder

An incremental encoder (incremental position encoder) is a pulse encoder (square-wave encoder, sin/cos encoder). Homing is required to determine the absolute position.

Instance

Individually named copy of a data structure, which is linked to a function block type or program type and which is retained from one call of the associated operations to the next call (memory).
An instance of the function block is created in order to use the function block.

→ Function block

Interconnection interface

Interface for bi-directional data exchange between technology objects. Here, it is possible for different data to be transmitted in different operating phases. Interconnection interfaces in SIMOTION are not open interfaces but rather interfaces of the system SW or technology SW.

Interconnection interfaces between technology objects are created or selected explicitly during configuration.

Exception: explicit assignment such as synchronous object, output cam, measuring input for TO Axis.

Interface

Agreement on how information is to be interpreted or transferred. During data exchange, the sender must supply the data correctly and the receiver must be able to interpret it.

→ OPC

Interface section

The interface section contains instructions for importing and exporting data (data types, variables, function blocks, functions, and programs). Technology packages and libraries can also be downloaded.

→ Global user variable

Interpolation

Procedure for determining interim variables of measured values or traversing profiles, for example, or interim variables when converting setpoints.

Interpolator

The interpolator uses interpolation to perform motion control for a predefined traversing profile. The generated setpoints are fed to the servo. If the position control cycle clock is shorter than the interpolator cycle clock, the setpoints undergo fine interpolation in the servo.

→ Interpolation, Interpolator cycle clock, Position control cycle clock

Interpolator cycle clock

Cycle clock in which the interpolator is called.

→ Interpolator, Cycle clock, IPO cycle

Interrupt

Interrupts are events that trigger an InterruptTask. In SIMOTION, a distinction is made between a SystemInterrupt and a UserInterrupt.

→ InterruptTask, SystemInterrupt, UserInterrupt

InterruptTask

Is started when an interrupt occurs. All programs contained in the InterruptTask are executed.

In SIMOTION, a distinction is made between a SystemInterrupt and a UserInterrupt.

→ Execution level, SystemInterrupt, UserInterrupt

I/O

Input/Output

I/O errors

Fault signaled by I/O modules.

→ PeripheralFaultTask

I/O variable

→ Global user variable

I/O variable (process image)

Access to process image variable via an icon.

IPO

→ Interpolator

IPO cycle

→ Interpolator cycle clock

IPOSynchronousTask

The IPOSynchronousTask is activated in the interpolator cycle clock and runs in the cycle before the interpolator.

→ ServoSynchronousTask

IRT

Isochronous Real Time Ethernet

→ PROFINET IO with IRT

IRT High Flexibility

→ PROFINET IO with IRT

IRT High Performance

→ PROFINET IO with IRT

Isochronous execution level

Must be executed within a fixed predefined period, e.g. linked to IPO cycle clock.

→ Execution level

Isochronous mode

→ Equidistance mode

IT

Information technology

Generic term for information and data processing.

K

Kernel

The kernel contains the basic functionality of the system (e.g. communication, programmability, execution levels, task system, etc.) and essentially corresponds to a PLC functionality with a command set according to IEC 61131-3.

Every SIMOTION platform has its own kernel - the term "firmware" is also used for this. On a SIMOTION D, for example, the kernel is located on a CompactFlash Card.

The SIMOTION Kernel can be expanded by loading technology packages (TPs), e.g. for motion control or temperature controllers. These technology packages are first loaded onto the CompactFlash Card using a Project Download.

→ Device level, Execution level, Technology package, Firmware

Kinematic offset

Specifies the offset from the axis zero points (transformation zero point) relative to the path zero point for path interpolation.

Kinematics

The term "kinematics" in the context of robots and handling devices in motion control systems refers to the abstraction of a mechanical system onto the variables relevant for motion and motion control, i.e. the motion-capable elements (articulations) and their geometric positions relative to each other (arms).

During forward calculation of the kinematics (including direct kinematics, forward kinematics or forward transformation) for position and motion conversion, the position of the end point of the kinematics is determined in the basic coordinate system from the position of the articulation angle and its spatial arrangement.

During backward calculation (including backward transformation or inverse kinematics), the position of the individual articulation angle is determined from the position of the end point of the kinematics in the basic coordinate system.

For path interpolation, the position of the end point of the kinematics in the basic coordinate system is calculated over time.

→ Path interpolation, Basic coordinate system

Kinematics transformation

Conversion of specifications in Cartesian coordinates to specifications for the individual path axes, and vice versa.

KTY

Special temperature sensor

Kv factor

Gain factor in the position control loop.

→ Servo

L

LAD

→ Ladder diagram

Ladder diagram

Ladder diagram (LAD) is a graphics-based programming language. The instruction syntax corresponds to a circuit diagram. The LAD instructions consist of elements and boxes that are graphically connected to networks (which are displayed according to the IEC 61131-3 standard). LAD operations follow the rules of Boolean logic.

→ Function Block Diagram

LAD/FBD

Ladder diagram/function block diagram

→ Ladder diagram, Function Block Diagram

Leading axis

In the synchronous operation function, the leading axis is the axis that specifies the master value for the synchronous object in the following axis.
The leading axis is also referred to as the master axis.

→ Following axis, Synchronous operation

Leading synchronization

In leading synchronization, the synchronization movement is performed before the synchronization position is reached.

→ Trailing synchronization, Synchronization position (synchronous operation)

LEDs

Light emitting diode

Library

A library is a collection of reusable templates.

→ Vertical application, Library element

Library element

Library elements include user-defined data types, unit variables, functions, and function blocks and can be used by any SIMOTION device.

Library elements can be written in all available programming languages and can be called as functional units of user programs in all programming languages.

→ User-defined data types, Unit variable, Function, Function block

License package

Package that contains several licenses; collection of single licenses for increased user-friendliness (selection, ordering, delivery).

Linear axis

Axis with a linear (translatory) traversing range. User unit on the axis is a unit of length (e.g. mm, inch); support during configuration in accordance with the kinematic characteristics of a linear axis.

Linear path

2D- or 3D-path that describes a straight line.

Local measuring

Encoders and measuring inputs are connected to a common electronic measuring device, e.g. to C2xx (M1, M2) or to the drive; when the measuring signal occurs, the actual value of the actual value acquisition block is saved.
Local measuring inputs are axis-based.

→ Global measuring

Local user variable

Local user variables are defined within function blocks, functions, and programs. They are only available locally, i.e. they cannot be viewed or controlled either in the symbol browser or in the HMI/OPC. Local user variables can be monitored via the "Program status" function.

→ Variable, Global user variable, Symbol browser

M

Machine cycle

Execution cycle of a production machine.

→ Cycle clock

Main plane

X-Y, Y-Z or Z-X plane of the path interpolation or a plane parallel to this plane. The third coordinate is not evaluated.

→ Basic coordinate system

Manipulated variable

Value calculated by the controller from the difference between the actual value and the setpoint, which is then output to the final controlling element (as a % of the controller cycle time).
The temperature controller generates a digital, pulse-width-modulated signal (PWM).

Manual manipulated variable (temperature channel)

A manipulated variable predefined by the HMI that is output in manual mode in place of the manipulated variable calculated by the controller (the controller has no effect).

Manual output mode (temperature channel)

The manipulated variable is not calculated by the controller but is specified directly, either by entering it on the HMI or through a preset value in the control (manual manipulated variable). The control algorithm is switched off in manual mode.

Marshalling

Conversion of standard data types and user-defined data types into byte arrays and vice versa.

Master application cycle

The reduction ratio between the bus and position control cycle clocks must also be set on the drive as the master application cycle. This setting is necessary to enable reciprocal sign-of-life monitoring.

Master axis

→ Leading axis

MASTERDRIVES

Siemens product line of drives

Master object

Technology object in a synchronized group that provides a master value.

→ Master value

Master value

A master value is the input variable for synchronous operation (gearing, velocity gearing, camming).

For example, a master value can be the setpoint or actual value of a leading axis or the actual value of an external encoder.

→ Following axis, Synchronous operation

MC

Motion Control

MCC

→ Motion Control Chart

MCC module

Defines a combination of command blocks in the MCC, thus enabling motion sequences to be clearly structured.

Measuring

Determination of a process variable (actual position value) at the time a process event occurs (e.g. when a measuring input switches).

→ TO measuringInput, Local measuring, Global measuring

Memory Card

The term "memory card" refers to a memory card in general. This is the Compact-Flash Card for the SIMOTION D4xx and the Micro Memory Card for the SIMOTION C. On the SIMOTION P350, the memory card is displayed on the hard disk in the form of a virtual memory card.

MMC

Micro memory card

→ Memory Card

Module

→ MCC module

Modulo axis

Axis whose entire traversing range is not limited.
The unlimited complete traversing range is divided into defined, cyclically repeating continuous ranges (modulo ranges).
A modulo range is defined by the start point (the first one can be specified; the others are defined by the follow-up conditions) and the modulo length. A modulo axis may be a type of rotary axis or a linear axis.

→ Rotary axis, Linear axis

Motion Control Chart

MCC

Graphical programming language for programming logic and motion control according to the flow-chart principle.

→ Structured Text

MotionTask

MotionTasks are assigned to the round-robin execution level. They are used for the programming of sequential commands. They are not time-monitored. Several MotionTasks can be executed in parallel.

→ Round-robin execution level

MRES

Memory reset

→ Overall reset

Multiple clocking

→ Clocking, multiple

N

Non-cyclic application

Non-cyclic application means that the function, e.g. a cam, is used just once.

Non-volatile data

Non-volatile data is used to retain user- and system-relevant data, even when the machine is switched off, e.g. battery buffered. The data becomes effective automatically after the machine has ramped up.

NVRAM

Non-volatile random access memory

Retain memory (power failure-proof)

O

OEM

The OEM (Original Equipment Manufacturer) uses the user interface of the system to create their applications.

Offset

Offset of the input and/or output variables for synchronous operation, fixed gear and for the cam.

OLP

Optical Link Plug

OM

Operating mode

On-the-fly homing

→ Passive homing

OPC

A standardized software interface that enables data exchange between applications from different manufacturers.
OPC offers standardized access to SIMATIC S7 and SIMOTION for OPC-capable applications on Windows XP.

OPC allows the integration of automation products from various manufacturers.

→ Interface, Openness, Productivity and Collaboration

OPC XML DA

Data access based on XML. The OPC server is integrated into the SIMOTION device and is addressed by a partner station via web services and their XML-coded function calls. The partner stations are therefore independent of hardware and operating systems.

Openness, Productivity and Collaboration

→ OPC

Operating setpoint (temperature channel)

Temperature setpoint for the temperature controller during operation.

→ Setpoint

Output cam

An output cam generates position-dependent switching signals depending on the position values of the axes or an external encoder. A distinction is made between position-based cams, time-based cams and uni-directional output cams.

An output cam is defined by the following:

- Start position and end position (position-based cams)
- Start position or ON period (time-based cams)
- Start position (uni-directional output cams).

→ TO outputCam, Technology object, Cam track

Overall reset

When an overall reset is performed, the data is deleted with the exception of the communication configuration (baud rates, network addresses, etc.). The data on the memory card is retained during the overall reset.

P

Passive homing

This type of homing occurs during motion that was not initiated by a homing command.
Passive homing is also referred to as on-the-fly homing.
The following homing modes can be configured:

- Homing with homing output cam and encoder zero mark
- Homing with external zero mark only
- Homing with encoder zero mark only

→ Types of homing, Reference cam, External zero mark, Encoder zero mark

Path axis

→ TO pathAxis

Path axis offset

During path interpolation, the path axis offset specifies the constant offset of the zero position of the transformation in the direction of the zero position of the single axis.

Path interpolation

Motion along a path with a parameterizable dynamic response.

Path interpolation generates the traversing profile for the path, calculates the path interpolation points in the IPO cycle, and, based on this information, derives the axis setpoints for the IPO cycle time instants using the kinematics transformation.

Path interpolation grouping

Multiple path and position axes connected through a path object or an interpolation.

Path object

The path object provides the functionality for the path interpolation and for other tasks connected with the path interpolation. It also contains the kinematics transformations implemented in the system.

→ TO pathObject

Path-synchronous motion

Motion that occurs when an axis motion is coupled with a path motion; this motion is output to a position axis.

PB

PROFIBUS

→ PROFIBUS DP

PELV

Protective Extra Low Voltage

PeripheralFaultTask

Task that is triggered when process interrupts and diagnostic interrupts occur. It is used to program the response to such events.

→ SystemInterrupt, I/O errors

Persistent data

Non-volatile data in the device or user memory on the memory card

→ Non-volatile data

PG/PC

Programming device/personal computer

PIB

Proxy Ident Block

Standardized function blocks are available in the SCOUT command library for data exchange between the SIMOTION system and standard profile RFID systems (PIB).
The function blocks (FB) act as the communication interface between a standard profile RFID system (e.g. ASM 456) and the user program.
The FBs support functions such as configuration, command processing, reading/writing data, and diagnostics.

→ RFID, Command library, Function block

PIV

Parameter identifier value

Plausibility check (TControl only)

Based on the property of the controlled system, actual value changes can only be made within a predictable range (actual value plausibility); on the other hand, setpoint jumps must be followed by a change in the actual value within a specified period (control circuit plausibility).
This is monitored by the plausibility check.

PLC

Programmable logic controller (PLC)

PN

→ PROFINET

PNO

PROFIBUS user organization

PNO guideline

→ PROFIBUS DP-V2, PNO, PROFIBUS DP, PROFIdrive profile, PROFINET

Polynomial path

2D- or 3D-path that describes a polynomial segment.

Position control cycle clock

Inputs and outputs are updated in the position control cycle clock.
Position control for the axes and the processing of the centralized and distributed I/O are carried out in the position control cycle clock. The position control cycle clock can be operated relative to the bus cycle clock at a ratio of 1:1 or 1:2.

→ Servo

POU

→ Program organization unit

PQ valve

Special Q valve suitable for force, position, velocity, and pressure control.

→ Q valve

Pragma

A construct in a programming language to enter text in a source program (e.g. instructions), which influences its compilation. In the ST (Structured Text) programming language, these may be, for example, instructions for the preprocessor or attributes to control the compiler.

Precontrol

Specifies the manipulated variables for velocity and, if applicable, torque directly from the completed motion control or interpolation on the servo with additive switch-in into the lower-level drive controller(s).
When the dynamic response of the controlled axis system is known, precontrol can be applied to achieve a better dynamic response and the lower-level controllers can be set essentially for disturbance compensation.

Preprocessor

The preprocessor prepares a source program for compilation. For example, character strings can be defined as replacement texts for identifiers, or sections of the source program can be shown/hidden for compilation.

Pressure cut-off valve

→ P valve

Pressure-relief device

→ P valve

Process fault diagnostics

Messages can be issued and acknowledged from the user program. These are displayed on the OP. The mechanism is established in the existing SIMATIC S7/OP environment (ALARM_S).

→ Alarm_S

Process interrupt

→ Alarm, SystemInterrupt

PROFIBUS DP

Process Field Bus - Decentral Periphery

PROFIBUS DP is the PROFIBUS protocol format used for the connection of the distributed I/O (D IO). It is designed for rapid, cyclic data exchange in the sector. Functions are specified by the DP basis functions (DP-V0).

PROFIBUS DP-V0

The version DP-V0 describes the basic functionality of the PROFIBUS DP. This includes cyclic data exchange, as well as the configuration, parameterization, and diagnostics of a PROFIBUS station.

PROFIBUS DP-V1

The additional DP function expansions (DP-V1) make it possible to perform non-isochronous read and write functions and to transfer detailed diagnostic information as well as to process cyclic data communication.

Profibus user organization (PNO) extension to PROFIBUS DP in order to carry out acyclic communication.

PROFIBUS DP-V2

The equidistant PROFIBUS DP is an extension of the PROFIBUS DP. The extensions comprise the equidistance mode and data exchange broadcast functions.

Isochronous operation enables "isochronous data transmission". The data exchange broadcast enables a direct exchange of data between slave devices with no involvement from the master.

The new functions are specified in the PROFIBUS profile "Drive Technology", Version 3 from the PROFIBUS user organization (PNO) and are currently being integrated into Part 2 of the PROFIBUS standard EN 50170.

To use the new functions, you need a suitable isochronous master interface. The equidistant PROFIBUS DP is upwards compatible: In a mixed configuration, equidistant slaves and non-equidistant slaves (e.g. ET 200 intelligent terminal strip) can be operated together on an equidistant PROFIBUS DP bus line without problems.

→ Equidistance mode, Data exchange broadcast

PROFIdrive

PROFIBUS profile specified for speed- and position-controlled drives by the PROFIBUS user organization (German: PNO).

PROFIdrive profile

Guideline for connection of drive units to a controller via a bus system.

→ PROFIBUS DP, PROFINET IO

PROFINET

Open communication standard based on standard Ethernet (IEC 61158) for industrial communication in automation.
With PROFINET, devices can be linked from the field level up to the management level. PROFINET enables real-time communication and Ethernet-compliant communication to take place simultaneously on one cable.

PROFINET IO

Process data communication based on PROFINET, i.e. cyclic message frame exchange for input and output data between a controller and the associated distributed I/O.
A distinction must be made here between two real-time classes:

- RT (Real Time)
- IRT (Isochronous Real Time)

→ PROFINET IO with IRT

PROFINET IO controller

Controller within an IO system that collects data from the distributed IO devices (inputs) and writes data back to the outputs. The control program runs, the alarms are handled, and the bus ramp-up is controlled in the IO controller.
As a result, in the broadest sense the IO controller is the same as the Class 1 master in PROFIBUS.

PROFINET IO device

Distributed field device that exchanges data with an IO controller.
IO device is thus the same as a slave in PROFIBUS.

→ PROFINET IO controller

PROFINET IO Supervisor

Device that does not perform cyclic process data communication with controllers or devices. These devices are mainly used for diagnostics, commissioning or display purposes and, thus, correspond to a Class 2 master in PROFIBUS.

PROFINET IO with IRT

The process data message frames are sent cyclically in a deterministic cycle (Isochronous Real Time). A send interval is divided into an IRT interval, when real-time message frames are transmitted, and another interval, when standard Ethernet message frames are transmitted.
The bandwidth for the IRT interval is reserved by hardware mechanisms, so that the IRT message frames always remain uninfluenced by all other Ethernet communication taking place via the cable.

PROFINET with IRT can be operated in one of the two following ways:

- IRT High Flexibility

  Supports the above properties with a focus on the flexible interconnection of devices, without having to define a network topology via configuration.

- IRT High Performance

  An extension to IRT High Flexibility, message frame traffic can be optimized by defining a network topology during configuration.
  This enhances performance during data exchange and improves deterministic behavior.

PROFINET IO with IRT communication

→ PROFINET IO with IRT

PROFINET IO with RT

The real-time property of the message frame exchange is realized by prioritizing message frames in the switches of the Ethernet network.

Program

A program describes a self-contained process.
It can contain motion commands and logic. Functions and function blocks can be called from programs. Likewise, variables can be accessed from programs. Each program must be assigned to a task.

→ Function, Function block, Project data

Program execution error

Error while executing a user program.

→ Fault task, ExecutionFaultTask

Programming

→ Sequential programming, Cyclic programming

Programming languages

→ Motion Control Chart, Structured Text, Ladder diagram, Function Block Diagram, Drive Control Chart

Program organization unit

A program organization unit (POU) can be a program, a function or a function block.

Program simulation

The setpoints are calculated according to the programming, but are not output on the I/O or final controlling element. I/O devices or drives must be connected.

→ Simulation mode

Project

A project is the sum of all the data to be defined by the user, such as technology objects, programs, settings, etc., that are specified in SIMOTION SCOUT. It is defined uniquely with a name and represents the highest hierarchy of data that are to be managed consistently.
The scope of a project can involve one or more SIMOTION devices. A project can consist of one or more SIMOTION stations (SIMOTION devices).

→ Project data

Project data

General term for data that is stored in a project either explicitly by the user (e.g. programs, parameters, etc.) or implicitly by the system (configuration).

→ Project, User, Program

Project comparison

Project comparison is a function in the SIMOTION SCOUT engineering tool. Objects can be compared both online and offline using this function.

Project navigator

Device-oriented project structure that displays all project contents (devices, axes, output cams, programs, etc.) and allows them to be handled, e.g. creation, configuration, programming, etc.

→ Workbench

PS

Power supply

PTC

Positive temperature coefficient

Pulse width modulation

PWM

The actuating signal is output digitally as a pulse train of ON and OFF pulses.

P valve

Valve used for limiting the system pressure, suitable for protecting a hydraulic system against high pressure.

(Pressure cut-off valve, pressure-relief device)

PWM

→ Pulse width modulation

PZD

Process data in the PROFIdrive profile

→ PROFIdrive profile

Q

Q valve

Hydraulic valve for control of direction and quantity of a volume flow rate, suitable for velocity control.

(valve for closed-loop control of a volume flow rate, way valve)

R

RAM

After the controller ramps up, the RAM contains the code and the data of the technology objects (technology packages and data of the instanced technology objects) and the executable program units.

RAM disk

After the download, the RAM disk contains the hardware and device configuration, technology packages, configuration data of the technology objects, and the program units.
By copying RAM to ROM, the contents of the RAM disk are copied to the memory card and the memory on the RAM disk is cleared. During subsequent downloads, only modified data is loaded to the RAM disk.

The RAM disk or the memory card also contains the user data that were created from the user program using file system functions.
Copying RAM to ROM affects only the project data from the download.

Reference cam

The reference cam outputs an enabling signal for the actual reference signal (encoder zero mark or external zero mark).
The drive reduces the velocity on the basis of the switching edge returned by the reference cam and waits for the next incoming reference signal in order to carry out the homing.

→ Reference mark, Encoder zero mark, External zero mark

Reference mark

A reference mark is a hardware signal that is used for homing.

→ Encoder zero mark, External zero mark

Reference point

When the axis has been synchronized and the reference point offset has been traveled, the axis is at the reference point and has the value specified in the reference point coordinate.

→ Reference point offset

Reference point offset

Offset between the reference point and the synchronization point; it is only effective during active homing. It is applied after the axis is synchronized via the homing command.
In modulo axes, the reference point offset is always applied with the Shortest_Way direction setting.

→ Reference point, Synchronization point (homing), Active homing

Relative direct homing

The actual position value of the axis is offset by a preset amount without traversing having taken place.

→ Types of homing

Resolver

The resolver is a rotary transducer with very robust mechanical and electrical properties.
It comes in versions with different numbers of pole pairs.

A phase offset of n* 360 degrees per rotor revolution is produced in the output signal, where n is the number of pole pairs (a resolver can be used as a sin/cos incremental encoder with n encoder pulses per revolution).

A two-pole (or single pole pair) resolver can also be used as a single-turn absolute encoder. Resolvers with multiple pole pairs are usually used as motor encoders, with the number of pole pairs corresponding to that of the motor (usually 3).

Response threshold (temperature channel)

Heating and cooling equipment is controlled by means of digital contact blocks; very short switching pulses (ON and OFF pulses) place a burden on contact elements of this type. For this reason, it is possible to specify a so-called response threshold, which defines the minimum permissible magnitude of a manipulated variable before it can be output to the contact element.

The manipulated variable calculated by the controller is converted to an appropriate ON period for each controller sampling time using pulse width modulation: the response threshold prevents the generation of very small ON or OFF pulses.

Retentive data

Non-volatile data in the device (declared with VAR_GLOBAL_RETAIN)

→ Non-volatile data

Reversing cam

In the case of the reversing cam, the effective direction in relation to the switch-off position is changed after reversing the direction of motion of the axis. The switch-off position can be in front of or behind the switch-on position.

RFID

Radio Frequency Identification

RFID-based identification systems enable data exchange between a mobile data memory and the write/read device.
Data exchange is fully automatic and contact-free via radio frequency (RF), and no direct visual contact is required.

Rotary axis

Axis with rotary traversing range; user unit on the axis is a rotary unit (e.g. °, degrees); support during configuration in accordance with the kinematic properties of a rotary axis.

Round-robin execution level

The BackgroundTask and the MotionTasks are executed in the round-robin execution level. The time allocation for the BackgroundTask and the MotionTasks can be set as a percentage.

→ Execution level, BackgroundTask, MotionTask

RT

Real Time Ethernet (non-isochronous PROFINET)

→ PROFINET IO with RT

Runtime system

The SIMOTION RT comprises the SIMOTION Kernel and the technology packages.

→ SIMOTION RT, SIMOTION Kernel, Technology package

S

Safety Integrated

Safety Integrated is the term given by Siemens to a harmonized product line that can be used in automation and drive systems to implement innovative solutions for plant and system safety.
Safety Integrated is an integral component of standard automation.

The individual components can communicate via standard buses, thereby offering innovative safety technology whose flexibility, diagnostics capability and standardization are equal to that of standard automation components.

For example, the Safety Integrated product line includes safety-oriented low-voltage controls and sensors, fail-safe PLCs, computerized numerical control, and variable-speed drives with integrated safety functions.

Safety technology

→ Safety Integrated

Sampling controller

Controller that acquires the current value of a process variable at constant intervals (sampling time) and uses it to calculate a new value for the manipulated variable.

Sampling time

Time interval between two equidistant activations of a function, e.g. of a sampling controller.

SBC

Safe Brake Control

Scaling

Multiplication of the input or output variables by a factor for synchronous operation and for the cam.

SCOUT

SIMOTION Controlling with Optimized Usability Toolbox

→ SIMOTION SCOUT

SDB

→ System data block

Self tuning (TControl only)

The controller parameters are automatically determined by an integrated process (analysis of the step response).

Sending cycle

This is the period between two successive intervals for IRT or RT communication on PROFINET.

The sending cycle is the shortest possible transmit interval for exchanging data. The calculated/configured update times are always a multiple of the sending cycle. The sending cycle therefore corresponds to the shortest possible update time. Within this time, IRT data and non-IRT data (CBA, TCP/IP) is transmitted. All devices within a sync domain work with the same sending cycle.

→ Update time, Bus cycle clock

Sensor

→ TO Sensor

Sequential programming

Synchronous command execution The technology function is transferred to the technology object and the calling task is stopped.
The technology object executes the function and calls for program execution to be resumed as soon as the specified step enabling condition is satisfied or the command has been aborted.

This method of program execution is referred to as *sequential programming*. It is supported particularly by the MotionTasks.

→ Technology function, Technology object, MotionTask, Cyclic programming

Servo

Designates a module that contains the position controller, actual value system, and setpoint system of the axis and performs monitoring activities on the axis.

The servo cyclically compares the internal digital position setpoint with the digital actual value of the position encoder. It calculates the speed setpoint according to a control rule.

→ Servo drive, Servo axis, Servo motor, Position control cycle clock, Kv factor

Servo axis

High-performance electrical axis (with respect to dynamic response and accuracy), velocity- and position-controlled.

→ Servo

Servo drive

Designates a high-performance drive (with respect to dynamic response and accuracy)

→ Servo

Servo motor

Designates a high-performance motor (with respect to dynamic response and accuracy)

→ Servo

ServoSynchronousTask

ServoSynchronousTasks are available for applications where processes need to be performed quickly using isochronous mode (as of V4.0).
They run within a position control cycle clock and behave in a similar way to IPO-SynchronousTasks.

→ IPOSynchronousTask

Setpoint

Temperature value to be maintained by the temperature controller; the operating setpoint and reduced setpoint are generally set.

→ Operating setpoint (temperature channel)

SF

System Failed

SFC

System functions represent access to the functionality of the SIMOTION Kernel and technology packages or objects. They can be used in the user program.

→ Command library

Shutdown task

The task is executed once at the transition from RUN to STOP, i.e. all the programs contained therein are executed.

→ Execution level, ShutdownTask

ShutdownTask

Task that is triggered during the system transition to STOP. It is used for programming the response required here.

→ Execution level, SystemInterrupt

SIMATIC Manager

Graphical user interface for SIMATIC users on Windows.
HMI devices and drives are also integrated in the SIMATIC Manager under Totally Integrated Automation (TIA).
The SIMOTION system can be integrated seamlessly in the SIMATIC Manager or operated on its own.

SIMODRIVE

Siemens product line of drives

SIMOTION

Siemens product line for optimal automation of production machines

- With emphasis on motion control and technology
- Combination of logic and motion
- Cross-device global project programming
- Target system-neutral architecture
- Engineering (SIMATIC-compliant)
- Expanded to include graphic programming
- Optimized for motion sequences
- Open system approach
- Expandable to include (industry sector-) specific technology functions, even by OEMs

SIMOTION C

Standalone motion controller, SIMOTION automation device based on the SIMATIC S7-300 mounting technology.

→ SIMOTION hardware platform

SIMOTION CamTool

Easy-to-use cam editor for graphical entry of motion profiles. Optional package for SIMOTION SCOUT

SIMOTION complete system

The computer unit and panel front together form a complete SIMOTION system.

SIMOTION D

SIMOTION D is a drive-based version of SIMOTION based on the SINAMICS S120 drive range.

On SIMOTION D, the SIMOTION PLC and Motion Control functionalities run together on control hardware, along with the SINAMICS S120 drive runtime software. The SIMOTION D is available in two designs, as a single-axis system (SIMOTION D410) and as a multi-axis system (SIMOTION D4x5), and also in different performance versions (D425/D435/D445).

→ SIMOTION hardware platform

SIMOTION hardware platform

Hardware on which a SIMOTION system can be executed, e.g.:

- SIMOTION C
- SIMOTION P350
- SIMOTION D4xx

SIMOTION Kernel

SIMOTION Kernel is the name given to the basic functionality of the SIMOTION RT without technology packages.

→ Technology package, Device level, SIMOTION RT, Kernel

SIMOTION MMC

SIMOTION micro memory card

SIMOTION P

Designates a PC-based SIMOTION device. Includes the computing unit and the operating unit. Consists of the SIMOTION P350 hardware including the panel, conditioner card (PROFIBUS and/or PROFINET) and SIMOTION Kernel software that runs in real-time on a Windows operating system on this PC.
The system is supplied ready for operation.

→ SIMOTION hardware platform

SIMOTION P350

SIMOTION P hardware platform

→ SIMOTION hardware platform

SIMOTION P Control Manager

Component anchored in the Control Panel of Windows XP.
Enables the setting of basic information of SIMOTION P (e.g. runtime allocation or SIMOTION P and Windows XP, memory available in SIMOTION P, etc.).

SIMOTION P Startup

SIMOTION P Diagnostics application contains the following:

- LEDs
- HW switch
- Management of miniature flash cards
- Detection of real time violations

SIMOTION P state

Application for the display of operating states of SIMOTION P.
Enables their manipulation (HW keyswitch, ramp-up information, system fault LED, etc.).

SIMOTION RT

The SIMOTION RT comprises the components:

→ SIMOTION Kernel

→ Technology packagee

SIMOTION SCOUT

SIMOTION SCOUT is the engineering system of the SIMOTION product family, which is integrated in STEP 7.

SIMOTION SCOUT provides all the necessary tools for the following functionalities:

- Configuration
- Parameterization
- Programming
- Test
- Diagnostics.

The following tasks are graphically supported with operator prompting:

- Creation of the hardware and network configuration
- Creation, configuration, and parameterization of technology objects such as axes, output cams, and cams.

SIMOTION station

Organization unit for the hardware configuration in SIMOTION SCOUT.

Simulation mode

In SIMOTION, simulation mode is divided into:

→ Program simulation

→ Axis simulation

SINAMICS

Siemens product line of drives

SINAMICS G

Siemens product line of drives

SINAMICS Integrated

SINAMICS Integrated designates the SINAMICS S120 drive integrated into the SIMOTION D control unit. The SINAMICS S120 drive runtime software runs on a module, together with the classic SIMOTION functionalities (PLC and Motion Control functions). Therefore, the complete system (consisting of the open-loop control and the drive) is extremely compact and responds very quickly.

The SIMOTION D integrated drive control has the same control properties and performance features as the SINAMICS S120 control units, with few exceptions. (Exceptions: no basic positioner, for example)

SINAMICS S

Siemens product line of drives

SITOP

Power supply

Slave axis

→ Following axis

SLS

Safely Limited Speed

SM

Signal module of the SIMATIC S7 product family (e.g. I/O module).

SMC

Sensor Module Cabinet, encoder connection to DRIVE-CLiQ

For motor encoder and temperature evaluation of motors without DRIVE-CLiQ or when additional encoders are used (for example, machine encoders).

SOS

Safe Operational Stop

Speed controller

A speed control is generally a device that adjusts the speed of an axis to a constant value. Depending on the situation at hand, the control intervention occurs either at the torque-generating location (motor) or through targeted application of a braking torque.

SS1

Safe Stop 1 (SS1)

SS2

Safe Stop 2 (SS2)

SSM

Safe Speed Monitor (SSM)

ST

→ Structured Text

Standard function

Standard functions are function blocks or programs that provide standard and basic functionality (e.g. homing mode, simple print mark, etc.) not provided by the technology commands.

→ Function block, Program, Library

STARTER

Functionally integrated in SIMOTION SCOUT for commissioning of digital drives of the SINAMICS line.

Startup characteristic (temperature channel)

A hot runner requires controlled heating. This is ensured by a so-called startup characteristic that controls the setpoint according to time and/or the current temperature value.

StartupTask

Task that is triggered at the system startup and during the transition from STOP to RUN. It is used to program the response, especially initialization, to this transition.

→ System startup, Execution level

Station

Device that can be connected to one or more subnets as a consistent unit, e.g. SIMATIC, SIMOTION device, programming device, operator panel. A project can consist of one or more stations.

ST editor

Editor for writing programs in ST.

→ Structured Text

STO

Safe Torque Off

Structured Text

Text-based high-level language for SIMOTION, which is IEC 61131-3-compliant and has been extended with motion control and other language commands. These are integrated as functions or function blocks.

→ Motion Control Chart

ST source file

An ST source file is a logical unit, which is created in the SIMOTION project and can be present several times (corresponds to a unit of the program).

→ Unit

Superimposed synchronous operation

In superimposed synchronous operation, two synchronous objects can be interconnected with one slave axis. These two synchronous operations are superimposed on one another.

SW

Software

Symbol browser

Functionality integrated into SIMOTION SCOUT for the creation and visualization of user variables which are global to a project and for the visualization of system variables. The symbol browser is integrated into the detail view of the workbench.

→ Variable, Detail view

Synchronization point (homing)

In the synchronization point, the actual value of the axis is set to the value home position coordinate minus home position offset, due to an external or internal event.

Synchronization position (synchronous operation)

The synchronization position is entered explicitly or implicitly in a synchronous command (current position at immediate synchronization).

The command can also specify whether synchronization should occur before the synchronization position (leading synchronization), symmetrically with the synchronization position (only when synchronizing over master value lengths), or after the synchronization position (trailing synchronization).

Synchronized group

A synchronized group consists of at least one master object (technology object) that supplies the master value (master) of, for example, a position axis or an external encoder and a following axis.

Synchronous command execution

→ Sequential programming

Synchronous exception

Refers directly to a faulty programming command (e.g. division by 0).

→ Asynchronous exception, ExecutionFaultTask

Synchronous object

Synchronous operation functionality is made available via the synchronous object. It can be interconnected on the input side with a technology object that supplies a master value. On the output side, the synchronous object is permanently interconnected with a following axis.

Synchronous operation

Defined motion of a slave axis relative to a master axis. This can be linear (gear) or non-linear (cam).

→ Axis, Technology object, Technology package, Synchronous exception, Following axis

SynchronousTask

IPOsynchronousTask (IPO and IPO2), which is called in the interpolator cycle clock or an integer multiple of the interpolator cycle clock and executed once.

ServoSynchronousTask, which is called in the position control cycle clock and executed once.

→ Execution level

Synchronous velocity operation

Synchronous velocity operation is the synchronous operation of the velocities between two axes with a constant gear ratio.

System data block

System data blocks (SDB) are data containers for the storage of
HW configuration and network configuration information.
SDBs are generated during configuration of hardware and networks and are loaded to the SIMOTION RT.

SystemFaultTask

SystemInterruptTasks that are started for system alarms (Kernel events), e.g. communication overload, system overload. They are used for programming of the required responses.

→ SystemInterrupt

System function

System functions represent access to the functionality of the SIMOTION Kernel and technology packages or objects. They can be used in the user program.

SystemInterrupt

SystemInterrupts are defined in the system and start a SystemInterruptTask when the defined event occurs. Depending on the event, the appropriate System-InterruptTask is started and all the programs contained therein are executed. The execution level of the SystemInterrupts contains the following tasks:

- TimeFaultTask
- TechnologicalFaultTask
- PeripheralFaultTask
- TimeFaultBackgroundTask
- ExecutionFaultTask

→ UserInterrupt, BackgroundTask, Fault task

SystemInterruptTask

The SystemInterruptTasks are called when certain events occur (e.g. system fault) and executed once.

→ SystemInterrupt

System startup

During system startup, the StartupTask is called once and executed exactly once.

→ StartupTask

System variable

System variables are unique throughout a project thanks to specific identifiers. They have a default setting and can be changed during commissioning if required. The system variables can also be changed online from the user program or from the HMI.
SIMOTION distinguishes between the following system variables:

- System variables of the SIMOTION Kernel

  These variables are always available.

- System variables of technology objects

  System variables of technology objects are only available once the technology object is configured in the project.

→ Variable, Configuration data, Technology object

T

Target device

The target device is an individual SIMOTION device.

Target system

The target system is the automation system or automation component on which the (SIMOTION) program runs.
A target system can contain one or more target devices.

→ SIMOTION hardware platform

Task

A task provides an execution framework for programs.
It contains at least one program that, in turn, can call functions and function blocks. Tasks are assigned to an execution system.

→ Execution level, Function, Function block, Program

Task system

→ Execution system

TB

Terminal Board, connected to the option slot in the closed-loop control module and used for expansion (example: TB30 with 4 DI, 4 DO, 2 AI, 2 AO).

TCP/IP

Transmission Control Protocol/Internet Protocol

Connection-oriented network protocol; generally accepted standard for data exchange in heterogeneous networks.

Technological alarm

→ Technology alarm

TechnologicalFaultTask

Is started when technology alarms occur. This fault task is used for programming user responses.

→ SystemInterrupt, Fault task

Technology

Technology in SIMOTION describes the functionality implemented by the system and via the program, e.g. motion control on the axis, precise switching of outputs (output cams) supported by the system, precise measurement, temperature control and further complex controllers, programmed solutions for application segments, industrial sectors.

→ Technology object

Technology alarm

Technology alarms are generated by the technology objects in the SIMOTION RT. The quantity framework of the available technology alarms depends on the loaded technology packages and the technology objects contained therein.
A default system response is defined for each technological alarm. A System-Interrupt is triggered in the execution system when a technological alarm occurs. Alternatively, another response can be specified in the alarm configuration (e.g. SIMOTION device in STOP, stop axis).
Technology alarms can be displayed in SIMOTION SCOUT and on the OP (not in STEP 7).

→ Alarm, SystemInterrupt

Technology command

→ Technology object

Technology function

Technology functions are functions or function blocks that provide technological functionality, e.g. winder functions, and are based on the commands and variables of the SIMOTION Kernel and the technology objects.

Technology level

The technology level contains technology objects (TOs). These expand the basic functionality of the SIMOTION system. Technology objects are combined to form technology packages. These can be accessed by the user via technology commands.
The technology level can be omitted completely if desired, i.e. a SIMOTION Kernel can be operated without technology packages.

- SIMOTION devices are programmable and provide a basic set of commands in accordance with IEC 61131

This basic command set can be extended with technology packages:

- Technology packages contain technology objects that can be accessed from the user program via technology commands.

→ Technology object, Technology package

Technology object

A technology object (TO) maps technology onto SIMOTION.
For example, there are technology objects for various axis types:

- TO driveAxis
- TO posAxis
- TO followingAxis
- TO followingObject
- TO Cam

as well as for other technological units, such as:

- TO externalEncoder
- TO measuringInput
- TO outputCam
- TO camTrack
- TO additionObject
- TO fixedGear
- TO formulaObject
- TO controllerObject
- TO Sensor
- TO pathObject
- TO pathAxis
- TO temperatureController

Technology objects contain technology commands, system variables, technology alarms and configuration data. Technology objects are combined to form technology packages.

→ TO, Technology package

Technology package

A technology package (TP) combines technology functions that are required for machine construction automation in various industrial sectors.
The following standard technology packages are available for SIMOTION:

- TP Cam
  contains the basic technologies for motion control, such as drive axis, position axis, following axis, cam, output cam, cam track, and measuring input.
- TP Path
  also contains the path interpolation technology.
- TP Cam_ext
  also contains objects for the preparation of technological data on the system level, e.g. addition object, formula object.
- TControl
  contains the temperature controller technology.
- DCC
  contains interconnectable blocks for drive-related controller functions.

More sector-specific technology packages are also available as separate products.

→ Technology object

Temperature channel

→ TO temperatureController

TFT

Thin film transistor

T_i

Adjustable time in the PROFIBUS configuration for acquisition of actual values. For distributed drives in accordance with PROFIdrive, the current encoder values are stored (sampled) at time T_i before the equidistant bus cycle.

→ T_o, PROFIdrive

TIA

→ Totally Integrated Automation

TimeFaultBackgroundTask

Is started when a time-out occurs in the BackgroundTask. It is used for programming the response required here.

→ SystemInterrupt

TimeFaultTask

Is started when a time-out occurs in the TimerInterruptTask. It is used for programming the response required here.

→ SystemInterrupt

TimerInterruptTask

The TimerInterruptTasks are called in the configured time frame and executed once.

→ SystemInterrupt

TM

Terminal Modules

Used to expand the control unit to include, e.g. digital and analog inputs and outputs via DRIVE-CLiQ.

T_o

Adjustable time in the PROFIBUS configuration for output of setpoints.
For distributed drives in accordance with PROFIdrive, the velocity setpoint is applied in the drive controller at time T_o after the equidistant bus cycle.

→ T_i, PROFIdrive

TO

Abbreviation for technology object; only used when referring to the name of a technology object: e.g. TO followingObject

→ Technology object

TO additionObject

The TO additionObject provides the functionality for an output vector, which results from the addition of up to four input vectors.
The _AdditionObjectType data type is used to designate the addition object during programming.

→ Technology object

TO Axis

→ TO driveAxis, TO posAxis, TO

TO Cam

The TO Cam can be used to define a transmission function and apply it with other technology objects.

The TO Cam describes the functional relationship between master values (e.g. master axis) and slave values (e.g. slave axis).
With mechanical systems, this relationship is established via a cam. The cam function is defined according to VDI directive 2143, motion rule for cam gears.

In SIMOTION the cam can also be applied for mapping valve characteristics, velocity profiles, pressure profiles, etc.

The CamType data type is used to designate the cam during programming.

→ Technology object, CamEdit, CamTool

TO camTrack

TO camTrack:

- Generates position-dependent switching signals
- Can be assigned to positioning axes, following axes or external encoders
- Axes can be real or virtual

The _CamTrackType data type is used to designate the cam track during programming.

→ SIMOTION CamTool, Cam track, Technology object

TO controllerObject

The TO controllerObject provides the controller and precontrol functionality for scalar variables.

The _ControllerObjectType data type is used to designate the controller object during programming.

→ Technology object

TO driveAxis

The TO driveAxis provides the functionality for motion control via a speed specification without position control.
The DriveAxis data type is used to designate the drive axis during programming.

→ Drive Control Chart, Technology object

TO externalEncoder

The TO externalEncoder provides the functionality for a position encoder that is built onto the production machine to record a position and an angle.

In programming, the external encoder is designated with the data type External-EncoderType.

→ Technology object

TO fixedGear

The TO fixedGear provides the functionality for fixed gearing on the basis of a specifiable gear ratio.

The _FixedGearType data type is used to designate the fixed gear during programming.

→ Technology object

TO followingAxis

The TO followingAxis is a grouping of the following axis (slave) and synchronous object. The synchronous object provides the functionality of the master value coupling, synchronization and desynchronization of the synchronous operation, as well as gearing and camming.
The FollowingAxis data type is used to designate the following axis and the FollowingObjectType data type to designate the synchronous object during programming.

→ Technology object

TO followingObject

→ TO followingAxis, Technology object

TO formulaObject

The TO formulaObject provides the functionality for the use of formulas on scalar variables and motion factors. The _FormulaObjectType data type is used to designate the formula object during programming.

→ FPU, Technology object

Tolerance band (TControl only)

Actual values are monitored using two tolerance bands in order to be able to detect faults or errors during acquisition (sensor) or in the final controlling element.
During fault-free operation, the actual value must always lie within the tolerance band, or may only exceed it briefly (inner tolerance band).

If the input value lies within the tolerance band, a change is not forwarded to the output (no error response is triggered).

TO measuringInput

The TO measuringInput provides the functionality for fast, accurate measurement of actual positions. This is achieved through hardware support (e.g. measuring input on the associated drive unit), or through exact recording of the time when the measuring event occurred.
The MeasuringInputType data type is used to designate the measuring input during programming.

→ Technology object, Local measuring, Global measuring

TO outputCam

TO outputCam:

- Generates position-dependent switching signals
- Can be assigned to positioning axes, following axes or external encoders
- Axes can be real or virtual

The OutputCamType data type is used to designate the output cam during programming.

→ Technology object, Output cam

TO pathAxis

The TO pathAxis provides the functionality of an axis that can perform a path motion together with other path axes via a path object.
The _pathAxis data type is used to designate the path axis during programming.

→ Technology object

TO pathObject

The path interpolation provides the functionality for traversing a path using a parameterizable velocity profile.
The _PathObjectType data type is used to designate the path interpolation during programming.

→ Path interpolation, PeripheralFaultTask, Technology object

TO posAxis

The TO posAxis provides the functionality of the motion control with position control.
The PosAxis data type is used to designate the position axis during programming.

→ Technology object

TO Sensor

The TO Sensor provides the functionality for the acquisition of scalar measured values.

The _SensorType data type is used to designate the sensor in the programming.

→ Technology object

TO temperatureController

The TO temperatureController provides the functionality for temperature control.

The TemperatureControllerType data type is used to designate the temperature controller in the programming.

→ Technology object

Torque limiting

Limiting of motor torque on an axis.

Totally Integrated Automation

Totally Integrated Automation (TIA) ensures uniformity in terms of the following:

- Configuring
- Parameterizing
- Programming and
- Diagnostics data

And three-fold uniformity in terms of:

- Data management
- Communication
- User interfaces
- and cooperation of the necessary tools.

  This saves considerable effort for engineering and, thus, provides significant benefits to customers using SIMATIC or SIMOTION together with other TIA-capable system components.

TP

Abbreviation for technology package.

→ Technology package

Trace

Cyclic saving of process signals or user and system variables for visualization and data acquisition.

Trailing synchronization

In trailing synchronization, synchronization movements start when the synchronization position is reached.

→ Leading synchronization, Synchronization position (synchronous operation)

Travel to fixed stop

Travel to fixed stop is an NC function that enables forces to be exerted on mechanical components of the machine tool.

This function can be used to move a motor to a fixed stop at a specified torque/force without a fault being signaled. When the stop is reached, the parameterized torque/force is built up and maintained.

TSI

TaskStartInfo

TTL

Transistor-transistor logic

Types of homing

The following types of homing are supported by the SIMOTION controller:

→ Active homing

→ Passive homing/On-the-fly homing

→ Direct homing/Home position setting

→ Relative direct homing

→ Homing absolute encoder/Absolute encoder adjustment

U

UDP

User Datagram Protocol

Minimal, connectionless network protocol associated with the transport layer of the Internet protocol family. The purpose of UDP is to accord the proper application to data that are transferred via the Internet.

UDT

User Defined Type

→ User-defined data types

Unit

The term "unit" refers to an ST source file in the "Programs" container (project navigator). It can contain 1 to n programs, function blocks (FB), functions (FC) or user-defined data types (UDT).

→ Global user variable, Function block, Function, User-defined data types

Unit variable

→ Global user variable

Update time

Within this time interval an IO device/IO controller in the PROFINET IO system is supplied with new data by the IO controller/IO device.

The update time can be configured separately for each IO device and determines the interval at which data is sent from the IO controller to the IO device (outputs) as well as from the IO device to the IO controller (inputs).

→ Sending cycle

Upgrading

The upgrading of a SIMOTION device, e.g. via the Device Update tool.

User

Users are those who generate project data with the aid of SIMOTION SCOUT.

→ Project data

User-defined data types

User-defined data types (UDT) are used to define specific data types, which, in turn, can consist of one or more elementary data types or user-defined data types.

→ Data type, Elementary data types, UDT

UserInterrupt

UserInterrupts can be defined by the user through the configuration of events. The UserInterruptTask is started when an event occurs. Depending on the occurrence of a user-defined condition, one of the UserInterruptTask is started and all the programs contained therein are executed. The condition consists of a logic expression, which can contain variables, system variables and I/O.

→ SystemInterrupt, InterruptTask

UserInterruptTask

The UserInterruptTasks are called when a user-defined event (condition) occurs and are executed once.

User program

A SIMOTION user program solves a comprehensive, but self-contained task (logic, motion control, and technology).
This can be the control of a complete machine or machine component.
A user program comprises one or more programs and is created in the ST (text-based) or MCC, LAD/FBD, and DCC (graphics-based) programming languages.

→ System function, Technology package, ST, MCC, LAD/FBD, DCC

V

Valve characteristic

The valve characteristic indicates the flow rate versus the valve position.

Inverse to this, the valve position must be calculated and output in the controller for the intended flow rate or velocity.

Variable

SIMOTION distinguishes between:

- System variables

  Variables that are predefined by the system

- Global user variables and local user variables

  Variables that are defined by the user

→ System variable, Global user variable, Local user variable

Variable declaration

You specify names, data types and initial values for variables in the variable declaration.

Vertical application

Vertical applications are programs (subroutines) of a machine that solve specific sub-tasks (e.g. winder, flying shears).

→ Library

Virtual axis

All axis types can also be set as virtual axes, i.e. they do not have a real drive, but are only used for calculation, e.g. as a master axis for several slave axes.

W

Workbench

The SIMOTION user interface is divided into three function areas:

- Project navigator
- Working area
- Detail view

Different tools can be opened simultaneously in the working area and detail view and can be moved to the foreground by selecting tabs. Different toolbars are displayed depending on the active tab.

→ Project navigator, Working area (workbench), Detail view

Working area (workbench)

The term "working area" refers to an area on the SIMOTION SCOUT workbench. All tools called from the SIMOTION workbench to configure, parameterize, program, commission, etc. a SIMOTION project are displayed in the working area. Active tools can be moved to the foreground by selecting tabs.

→ SIMOTION SCOUT, Workbench

X

XML

Extensible Markup Language

XML is a standard for creation of documents in the form of a tree structure that can be read by both machines and people. XML defines the rules for the structure of such documents. The details of the respective document must be specified for a specific application case
("XML application"). This especially affects the definition of the structure elements and their arrangement within the document tree.