SEW EURODRIVE

Manual

MOVIKIT®

MultiMotion Auxiliary Velocity, MultiMotion Auxiliary Positioning

Edition 03/2020

25967886/EN

Table of contents

25967886/EN – 03/2020

25967886/EN – 03/2020

1 General information

1.1 About this documentation

This documentation is an integral part of the product. The documentation is intended for all employees who perform work on the product.

Make sure this documentation is accessible and legible. Ensure that persons responsible for the systems and their operation as well as persons who work with the product independently have read through the documentation carefully and understood it. If you are unclear about any of the information in this documentation, or if you require further information, contact SEW-EURODRIVE.

1.2 Content of the documentation

The descriptions in this documentation apply to the software and firmware versions applicable at the time of publication. These descriptions might differ if you install later software or firmware versions. In this case, contact SEW-EURODRIVE.

1.3 Structure of the warning notes

1.3.1 Meaning of signal words

The following table shows the grading and meaning of the signal words for safety notes.

Signal word	Meaning	Consequences if disregarded
⚠ **DANGER**	Imminent hazard	Severe or fatal injuries
⚠ **WARNING**	Possible dangerous situation	Severe or fatal injuries
⚠ **CAUTION**	Possible dangerous situation	Minor injuries
NOTICE	Possible damage to property	Damage to the product or its environment
INFORMATION	Useful information or tip: Simplifies handling of the product.	

1.3.2 Structure of section-related safety notes

Section-related safety notes do not apply to a specific action but to several actions pertaining to one subject. The hazard symbols used either indicate a general hazard or a specific hazard.

This is the formal structure of a safety note for a specific section:

SIGNAL WORD

Type and source of hazard.

Possible consequence(s) if disregarded.

- Measure(s) to prevent the hazard.

25967886/EN – 03/2020

Meaning of the hazard symbols

The hazard symbols in the safety notes have the following meaning:

Hazard symbol	Meaning
	General hazard

1.3.3 Structure of embedded safety notes

Embedded safety notes are directly integrated into the instructions just before the description of the dangerous action.

This is the formal structure of an embedded safety note:

⚠ **SIGNAL WORD!** Type and source of hazard. Possible consequence(s) if disregarded. Measure(s) to prevent the hazard.

1.4 Decimal separator in numerical values

In this document, a period is used to indicate the decimal separator.

Example: 30.5 kg

1.5 Rights to claim under limited warranty

Read the information in this documentation. This is essential for fault-free operation and fulfillment of any rights to claim under limited warranty. Read the documentation before you start working with the product.

1.6 Product names and trademarks

The brands and product names in this documentation are trademarks or registered trademarks of their respective titleholders.

1.7 Copyright notice

25967886/EN – 03/2020

1.8 Other applicable documentation

Observe the corresponding documentation for all further components.

Always use the latest edition of the documentation and the software.

The SEW-EURODRIVE website (www.sew-eurodrive.com) provides a wide selection of documents for download in various languages. If required, you can also order printed and bound copies of the documentation from SEW-EURODRIVE.

1.9 Short designation

The following short designations are used in this documentation:

Type designation	Short designation
MOVIKIT® MultiMotion Auxiliary Velocity	Software module
MOVIKIT® MultiMotion Auxiliary Positioning	Software module

25967886/EN – 03/2020

2 Safety notes

2.1 Preliminary information

The following general safety notes serve the purpose of preventing injury to persons and damage to property. They primarily apply to the use of products described in this documentation. If you use additional components, also observe the relevant warning and safety notes.

2.2 Target group

Software specialist

Any work with the software may only be performed by a specialist with suitable training. A specialist in this context is someone who has the following qualifications:

- Appropriate training
- Knowledge of this documentation and other applicable documentation
- SEW-EURODRIVE recommends additional training for products that are operated using this software.

2.3 Network security and access protection

A bus system makes it possible to adapt electronic drive technology components to the particulars of the machinery within wide limits. There is a risk that a change of parameters that cannot be detected externally may result in unexpected but not uncontrolled system behavior and may have a negative impact on operational safety, system availability, or data security.

Ensure that unauthorized access is prevented, especially with respect to Ethernet-based networked systems and engineering interfaces.

Use IT-specific safety standards to increase access protection to the ports. For a port overview, refer to the respective technical data of the device in use.

2.4 Designated use

MOVIKIT® MultiMotion Auxiliary Velocity and MOVIKIT® MultiMotion Auxiliary Positioning are software modules for MOVI-C® CONTROLLER that provide the user with simple motion functions such as speed control and positioning via an interface in the IEC program. The software modules allow for controlling non-synchronized auxiliary axes, for example for driving conveyor belts or variable-speed gear units.

Use the device-independent MOVISUITE® engineering software to start up and configure the axes and to download the complete configuration to a MOVI-C® CONTROLLER.

25967886/EN – 03/2020

3 System description

3.1 Module description

MOVIKIT® MultiMotion Auxiliary Velocity allows for configuring speed and torque control for non-interpolating axes.

MOVIKIT® MultiMotion Auxiliary Positioning provides the range of functions of MOVIKIT® MultiMotion Auxiliary Velocity and also allows for configuring positioning.

The software modules provide users with an interface in the IEC program similar to the interface of MOVIKIT® MultiMotion. The software modules are configured in the MOVISUITE® engineering tool using graphical user interfaces and can be monitored and controlled using a monitoring tool.

3.1.1 Advantages

The software modules offer the following advantages:

- Startup and configuration using the user interfaces in MOVISUITE®
- Control and diagnostics using a monitor tool integrated in MOVISUITE®
- Integration into the application program through automatic code generation
- Simple control of the functions via global variable interfaces
- User interface in the IEC program based on the interface of MOVIKIT® MultiMotion
- Use of drive-based functions
- Use of lean, scalable EtherCAT® telegrams

3.1.2 Areas of application

The software modules are used for controlling simple non-synchronized motion sequences of auxiliary axes. Examples of simple speed control are conveyor belts or roller conveyors for simple positioning tasks, and drives for format adjustments.

25967886/EN – 03/2020

3.2 Functions

3.2.1 MOVIKIT® MultiMotion Auxiliary Velocity

Overview of functions:

- "Speed control" operating mode: Speed, acceleration and deceleration are specified.
- "Torque control" operating mode: Torque and maximum/minimum speed are specified.

3.2.2 MOVIKIT® MultiMotion Auxiliary Positioning

Overview of functions:

- "Speed control" operating mode: Speed, acceleration and deceleration are specified.
- "Torque control" operating mode: Torque and maximum/minimum speed are specified.
- "Reference travel" operating mode: Reference travel can be configured.
- "Position control" operating mode: Position, speed, acceleration, deceleration, and jerk time are specified.
- "Jog" operating mode: Input of speed, acceleration, deceleration, and jerk time

25967886/EN – 03/2020

4 Project planning information

4.1 Requirement

Correct project planning and proper installation of the devices are required for successful startup and operation.

For detailed project planning information, refer to the documentation of the respective devices.

4.2 Hardware

The following hardware is required:

- MOVI-C® CONTROLLER (all power classes)

4.3 Software

The following software is required:

- MOVISUITE® engineering software

  (includes MOVIRUN® flexible and the IEC Editor)

For more detailed information on the hardware requirements of the individual software components, see the documentation for the respective software.

4.4 Licensing

The following licenses are available and are required:

- MOVIRUN® flexible

  License for the software platform MOVIRUN® flexible that also includes the licenses for MOVIKIT® MultiMotion Auxiliary Velocity and MOVIKIT® MultiMotion Auxiliary Positioning.

The license/licenses are referred to as performance licenses. They only have to be purchased once per MOVI-C® CONTROLLER and can then be used for any number of axes.

For further information on licensing, refer to the document "MOVI-C® Software Components". You can download the document from the SEW-EURODRIVE website (www.sew-eurodrive.com).

25967886/EN – 03/2020

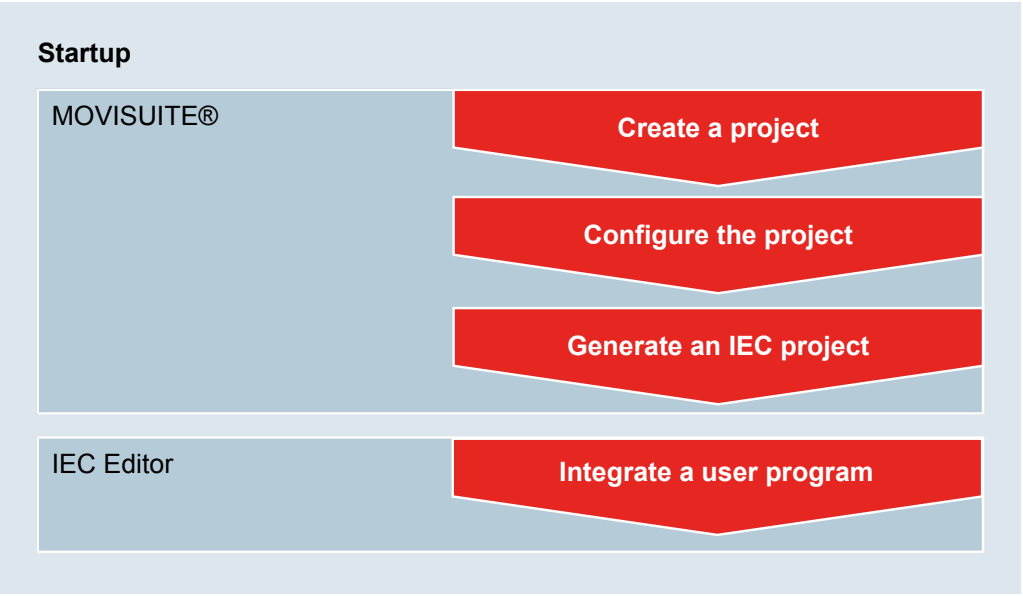
5 Startup

5.1 Requirements

- Check the installation of the inverters and, if installed, also check the encoder connection.
- Observe the installation notes in the documentation of the respective device and software components.
- The devices to be started up are displayed in MOVISUITE®.

5.2 Startup procedure

The schematic diagram below shows the startup procedure:



18014421798899467

The startup steps specific to these software modules are explained in detail in the following chapters of this manual. For startup, also observe the documentation of all the other components in use.

25967886/EN – 03/2020

5.3 Configuring a project

INFORMATION

For detailed information on how to operate the MOVISUITE® engineering software, refer to the corresponding documentation.

✓ A MOVISUITE® project has been created and is open.

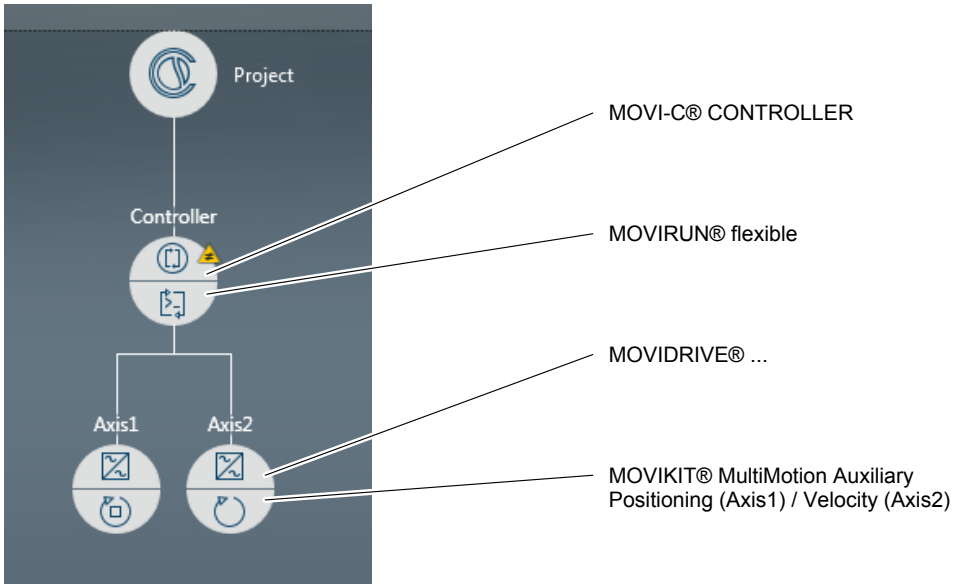
1. Add required device nodes, software nodes (MOVI-C® SoftwareNode) and software modules to the project.

   ⇨ See "Example project".

2. Configure the added devices or software modules. If available, observe the specific notes in the following chapters that apply to MOVIKIT® MultiMotion Auxiliary Velocity, MultiMotion Auxiliary Positioning. For detailed information on the configuration of devices or other software modules, refer to the respective documentation.

5.3.1 Example project

The following figure shows an example project:



31222727435

5.3.2 Adding MOVIKIT® MultiMotion Auxiliary Velocity, MultiMotion Auxiliary Positioning

INFORMATION

For detailed information on how to operate the MOVISUITE® engineering software, refer to the corresponding documentation.

✓ A MOVISUITE® project has been created and is open.

1. Click on the empty software module section of the required node.
   ⇨ The catalog section opens and displays the available software modules.
2. In the catalog section, click on MOVIKIT® MultiMotion Auxiliary Velocity, MultiMotion Auxiliary Positioning.
   ⇨ A context menu opens.
3. Select the version from the respective drop-down list in the context menu and confirm your selection with [Apply].

⇨ MOVIKIT® MultiMotion Auxiliary Velocity, MultiMotion Auxiliary Positioning is assigned to the node, the configuration is created, and the basic settings are performed.

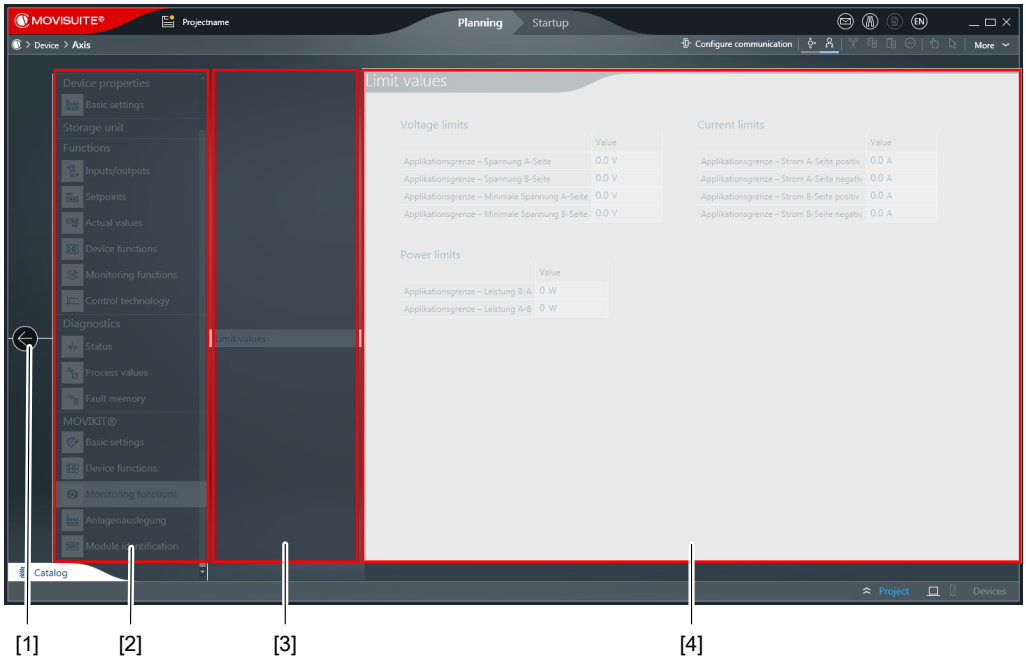
25967886/EN – 03/2020

5.3.3 Configuring MOVIKIT® MultiMotion Auxiliary Velocity, MultiMotion Auxiliary Positioning

INFORMATION

For detailed information on how to operate the MOVISUITE® engineering software, refer to the corresponding documentation.

1. In MOVISUITE®, click on MOVIKIT® MultiMotion Auxiliary Velocity, MultiMotion Auxiliary Positioning.
   ⇨ The configuration menus of the software module are displayed. The configuration menus are explained in the following subchapters.



9007228165413771

[1] Button to return to the project overview
[2] Main menu of the software module configuration (MOVIKIT® section)
[3] Submenus of the configuration
[4] Setting fields of the respective submenu

2. Configure the software module using the respective setting fields.
3. Click button [1] after having completed the configuration.
   ⇨ The project overview is displayed.

INFORMATION

For the changes made to the configuration to take effect, you have to update the configuration data. To do so, click [Update configuration data] in the respective notification at the node or in the context menu of the MOVI-C® CONTROLLER. The MOVI-C® CONTROLLER is stopped and restarted for updating the configuration data.

25967886/EN – 03/2020

Monitoring functions

Reference signals

Parameter group	Description
Comparison of setpoint/actual speed values	
Half window width	Speed setpoint window
	Index: 8324.3
	IEC name: -
Delay time	Delay time for the comparison of setpoint and actual value. The condition for the signal must be fulfilled for at least this time period so that the signal is issued.
	Index: 8324.4
	IEC name: -
Polarity	Determines when the signal is set. • Signal if actual value = setpoint • Signal if actual value <> setpoint
	Index: 8324.5
	IEC name: -
In position	
Window width	The "In position" signal is set when the difference between the actual position and setpoint position is smaller than half this value.
	Index: 8331.1
	IEC name: -
Hysteresis	Target position hysteresis. When the position window is left, the "In position" signal is maintained until this value is exceeded.
	Index: 8331.2
	IEC name: -
Actual target position in user units	Actual target position in user units
	Index: 8331.3
	IEC name: Parameter.InPosSignal.ActualTargetPosition
Setpoint/actual torque comparison	
Half window width	Torque setpoint hysteresis of FCB 07 in drive train 1
	Index: 8339.3
	IEC name: Parameter.TorqueSetpointActValComp.HalfWindowWidth1

25967886/EN – 03/2020

Parameter group	Description
Delay time	Delay time for comparison of setpoint and actual value in drive train 1. To have the signal issued, the condition for the signal must be fulfilled for at least this time period.
	Index: 8339.4
	IEC name: Parameter.TorqueSetpointActVal-Comp.DelayTime1
Polarity	Specifies when the signal is issued in drive train 1: • Signal if actual value = setpoint • Signal if actual value <> setpoint
	Index: 8339.5
	IEC name: Parameter.TorqueSetpointActVal-Comp.Polarity1

Software limit switches

Parameter name	Value
Software limit switches	
Monitoring negative SW limit switch	• On Activate monitoring for negative software limit switch(es) • **Off** Deactivate monitoring for negative software limit switch(es)
	Index: 8572.3
	IEC name: LimitSwitchEvaluation.SoftwareLimit-Switch.In.xActivateMonitoringNegative
SW limit switch negative	Position of the negative software limit switch (in user units)
	Index: 8572.4
	IEC name: LimitSwitchEvaluation.SoftwareLimit-Switch.In.lrLimitNegative
Monitoring positive SW limit switch	• On Activate monitoring for positive software limit switch(es) • **Off** Deactivate monitoring for positive software limit switch(es)
	Index: 8572.5
	IEC name: LimitSwitchEvaluation.SoftwareLimit-Switch.In.xActivateMonitoringPositive

25967886/EN – 03/2020

Parameter name	Value
SW limit switch positive	Position of the positive software limit switch (in user units)
	Index: 8572.6
	IEC name: LimitSwitchEvaluation.SoftwareLimit-Switch.In.lrLimitPositive

Limit values

Parameter name	Value
Application limits	
Positive speed	Limits the maximum positive speed permitted for moving the system. (in user units)
	Index: 8357.10
	IEC name: ConfigHandling._stAxisConfig.lrAppLim-itVelocityPositive
Negative speed	Limits the maximum negative speed permitted for moving the system. (in user units)
	Index: 8357.11
	IEC name: ConfigHandling._stAxisConfig.lrAppLim-itVelocityNegative
Acceleration	Limits the maximum acceleration permitted for accelerating the system. (in user units)
	Index: 8357.12
	IEC name: ConfigHandling._stAxisConfig.lrAppLimit-Acceleration
Deceleration	Limits the maximum deceleration permitted for braking the system. (in user units)
	Index: 8357.13
	IEC name: ConfigHandling._stAxisConfig.lrAppLimit-Deceleration
Jerk time	Limits the jerk time in [ms] The jerk time takes effect in torque control (FCB 07), speed control (FCB 05), and positioning control (FCB 09), as well as in manual mode. The positioning process extends to twice the set jerk time.
	Index: 8357.14
	IEC name: ConfigHandling._stAxisConfig.lrAppLim-itJerkTime

25967886/EN – 03/2020

Parameter name	Value
Torque	Limits the maximum torque that may be applied to the system in [Nm]
	Index: 8357.15
	IEC name: -
Limits	
Emergency stop deceleration	Deceleration for the ramp that is activated during an emergency stop. An emergency stop can be programmed as a response to a fault (in user units).
	Index: 8357.20
	IEC name: ConfigHandling._stAxisConfig.lrRapidStopDeceleration
Cycle limit	
Modulo minimum	Lower modulo limits for handling process data. This limit is required for handling process data with a limited range of values (in user units).
	Index: 8357.30
	IEC name: ConfigHandling._stAxisConfig.lrModuloMin
Modulo maximum	Upper modulo limits for handling process data. This limit is required for handling process data with a limited range of values (in user units).
	Index: 8357.31
	IEC name: ConfigHandling._stAxisConfig.lrModuloMax
Lag error	
Lag error window	Lag error from which drive train 1 signals a fault (in user units).
	Index: 8510.4
	IEC name: -
Limit values from startup	
Maximum speed at motor shaft	Maximum permitted speed at the motor shaft calculated from motor and gear unit data during startup in [min^{-1}].
	Index: 8360.9
	IEC name: -
Maximum torque at motor shaft	Maximum permitted torque at the motor shaft calculated from motor and gear unit data during startup in [Nm].
	Index: 8360.11
	IEC name: -

25967886/EN – 03/2020

Drive functions

Scaling

Parameter name	Value
Encoder	
Actual position source	Encoder that acts as a source for generating the actual position.
	Index: 8565.3
	IEC name: -
Inverter scaling	
Display of the inverter scaling	
Index: 8554.1-4 (position), 8557.1-4 (speed), 8560.1-4 (acceleration)	
IEC name: -	

FCB 07 Torque control

Parameter name	Value
Setpoint/actual torque comparison	
Half window width	Torque setpoint hysteresis of FCB 07 in drive train 1
	Index: 8339.3
Delay time	Delay time for comparison of setpoint and actual value in drive train 1. To have the signal issued, the condition for the signal must be fulfilled for at least this time period.
	Index: 8339.4
Polarity	Specifies when the signal is issued in drive train 1: • Signal if actual value = setpoint • Signal if actual value <> setpoint
	Index: 8339.5

25967886/EN – 03/2020

FCB 09 Position control

Parameter name	Value
Lag error	
Lag error window	Specifies from which lag error the drive signals a fault (drive train 1). The "Lag error window" parameter takes effect for FCB 09 and FCB 26.
	Index: 8509.4
Response to positioning lag error	Specifies how the device responds to a lag error (lag error window exceeded, Index 8509.4). The "Response to positioning lag error" parameter takes effect for FCB 09, FCB 10, and FCB 26.
	Index: 8622.3

FCB 12 reference travel

Parameter name	Value
FCB 12 Reference travel	
Type	• Deactivated • Zero pulse – negative direction • Reference cam – negative end • Reference cam – positive end • Positive limit switch • Negative limit switch • Reference cam flush – positive limit switch • Reference cam flush – negative limit switch • Referencing without reference travel • Positive fixed stop • Negative fixed stop
	Index: 8552.1
Reference to zero pulse	Activates or deactivates referencing to zero pulse
	Index: 8552.2
Reference offset	Deviation of the cam from the machine zero
	Index: 8552.5
Search speed	Search speed for reference travel
	Index: 8552.8
Retraction speed	Retraction speed for reference travel
	Index: 8552.9
Acceleration	Acceleration of reference travel
	Index: 8552.11

25967886/EN – 03/2020

Parameter name	Value
Deceleration	Deceleration of reference travel
	Index: 8552.12
Advanced settings	
Go to home position	Activates or deactivates homing
	Index: 8552.3
Home position	Home position that is approached automatically after reference travel is complete.
	Index: 8552.7
Homing speed	Speed for approaching the home position after referencing.
	Index: 8552.10
Jerk time	Homing jerk time
	Index: 8552.13
Speed changeover before fixed stop	For setting whether search speed changes over to retraction speed.
	Index: 8552.4
Dwell time at fixed stop	Dwell time at fixed stop
	Index: 8552.15
Torque limit fixed stop	Limits the torque when referencing to the fixed stop.
	Index: 8552.14

25967886/EN – 03/2020

Advanced settings

Parameter settings

Parameter name	Value
Delivery state	
"Initialize settings" button	Initialize the software module and the process data interface between controller and inverter. **Information:** If you perform an initialization, add a software module, or update the version, the inputs of the inverter are set to "No function". Settings, such as HW limit switches or reference cams, will be lost. For information on how to update the version of a software module, refer to the installation instructions "MOVISUITE® standard – Installation and Project Adjustment".

Process data profile

Parameter name	Value
Select process data profile	
Process data profile	Setting regarding how much and which data is exchanged between inverter and MOVI-C® CONTROLLER.
	Index: 50000.10
	IEC name: -
Apply process data settings	
“Apply process data settings” button	Configure the process data interface according to the selected process data profile.

Module identification

Parameter group	Description
Module identification	Includes name and version for identifying the software module.

25967886/EN – 03/2020

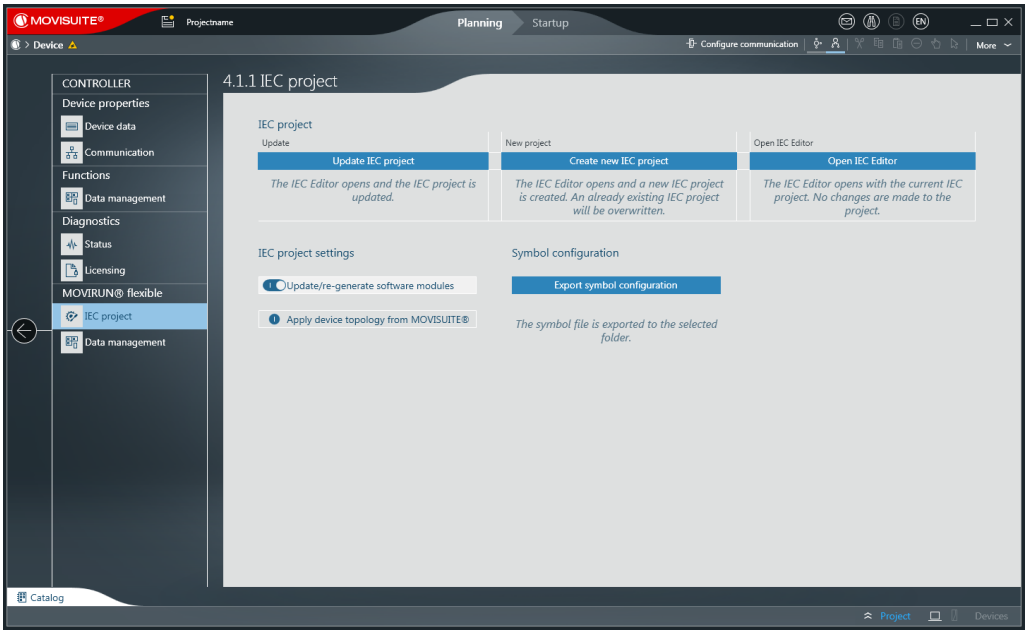
5.4 Generating an IEC project

Carry out the following steps to create an IEC project using automatic code generation and based on the configuration settings in MOVISUITE®.

✓ Configuration of the MOVISUITE® project has been completed.

1. In the function view of MOVISUITE®, click the software module section of the MOVI-C® CONTROLLER.

   ⇨ The "IEC project" menu opens.



27021618448637067

INFORMATION

If you have carried out the configuration in MOVISUITE® using the "Startup" mode and the message "Device cannot be reached" appears, proceed as follows:

- If the MOVI-C® CONTROLLER is not available via the network, switch over to "Planning" mode.
- If the MOVI-C® CONTROLLER is available via the network, carry out a network scan and connect the MOVI-C® CONTROLLER in the network view with the MOVI-C® CONTROLLER in the function view.

2. Click [Create new IEC project].

   ⇨ The IEC Editor opens and a new IEC project is created.

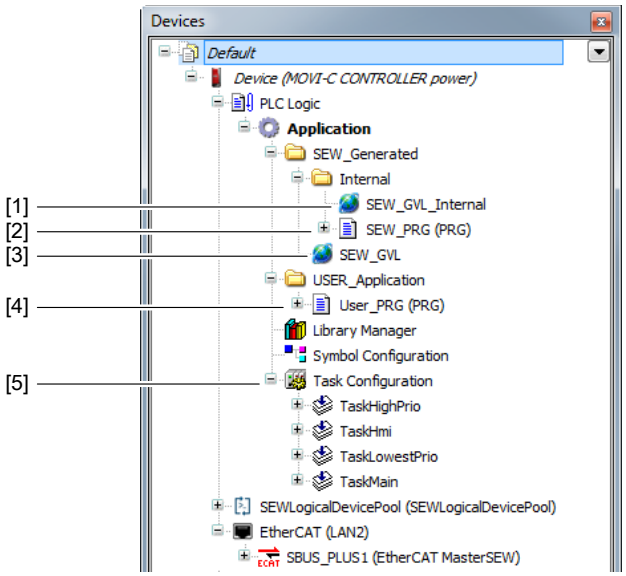
INFORMATION

If changes are made to the project structure, to inverter data sets, or to a software module configuration after the IEC project is generated for the first time, a notification symbol is displayed on the MOVI-C® CONTROLLER node. Click on the message icon for more information about the change, and to update the IEC project.

25967886/EN – 03/2020

5.4.1 IEC project structure

The IEC project has the following basic structure:

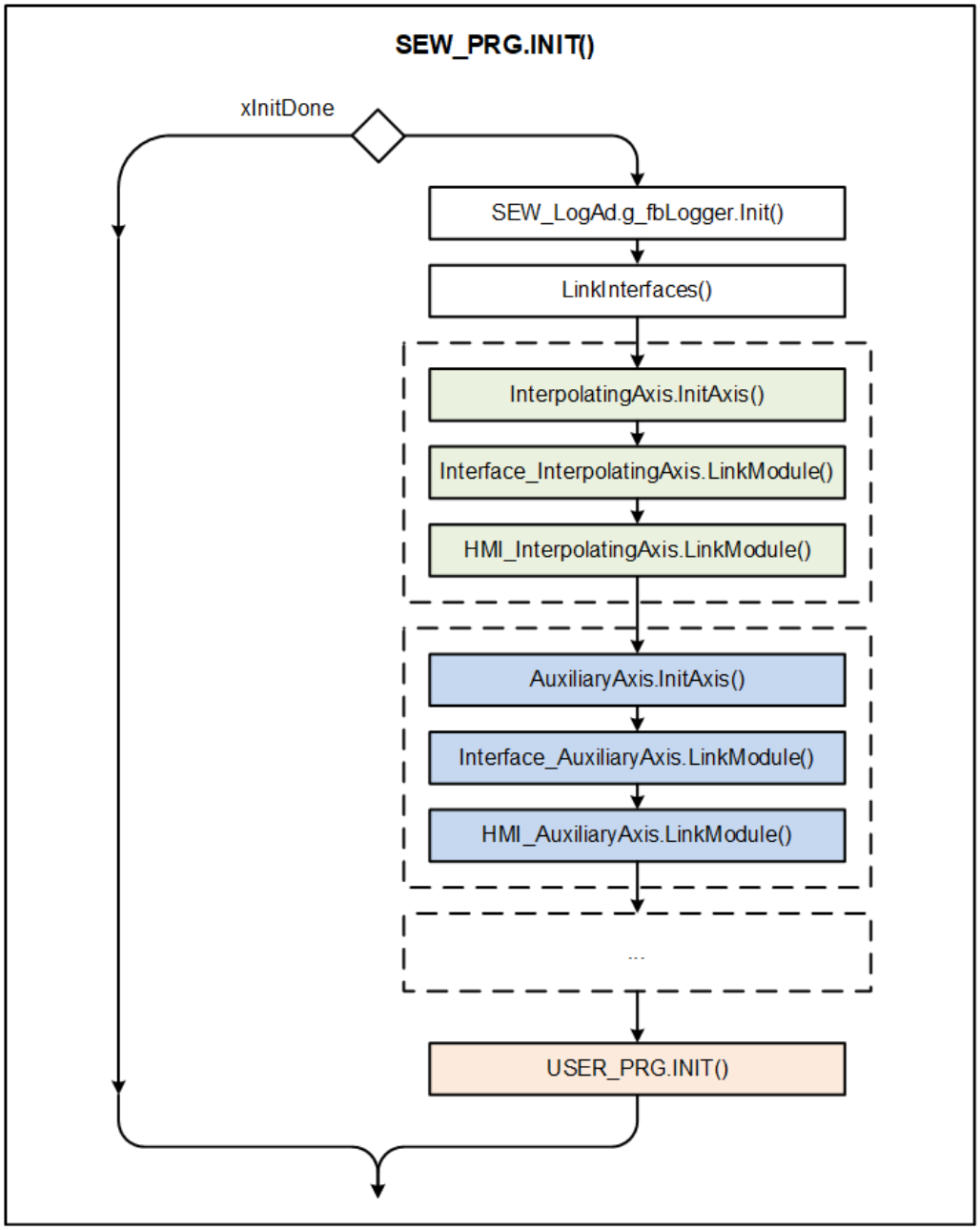


18014423003085323

No.	Name	Description
[1]	SEW_GVL_Internal	The SEW_GVL_Internal global list of variables contains the instances that correspond to the software module used. These variables may not be written to from the user program. In addition, the structure contains an instance as a communication buffer for controlling or monitoring the software module by means of a monitor.
[2]	SEW_PRG	Program that contains all the important instance calls. Automatic code generation recreates this program in accordance with the configuration made in MOVISUITE® each time the IEC project is created, thereby overwriting the previous version. Therefore, you should not make any changes to this program.
[3]	SEW_GVL	The SEW_GVL global list of variables is the interface for accessing the software module features.
[4]	User_PRG	The user program is created once, initially, by automatic code generation. Since the program is not overwritten with each subsequent creation, this is the appropriate place for integrating user programs. The program is divided into five actions. These actions differ in the time at which they are called during the program sequence.
[5]	Task configuration	The list of tasks created in the project. Automatic code generation initially adds tasks that differ in how they are prioritized. The user can add additional programs to existing tasks or create new tasks. It is the responsibility of the user to design the capacity utilization of the tasks to enable the tasks to be processed within the required cycle time. Moving beyond the cyclical tasks, in particular, prevents setpoints for the interpolating axes being generated in time, which means that these axes cannot be operated properly.

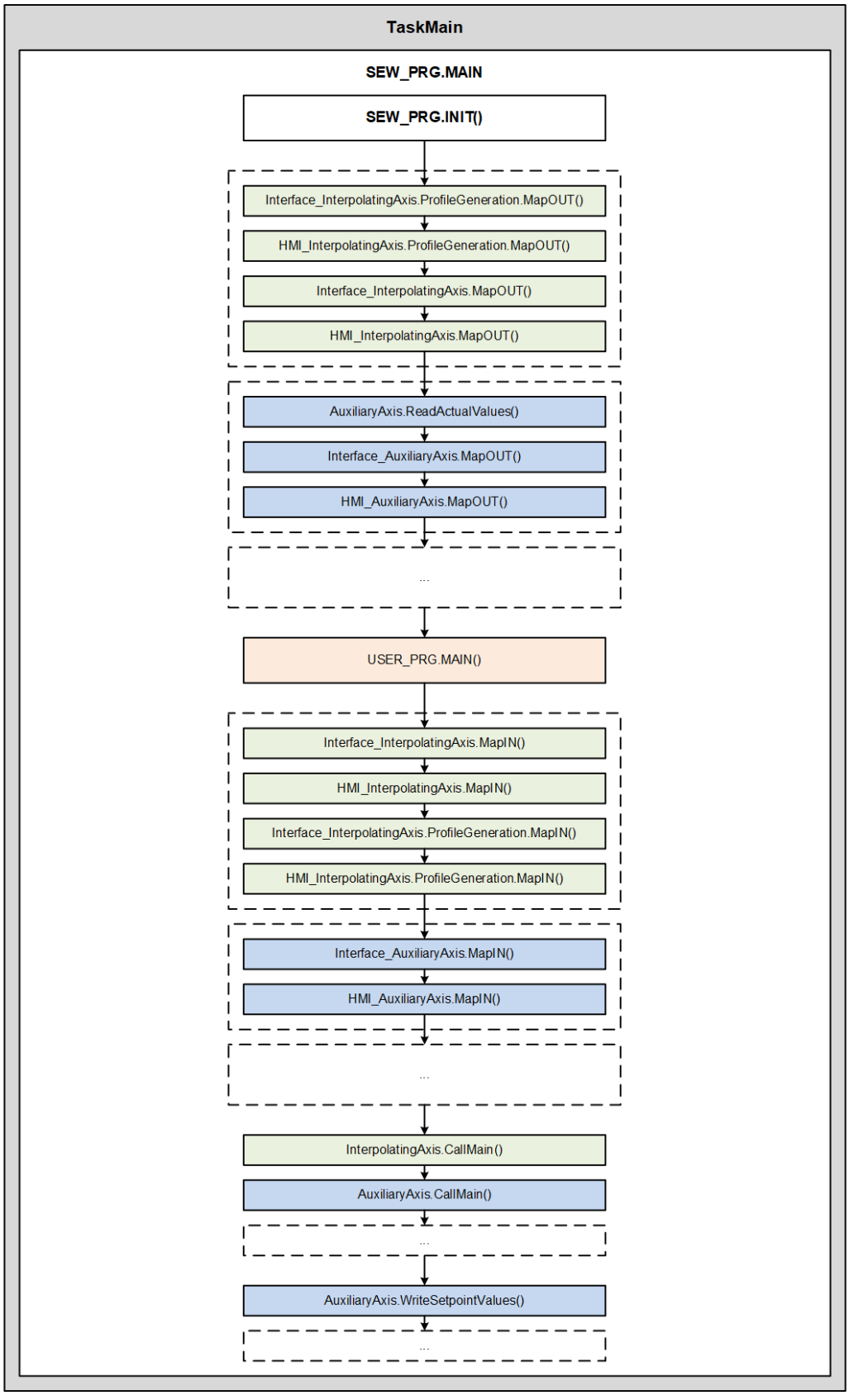
5.5 Integrating a user program

There are various ways to integrate a user program into the IEC project created. The figure below shows the basic structure of the software. In addition, you can see where the actions and methods are processed during program execution. The actions or methods highlighted in red are available for executing the user program. Program calls of an auxiliary axis are indicated in blue. Program calls of an interpolating axis are indicated in green.

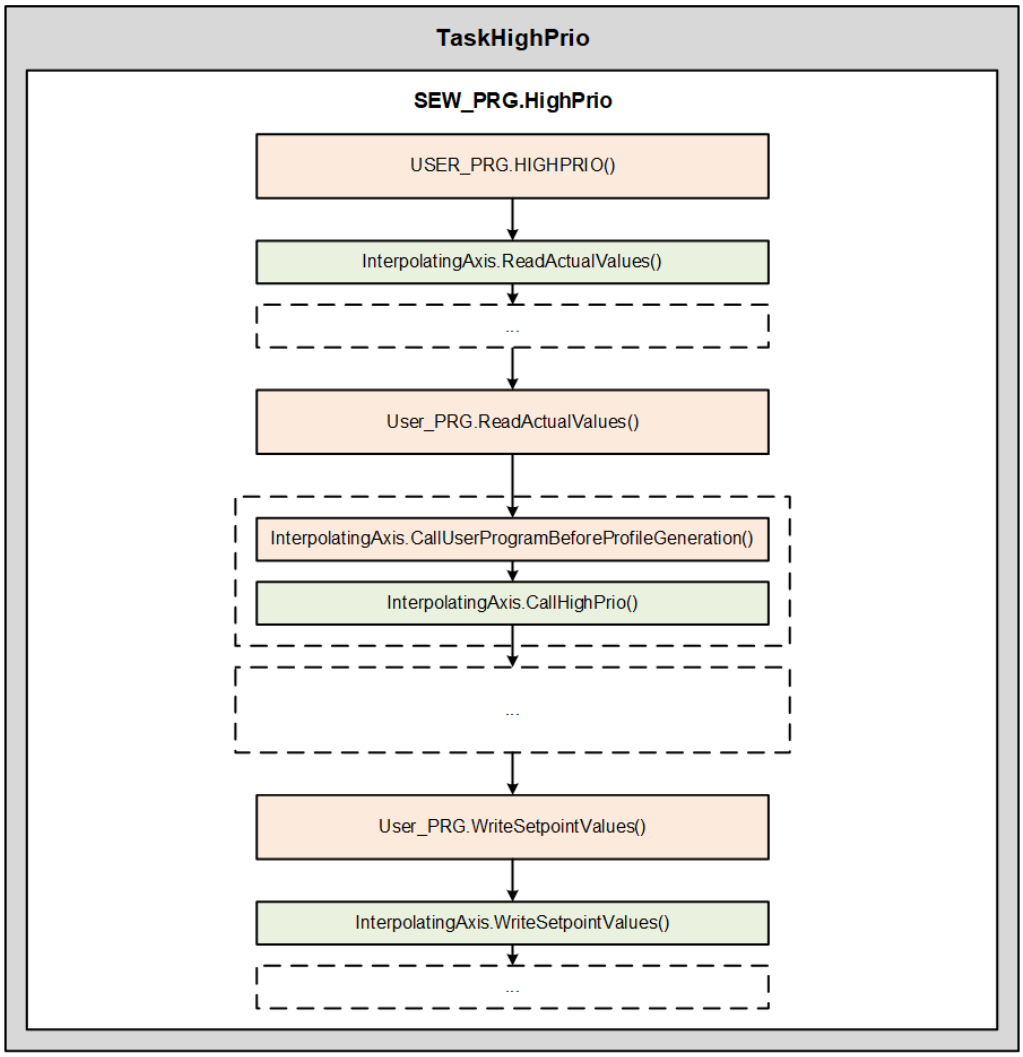


25291027851

25967886/EN – 03/2020



25291030411



25291032971

5.5.1 Tasks of the mapping functions

Code generation places the mapping to the interfaces in the free-wheeling task because in most cases, the user program is also processed within the free-wheeling task (e.g. in the *User_PRG.MAIN* action). Keep in mind that for cyclical tasks (e.g. in profile generation) all information written from the free-wheeling task to the interfaces (e.g. *Interface_AuxiliaryAxis*) is processed asynchronously and, as a result, in a potentially inconsistent manner.

In certain cases, however, it may be important to control the system synchronously to the cyclical task from the user program (e.g. during an on-the-fly changeover). The corresponding control actions of the user program must then be processed in the cyclical task (e.g. in the *User_PRG.HIGHPRIO* action). Furthermore, the mapping functions of the status signals or control signals processed within the cyclical task must also be called from within the cyclical task, i.e. the corresponding calls must be shifted from the *SEW_PRG.MAIN* action to the *SEW_PRG.HIGHPRIO* action.

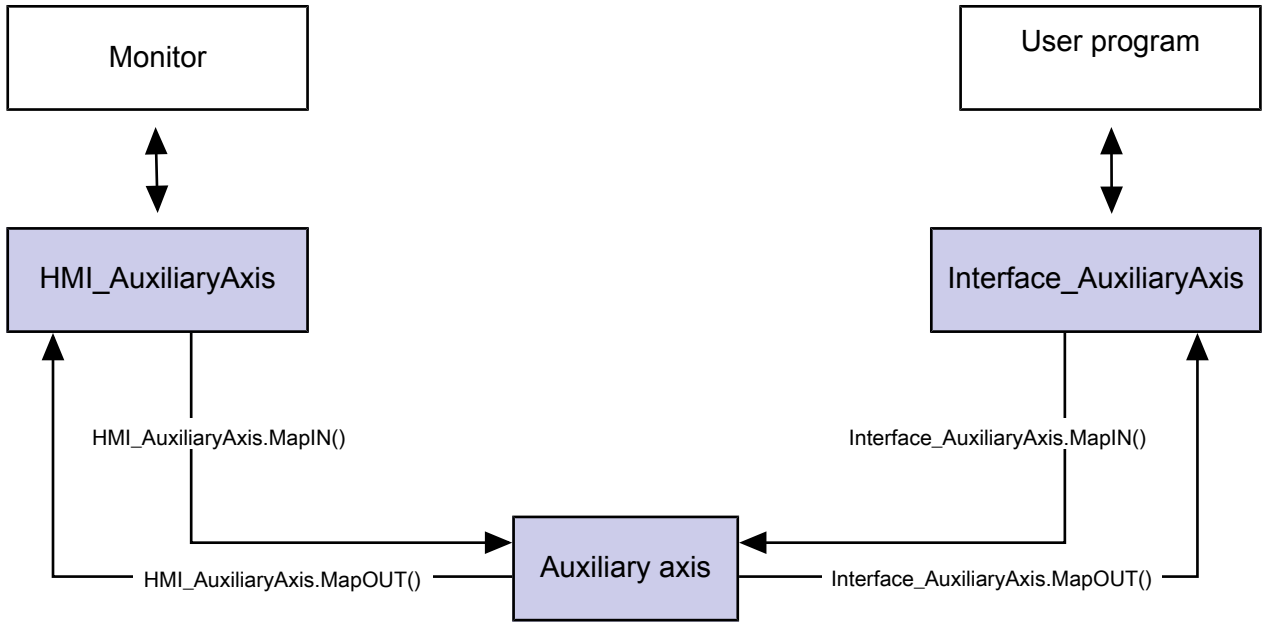
INFORMATION

It is the responsibility of the user to ensure consistency by taking the appropriate measures if necessary.

25967886/EN – 03/2020

AuxiliaryAxes mapping functions

The diagram below presents an overview of the mapping functions and their tasks. If necessary, you can use these tasks to decide which calls are to be modified.



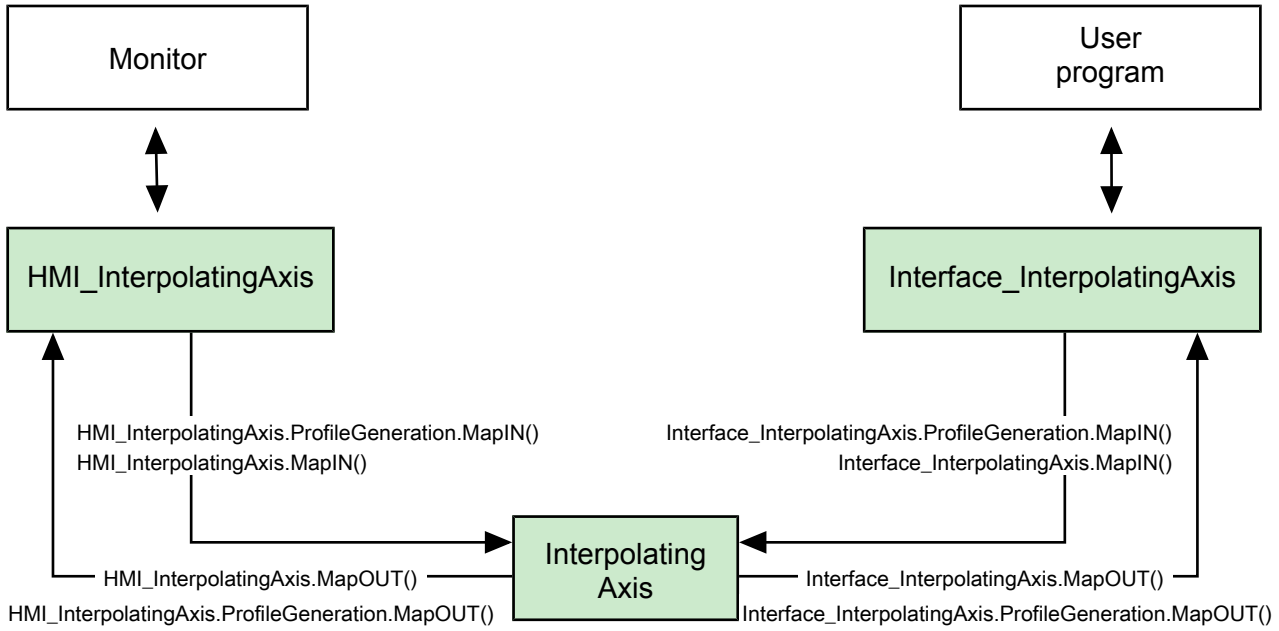
25291114891

- *Interface_AuxiliaryAxis.MapOUT()* copies the status information from *AuxiliaryAxis* to *Interface_AuxiliaryAxis*.
- *Interface_AuxiliaryAxis.MapIN()* copies the status information from *Interface_AuxiliaryAxis* to *AuxiliaryAxis*.
- *HMI_AuxiliaryAxis.MapOUT()* copies the status information from *AuxiliaryAxis* to *HMI_Auxiliary*.
- *HMI_AuxiliaryAxis.MapIN()* copies the control information from *HMI_AuxiliaryAxis* to *AuxiliaryAxis*.

25967886/EN – 03/2020

InterpolatingAxes mapping functions

The diagram below presents an overview of the mapping functions and their tasks. If necessary, you can use these tasks to decide which calls are to be modified.



25291130123

- *Interface_InterpolatingAxis.MapOUT()* copies the status information from *InterpolatingAxis* to *Interface_InterpolatingAxis*.
- *Interface_InterpolatingAxis.ProfileGeneration.MapOUT()* copies the status information of profile generation from *InterpolatingAxis* to *Interface_InterpolatingAxis*.
- *Interface_InterpolatingAxis.MapIN()* copies the control information from *Interface_InterpolatingAxis* to *InterpolatingAxis*.
- *Interface_InterpolatingAxis.ProfileGeneration.MapIN()* copies the control information of profile generation from *Interface_InterpolatingAxis* to *InterpolatingAxis*.
- *HMI_InterpolatingAxis.MapOUT()* copies the status information from *InterpolatingAxis* to *HMI_InterpolatingAxis*.
- *HMI_InterpolatingAxis.ProfileGeneration.MapOUT()* copies the status information of profile generation from *InterpolatingAxis* to *HMI_InterpolatingAxis*.
- *HMI_InterpolatingAxis.MapIN()* copies the status information from *HMI_InterpolatingAxis* to *InterpolatingAxis*.
- *HMI_InterpolatingAxis.ProfileGeneration.MapIN()* copies the control information of profile generation from *HMI_InterpolatingAxis* to *InterpolatingAxis*.

25967886/EN – 03/2020

6 IEC programming

6.1 Opening the IEC project

- If an IEC project has already been generated, select the entry [IEC Editor] under "Tools" from the context menu of the MOVI-C® CONTROLLER in MOVISUITE®.
- If no IEC project has been generated, follow the steps described in the "Generating an IEC project" (→ 🗎 24) chapter.

6.2 User interface

The user interface is implemented in the IEC program by an instance in the global variable list *SEW_GVL*.

6.2.1 MOVIKIT® MultiMotion Auxiliary Velocity

The following figure shows the interface in the IEC Editor:

Interface_Axis1	SEW_MK_MultiMotionAux.MultiMotionVelocityAxis_UI	
xError	BOOL	FALSE
xWarning	BOOL	FALSE
udiMessageID	UDINT	16#00000000
sAdditionalText	STRING(Constants.gc_udiLengthAdditionalText)	''
xReset	BOOL	FALSE
xGetAccessControl	BOOL	TRUE
xControlActive	BOOL	FALSE
xInitDone	BOOL	FALSE
Basic	SEW_IAx.ST_Basic	
Inverter	SEW_IAx.ST_Inverter	
EnergySaving	SEW_IAx.ST_EnergySaving2	
Brake	SEW_IAx.ST_Brake	
SoftwareLimitSwitch	SEW_SWLS.SoftwareLimitSwitch_UI	
PosHoldCtrl	SEW_IAx.ST_PosHoldCtrl	
RotorPosIdentification	SEW_IAx.ST_RotorPosIdentification	
Velocity	SEW_UIDM.ModeVelocity_UI	
Torque	SEW_UIDM.ModeTorque_UI	

25967886/EN – 03/2020

6.2.2 MOVIKIT® MultiMotion Auxiliary Positioning

The following figure shows the interface in the IEC Editor:

Interface_Axis2	SEW_MK_MultiMotionAux.MultiMotionPositioningAxis_UI	
xError	BOOL	FALSE
xWarning	BOOL	FALSE
udiMessageID	UDINT	16#00000000
sAdditionalText	STRING(Constants.gc_udiLengthAdditionalText)	''
xReset	BOOL	FALSE
xGetAccessControl	BOOL	TRUE
xControlActive	BOOL	FALSE
xInitDone	BOOL	FALSE
Basic	SEW_IAx.ST_Basic	
Inverter	SEW_IAx.ST_Inverter	
EnergySaving	SEW_IAx.ST_EnergySaving2	
Brake	SEW_IAx.ST_Brake	
SoftwareLimitSwitch	SEW_SWLS.SoftwareLimitSwitch_UI	
PosHoldCtrl	SEW_IAx.ST_PosHoldCtrl	
RotorPosIdentification	SEW_IAx.ST_RotorPosIdentification	
Velocity	SEW_UIDM.ModeVelocity_UI	
Torque	SEW_UIDM.ModeTorque_UI	
Homing	SEW_UIDM.ModeHoming_UI	
Positioning	SEW_UIDM.ModePositioning_UI	
Jog	SEW_UIDM.ModeJog_UI	

25967886/EN – 03/2020

6.3 Basic functions

6.3.1 Diagnostics

Variables for reporting and writing errors and warnings.

Variable name	Description
xError	Data type – BOOL
	• TRUE – Error present • FALSE – No error present
xWarning	Data type – BOOL
	• TRUE – Warning present • FALSE – No warning present
udiMessageID	Data type: UDINT
	Message ID number
sAdditionalText	Data type: STRING
	Additional message text
xReset	Data type – BOOL
	• TRUE – Reset messages • FALSE – Do not reset messages

6.3.2 Access management

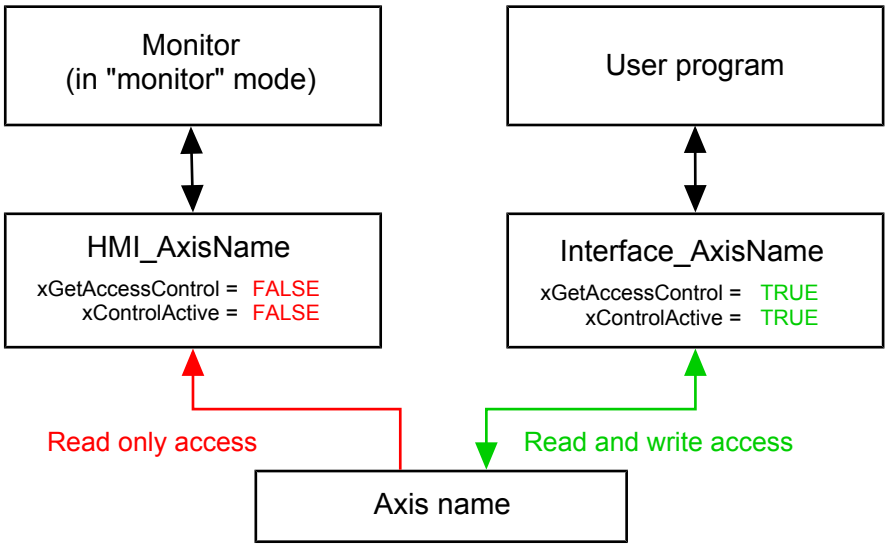
The access management system controls access to the user interface and ensures that no collisions occur during concurrent access to the axis functions.

Variable name	Description
xGetAccessControl	Data type – BOOL
	• TRUE – Request access • FALSE – Return access
xControlActive	Data type – BOOL
	• TRUE – Access granted • FALSE – Access denied

25967886/EN – 03/2020

User program access

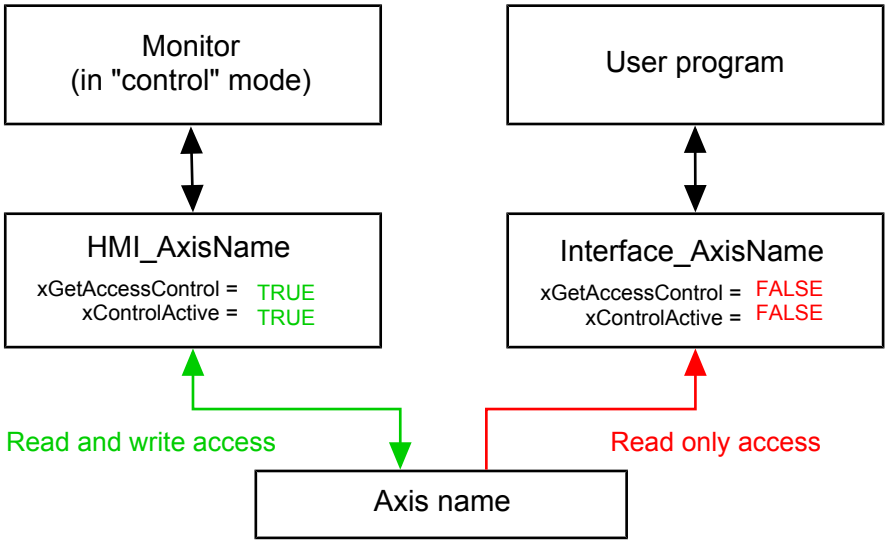
An instance requests access by setting *xGetAccessControl* to "TRUE." If *xControlActive* returns a "TRUE" value, access has been granted and is now permitted.



18014422491696011

Monitor access

If the axis function is accessed at the same time as the user program using a monitor in control mode, the *Interface_AxisName* user interface loses its access permission and *xControlActive* reports back "FALSE". Instead, access is realized using the *HMI_AxisName* interface to the monitor. In this case, *xControlActive* reports back "TRUE".



18014422491831691

25967886/EN – 03/2020

6.3.3 Basic axis functions (Basic)

The *Basic* structure contains the basic functions of the axis. This part of the user interface is identical for all axis types.

The following control and status variables are available:

Interface in the IEC Editor

9007219940779147

IN

Variable name	Description
xEnable_EmergencyStop	Data type: BOOL
	• TRUE – The software module does not execute an emergency stop using the set emergency stop ramp. (Requirement for enabling the axis) • FALSE – The software module executes an emergency stop using the set emergency stop ramp.
xEnable_ApplicationStop	Data type: BOOL
	• TRUE – The software module does not execute an application stop using the deceleration set to below the application limits. (Requirement for enabling the axis) • FALSE – The software module executes an application stop using the deceleration set under the application limits.

Comment:

To enable the axis, set *xEnable_EmergencyStop* and *xEnable_ApplicationStop* to "TRUE".

OUT

The dynamic parameters are scaled in user units.

Variable name	Description
lrActualPosition	Data type: LREAL – Floating-point number
	Actual position
lrActualVelocity	Data type: LREAL – Floating-point number
	Actual speed
xStandstill	Data type: BOOL
	• TRUE – Axis is at a standstill • FALSE – Axis is not at a standstill

25967886/EN – 03/2020

6.3.4 Inverter functions (Inverter)

The *Inverter* structure contains control and status variables for the inverter. This part of the user interface represents a hardware device and, as a result, only exists with real axes.

The following control and status variables are available:

Interface in the IEC Editor

Inverter	SEW_IAx.ST_Inverter2	
In	ST_Inverter_In	
xInhibit	BOOL	FALSE
wDigitalOutputs	WORD	0
xSimulation	BOOL	FALSE
lrTorqueLimit	LREAL	32.767
Out	ST_Inverter_Out2	
xConnected	BOOL	TRUE
xPowered	BOOL	FALSE
xReady	BOOL	TRUE
xReferenced	BOOL	FALSE
xSetpointActive	BOOL	FALSE
xSafeStop	BOOL	FALSE
xPositionValid	BOOL	FALSE
wDigitalInputs	WORD	0
lrActualTorque	LREAL	0
eActualInverterMode	E_INVERTERMODE	EmergencyStop
usiErrorID	USINT	0
usiErrorSubID	USINT	0
xSimulation	BOOL	FALSE

18014419195524875

IN

Variable name	Description
xInhibit	Data type: BOOL
	• TRUE – Inhibit the output stage of the device. When this signal is activated, the brake might engage; otherwise, the axis coasts to a halt. • FALSE – Enable the output stage of the device.
wDigitalOutputs	Data type: WORD
	Control of the digital outputs of the device **Information:** Depending on the software module used, digital outputs are already reserved and cannot be controlled here.
xSimulation (function not yet available)	Data type – BOOL
	• TRUE – Simulate the frequency inverter of the software module (e.g. when testing without hardware). • FALSE – Do not simulate the frequency inverter.
lrTorqueLimit	Data type: LREAL – Floating-point number
	Torque limit (1.0 = 100% M_N) **Information:** If a jerk time is to be processed, a process data word must be connected accordingly. See "Processing jerk time" (→ 🗎 72).

25967886/EN – 03/2020

OUT

Variable name	Description
xConnected	Data type – BOOL
	• TRUE – Communication connection with controller • FALSE – No communication connection with controller
xPowered	Data type – BOOL
	• TRUE – Output stages enabled (provide output voltage) • FALSE – Output stages not enabled
xReady	Data type – BOOL
	• TRUE – Ready for control by the controller • FALSE – Not ready for control by the controller
xReferenced	Data type: BOOL
	• TRUE – Referenced • FALSE – Not referenced
xSetpointActive	Data type – BOOL
	• TRUE – Setpoints are processed • FALSE – Setpoints are not processed
xSafeStop	Data type – BOOL
	• TRUE – Axis is at standstill (STO active) • FALSE – Axis is not at a standstill (STO is not active)
xPositionValid	Data type: BOOL
	• TRUE – Encoder position is valid • FALSE – Encoder fault is present (e.g. due to a bird strike)
wDigitalInputs	Data type: WORD
	State of the digital inputs **Information:** Depending on the software module used, digital inputs are already reserved and cannot be controlled here.
lrActualTorque	Data type: LREAL – Floating-point number
	Actual torque (1.0 = 100% M_N)

25967886/EN – 03/2020

Variable name	Description
eActualInverterMode	Data type: E_INVERTERMODE
	Operating mode of the inverter (FCB of the inverter): • Unknown • Default • OutputDisabled (FCB 01) • ManualMode (FCB 04) • Stop (FCB 02) • Homing (FCB 12) • JogMode (FCB 20) • BrakeTest (FCB 21) • Positioning (FCB 09) • PositioningInterpolated (FCB10) • Velocity (FCB 05) • VelocityInterpolated (FCB 06) • Torque (FCB 07) • TorqueInterpolated (FCB 08) • MotorParamMeasurement (FCB 25) • PosHoldCtrl (FCB 19) • RotorPosIdentification (FCB 18) • ApplicationStop (FCB 13) • EmergencyStop (FCB 14) • UserStop (FCB 26) *Library: SEW DeviceHandler Interfaces*
usiErrorID	Data type: USINT
	Error ID
usiErrorSubID	Data type: USINT
	Suberror ID
xSimulation	Data type – BOOL
	• TRUE – Simulation of the axis activated • FALSE – Simulation of the axis not activated

25967886/EN – 03/2020

6.3.5 Energy-saving function (EnergySaving)

The *EnergySaving* structure contains the control and status variables of the energy-saving function.

INFORMATION

The energy-saving function can only be activated when the axis is not enabled. When using a double axis, both axes must not be enabled. In this case, the control signal always sets both axes to energy-saving mode.

The following control and status variables are available:

Interface in the IEC Editor

9007219941252491

IN

Variable name	Description
xActivateStandBy	Data type – BOOL
	• TRUE – Activate energy-saving mode • FALSE – Deactivate energy-saving mode

OUT

Variable name	Description
xStandByActive	Data type – BOOL
	• TRUE – Energy-saving mode active • FALSE – Energy-saving mode not active

25967886/EN – 03/2020

6.3.6 Brake control (Brake)

⚠ WARNING

Uncontrolled system behavior when the brake is released.

Severe or fatal injuries, e.g. due to falling hoist.

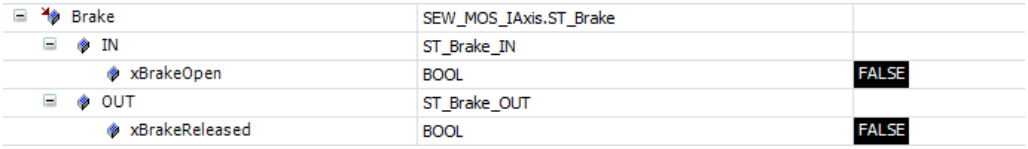
- Make sure that the system does not automatically start moving when the brake is released.
- Make sure that in hoist applications unintentional lowering of the load is prevented by suitable protective measures.

The *Brake* structure contains control variables and status variables for controlling the brake in special cases, e.g. if a drive is not supplied with current (is in controller inhibit) and is to be moved manually.

In the configuration of the axis in MOVISUITE®, set the parameter "Release brake with inhibited output stage –enable?" under "Drive functions" > "FCB 01 Output stage enable (FCB 01 Output stage inhibit)" to *Yes*. The brake can then be released by setting *xBrakeOpen* to the value "TRUE".

The following control and status variables are available:

Interface in the IEC Editor



20686516235

IN

Variable name	Description
xBrakeOpen	Data type – BOOL
	• TRUE – Release the brake. If the output stage of the device is inhibited (e.g. via *xInhibit* = "TRUE"), the user can use this variable to accurately control the release (opening) of the brake. • FALSE – Apply the brake.

OUT

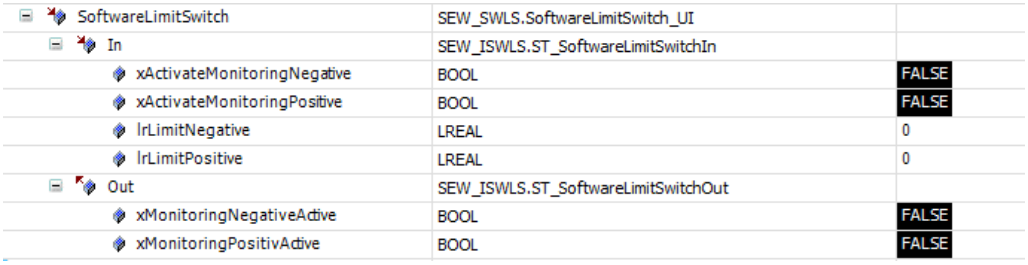
Variable name	Description
xBrakeReleased	Data type: BOOL
	• TRUE – The brake is released. • FALSE – The brake is applied.

25967886/EN – 03/2020

6.3.7 Software limit switches (SoftwareLimitSwitch)

The *SoftwareLimitSwitch* structure contains the control and status variables of the software limit switches.

Interface in the IEC Editor



9007223531035915

IN

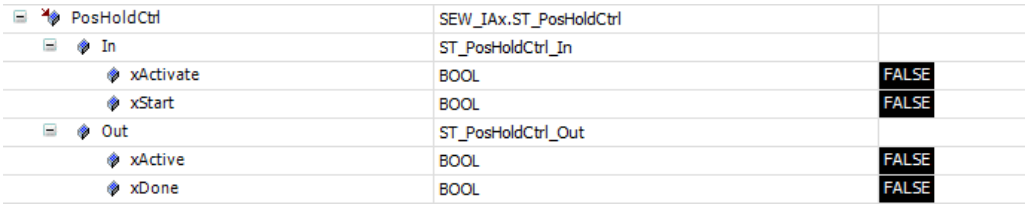
Variable name	Description
xActivateMonitoring-Negative	Data type – BOOL
	• TRUE – Activate monitoring of software limit switch in negative direction • FALSE – Deactivate monitoring of software limit switch in negative direction
xActivateMonitoring-Positive	Data type – BOOL
	• TRUE – Activate monitoring of software limit switch in positive direction • FALSE – Deactivate monitoring for a software limit switch in a positive direction.
lrLimitNegative	Data type: LREAL – Floating-point number
	Position of the software limit switch in negative direction (in user units)
lrLimitPositive	Data type: LREAL – Floating-point number
	Position of the software limit switch in positive direction (in user units)

OUT

Variable name	Description
xMonitoringNegat-iveActive	Data type – BOOL
	• TRUE – Monitoring for software limit switch in a negative direction is active. • FALSE – Monitoring of software limit switch in negative direction is not active
xMonitoringPositive-Active	Data type: BOOL
	• TRUE – Monitoring for software limit switch in a positive direction is active. • FALSE – Monitoring for software limit switch in a positive direction is not active.

6.3.8 Position hold control (PosHoldCtrl)

Interface in the IEC Editor



32294011019

IN

Variable name	Description
xActivate	Data type – BOOL
	• TRUE – Activate function • FALSE – Deactivate function
xStart	Data type – BOOL
	• TRUE – Start function • FALSE – Stop function

Note:

If *xActivate* and *xStart* have the value "TRUE", position hold control (FCB 19) is activated on the inverter. If one of these signals is removed, FCB 19 is deactivated.

Out

Variable name	Description
xActive	Data type – BOOL
	• TRUE – Activated • FALSE – Not activated
xDone	Data type – BOOL
	• TRUE – Process is complete • FALSE – Process is not complete

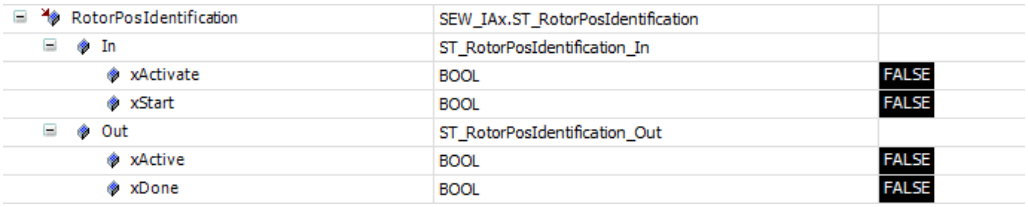
Note:

Both signals are "TRUE" as long as position hold control (FCB 19) is activated on the inverter. Otherwise, the signals output the value "FALSE".

25967886/EN – 03/2020

6.3.9 Rotor position identification (RotorPosIdentification)

Interface in the IEC Editor



32294128651

IN

Variable name	Description
xActivate	Data type – BOOL
	• TRUE – Activate function • FALSE – Deactivate function
xStart	Data type – BOOL
	• TRUE – Start function • FALSE – Stop function

Note:

Rotor position identification (FCB 18) is activated on the inverter if *xActivate* and *xStart* have the value "TRUE" If one of these signals is removed, FCB 18 is deactivated.

Out

Variable name	Description
xActive	Data type – BOOL
	• TRUE – Activated • FALSE – Not activated
xDone	Data type – BOOL
	• TRUE – Process is complete • FALSE – Process is not complete

25967886/EN – 03/2020

6.4 Speed control (Velocity)

The *Velocity* structure contains control and status variables for the "Speed control" operating mode. It is used to control the FCB 05 of the connected inverter. This operating mode is used to move drives at a specified speed. If no other operating mode is active when setting *xActivate* to *FALSE*, the inverter switches to FCB 26.

Interface in the IEC Editor

Velocity	SEW_UIDM.ModeVelocity_UI	
In	SEW_IDM.ST_ModeVelocityIn2	
xActivate	BOOL	FALSE
xStart	BOOL	FALSE
lrVelocity	LREAL	0
lrAcceleration	LREAL	0
lrDeceleration	LREAL	0
uiJerkTime	UINT	0
Out	SEW_IDM.ST_ModeVelocityOut	
xActive	BOOL	FALSE
lrActualVelocity	LREAL	0
xInVelocity	BOOL	FALSE

25294005515

6.4.1 IN

Changes to these variables are applied immediately.

The dynamic parameters are scaled in user units.

Variable name	Description
xActivate	Data type – BOOL
	• TRUE – Activate • FALSE – Stop If *xActivate* is set to "FALSE", the inverter switches to a stop FCB and stops with the ramp assigned to the stop FCB.
xStart	Data type – BOOL
	• TRUE – Start function • FALSE – Stop function
lrVelocity	Data type: LREAL – Floating-point number
	Speed (velocity) When the specified target speed has been reached, any change to *lrVelocity* takes immediate effect. The specified acceleration *lrAcceleration* or deceleration *lrDeceleration* is used to reach the newly specified speed.
lrAcceleration	Data type: LREAL – Floating-point number
	Acceleration
lrDeceleration	Data type: LREAL – Floating-point number
	Deceleration

25967886/EN – 03/2020

Variable name	Description
uiJerkTime	Data type: UINT
	Jerk time in ms No jerk time is transmitted in the "AuxVelocity" and "AuxPositioning" process data profiles. The matching configuration is required if a jerk time is to be processed. See chapter "Processing jerk time" (→ 🗎 72).

6.4.2 OUT

The dynamic parameters are scaled in user units.

Variable name	Description
xActive	Data type – BOOL
	• TRUE – Activated • FALSE – Not activated
lrActualVelocity	Data type: LREAL – Floating-point number
	Actual speed
xInVelocity	Data type – BOOL
	• TRUE – Axis has reached the specified speed • FALSE – Axis has not reached the specified speed

Note:

The feedback *xInVelocity* is controlled by the setting of the parameters in the "Comparison of setpoint/actual speed values" section in the "FCB 07 Torque control" (→ 🗎 20) menu of the configuration.

25967886/EN – 03/2020

6.5 Torque control (Torque)

The *Torque* structure contains control and status variables for the "Torque control" operating mode. It is used to control the FCB 07 of the connected inverter. This operating mode is used to set the drives to a specified torque. If no other operating mode is active when setting *xActivate* to *FALSE*, the inverter switches to FCB 26.

The resulting behavior depends on the load. Without load, the drive accelerates to the specified maximum or minimum speed. When doing so, the effective torque is reduced to such a degree that the specified speed limits are not exceeded.

The specified torque is only achieved if the load is accordingly large.

Interface in the IEC Editor

Torque	SEW_UIDM.ModeTorque_UI	
In	SEW_IDM.ST_ModeTorqueIn2	
xActivate	BOOL	FALSE
xStart	BOOL	FALSE
lrVelocityMax	LREAL	0
lrVelocityMin	LREAL	0
lrTorque	LREAL	0
uiJerkTime	UINT	0
Out	SEW_IDM.ST_ModeTorqueOut	
xActive	BOOL	FALSE
lrActualTorque	LREAL	0
xAtTorqueLimit	BOOL	FALSE

25294022539

6.5.1 IN

Changes to these variables are applied immediately.

The dynamic parameters are scaled in user units.

Variable name	Description
xActivate	Data type – BOOL
	• TRUE – Activate • FALSE – Stop If *xActivate* is set to "FALSE", the inverter switches to a stop FCB and stops with the ramp assigned to the stop FCB.
xStart	Data type – BOOL
	• TRUE – The specified torque takes effect. If the load of the drive is too low and the drive reaches the specified speed limit, the effective torque will be reduced accordingly. • FALSE – The torque is withdrawn.
lrVelocityMax	Data type: LREAL – Floating-point number
	Maximum speed in user units. This limit takes effect when the drive moves in positive direction when a positive torque is specified. The actual speed of the drive is always smaller than "lrMaxVelocity".

25967886/EN – 03/2020

Variable name	Description
lrVelocityMin	Data type: LREAL – Floating-point number
	Minimum speed in user units. This limit takes effect when the drive moves in negative direction when a negative torque is specified. The actual speed of the drive is always greater than "– lrMinVelocity".
lrTorque	Data type: LREAL – Floating-point number
	Torque in percent of the nominal motor torques (1.0 = 100% M_N)
uiJerkTime	Data type: UINT
	Jerk time in ms No jerk time is transmitted in the "AuxVelocity" and "Aux-Positioning" process data profiles. The matching configuration is required if a jerk time is to be processed. See chapter "Processing jerk time" (→ 🗎 72).

6.5.2 OUT

The dynamic parameters are scaled in user units.

Variable name	Description
xActive	Data type – BOOL
	• TRUE – Activated • FALSE – Not activated
lrActualTorque	Data type: LREAL – Floating-point number
	Current torque of the nominal motor torque in percent (1.0 = 100% M_N)
xAtTorqueLimit	Data type – BOOL
	• TRUE – The specified torque is reached. • FALSE – The specified torque is not reached.

Comment:

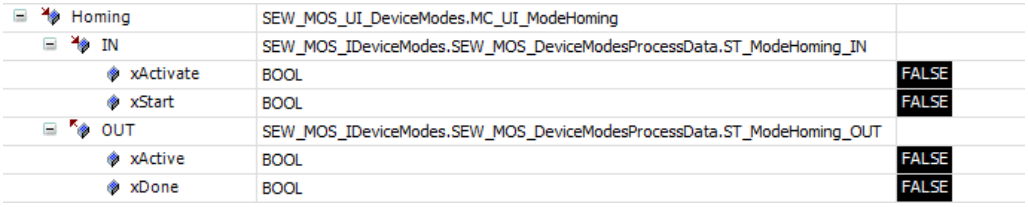
The feedback *xAtTorqueLimit* is controlled by the setting of the parameters in the "Set-point/actual torque comparison" section in the "FCB 07 Torque control" (→ 🗎 20) menu of the configuration.

25967886/EN – 03/2020

6.6 Reference travel (Homing)

The *Homing* structure contains control and status variables for the reference travels. It is used to control the FCB 12 of the connected inverter. The parameters for *FCB 12 Reference travel* set on the inverter apply (e.g. reference travel type, etc.).

Interface in the IEC Editor



20686538507

6.6.1 IN

Variable name	Description
xActivate	Data type – BOOL
	• TRUE – Activate • FALSE – Stop If *xActivate* is set to "FALSE", the inverter switches to a stop FCB and stops with the ramp assigned to the stop FCB.
xStart	Data type – BOOL
	• TRUE – Start function • FALSE – Stop function

Note:

To start a reference travel, set both signals to "TRUE". To stop an ongoing reference travel, set one of the two signals to "FALSE".

6.6.2 OUT

Variable name	Description
xActive	Data type – BOOL
	• TRUE – Activated • FALSE – Not activated
xDone	Data type – BOOL
	• TRUE – Process is complete • FALSE – Process is not complete

Note:

The feedback signal *xActive* becomes "TRUE" only when both input signals *xActivate* and *xStart* are set to "TRUE".

25967886/EN – 03/2020

6.7 Position control (Positioning)

The *Positioning* structure contains control and status variables for the "Position control" operating mode. It is used to control the FCB 09 of the connected inverter. This operating mode is used to move drives to specified positions. If no other operating mode is active when setting *xActivate* to *FALSE*, the inverter switches to FCB 26.

Interface in the IEC Editor

Positioning	SEW_UIDM.ModePositioning_UI	
In	SEW_IDM.ST_ModePositioningIn2	
xActivate	BOOL	FALSE
xStart	BOOL	FALSE
lrPosition	LREAL	0
lrVelocity	LREAL	0
lrAcceleration	LREAL	0
lrDeceleration	LREAL	0
uiJerkTime	UINT	0
Config	SEW_IDM.ST_ModePositioningConfig	
eMode	E_POSITIONINGMODE	eAbsolute
Out	SEW_IDM.ST_ModePositioningOut	
xActive	BOOL	FALSE
lrActualPosition	LREAL	0
xInPosition	BOOL	FALSE

25294026763

6.7.1 IN

Changes to these variables are applied immediately.

The dynamic parameters are scaled in user units.

Variable name	Description
xActivate	Data type – BOOL
	• TRUE – Activate • FALSE – Stop If *xActivate* is set to "FALSE", the inverter switches to a stop FCB and stops with the ramp assigned to the stop FCB.
xStart	Data type – BOOL
	• TRUE – Start function • FALSE – Stop function
lrPosition	Data type: LREAL – Floating-point number
	Position
lrVelocity	Data type: LREAL – Floating-point number
	Speed (velocity) When the specified target speed has been reached, any change to *lrVelocity* takes immediate effect. The specified acceleration *lrAcceleration* or deceleration *lrDeceleration* is used to reach the newly specified speed.
lrAcceleration	Data type: LREAL – Floating-point number
	Acceleration
lrDeceleration	Data type: LREAL – Floating-point number
	Deceleration

Variable name	Description
uiJerkTime	Data type: UINT
	Jerk time in ms No jerk time is transmitted in the "AuxVelocity" and "AuxPositioning" process data profiles. The matching configuration is required if a jerk time is to be processed. See chapter "Processing jerk time" (→ 🗎 72).

Note:

When activating this operating mode, make sure that speed, acceleration and deceleration are specified greater than zero, else the inverter signals a fault.

6.7.2 Config

Variable name	Description
eMode	Data type – E_POSITIONINGMODE
	• eAbsolute: absolute positioning • eRelative: relative positioning • eModuloAbsolutePositive: absolute modulo positioning in positive direction • eModuloAbsoluteNegative: absolute modulo positioning in negative direction • eModuloAbsoluteShortestWay: absolute modulo positioning at shortest distance • eModuloAbsoluteWithoutRef: absolute positioning without referencing

25967886/EN – 03/2020

6.7.3 OUT

The dynamic parameters are scaled in user units.

Variable name	Description
xActive	Data type – BOOL
	• TRUE – Activated • FALSE – Not activated
lrActualPosition	Data type: LREAL – Floating-point number
	Actual position
xInPosition	Data type: BOOL
	• TRUE – Specified position is reached • FALSE – Specified position is not reached

Comment:

The feedback *xInPosition* is controlled by setting the "In position" parameters in the "FCB 09 Position control" (→ 🗎 21) menu of the configuration.

25967886/EN – 03/2020

6.8 Jog

The *Jog* operating mode generates a "jogging" profile using the specified configuration.

The following control and status variables are available:

Interface in the IEC Editor

Jog	SEW_UIDM.ModeJog_UI	
In	SEW_IDM.ST_ModeJogIn	
xActivate	BOOL	FALSE
xJogPositive	BOOL	FALSE
xJogNegative	BOOL	FALSE
lrVelocity	LREAL	0
lrAcceleration	LREAL	0
lrDeceleration	LREAL	0
uiJerkTime	UINT	0
Out	SEW_IDM.ST_ModeJogOut	
xActive	BOOL	FALSE
lrActualVelocity	LREAL	0

32293726731

6.8.1 IN

Changes to these variables are applied immediately.

The dynamic parameters are scaled in user units.

Variable name	Description
xActivate	Data type – BOOL
	• TRUE – Activate • FALSE – Stop If *xActivate* is set to "FALSE", the inverter switches to a stop FCB and stops with the ramp assigned to the stop FCB.
xJogPositive	Data type – BOOL
	• TRUE – Start profile generation of the operating mode for moving in positive direction using the value of the specified speed *lrVelocity* The specified acceleration takes effect. • FALSE – Stop profile generation of the operating mode. The specified deceleration takes effect.
xJogNegative	Data type – BOOL
	• TRUE – Start profile generation of the operating mode for moving in a negative direction using the value of specified speed *lrVelocity*. The specified acceleration is effective. • FALSE – Stop profile generation of the operating mode The specified deceleration takes effect.
lrVelocity	Data type: LREAL – Floating-point number
	Speed (velocity)
lrAcceleration	Data type: LREAL – Floating-point number
	Acceleration

25967886/EN – 03/2020

Variable name	Description
lrDeceleration	Data type: LREAL – Floating-point number
	Deceleration
uiJerkTime	Data type: UINT
	Jerk time in ms No jerk time is transmitted in the "AuxVelocity" and "AuxPositioning" process data profiles. The matching configuration is required if a jerk time is to be processed. See chapter "Processing jerk time" (→ 🗎 72).

Note:

The drive stops if *xJogPositive* and *xJogNegative* are selected simultaneously.

6.8.2 Out

Variable name	Description
xActive	Data type – BOOL
	• TRUE – Activated • FALSE – Not activated
lrActualVelocity	Data type: LREAL – Floating-point number
	Actual speed

25967886/EN – 03/2020

7 Diagnostics

7.1 Monitor

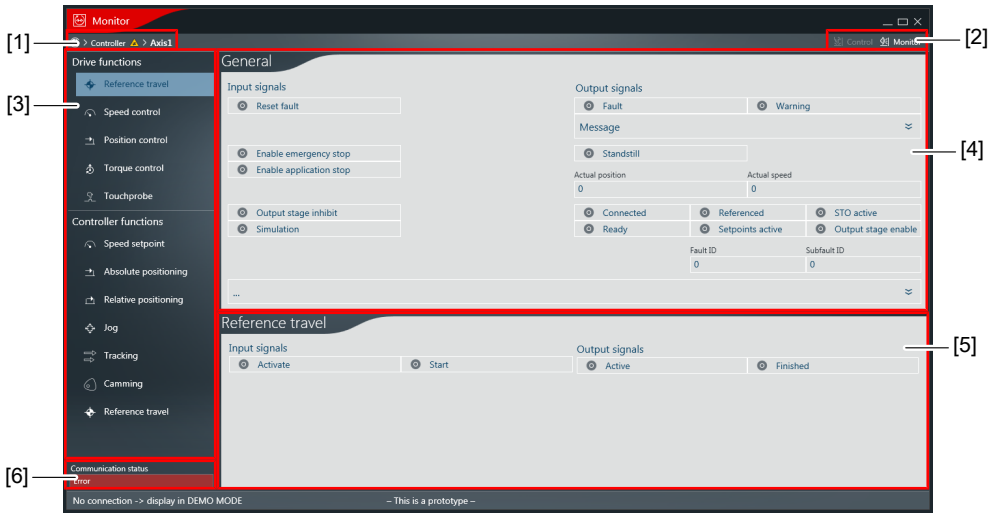
INFORMATION

Only available with "Advanced" permission level.

The MultiMotion monitor is a tool in the MOVISUITE® engineering software for monitoring and controlling axes or axis groups.

In MOVISUITE®, click [Monitor] in the Tools menu to open the tool from the context menu of a specific node.

The user interface of the MultiMotion monitor includes the following areas:



9007223554947083

No.	Area	Description
[1]	Device path and name	Path and name of the axis/axis group
[2]	Mode switching	• "Control" mode: Operating mode for setting control bits and control values. Control via the IEC program is ignored. • "Monitor" mode: Operating mode for monitoring the input and output values. In this mode, the monitor only has read-only rights and displays the values set by the controller.
[3]	Main menu	Main menu for opening the configuration menus of the functions available in each case, such as drive functions, controller functions, or functions of software modules.
[4]	General	Input and output signals of general settings
[5]	Functions	Depending on the function selected from the main menu, this area displays the corresponding input and output signals.
[6]	Communication status	Status of the communication link

25967886/EN – 03/2020

8 Application examples

8.1 General requirements for moving an axis

The following requirements must be met for moving an axis and consequently for all the examples described in the following sections:

- The *xError* variable reports back "FALSE". There is no error.
- The *usiErrorID* and *usiErrorSubID* variables report back the value "0".
- The *xInitDone* variable reports back "TRUE". The axis is now initialized.
- The *xConnected* variable reports back "TRUE". The inverter communicates with the MOVI-C® CONTROLLER.

Interface in the IEC Editor

xError	BOOL	FALSE
xWarning	BOOL	FALSE
udiMessageID	UDINT	16#00000000
sInstancePath	STRING(Constants.gc_udiLengthPathName)	'Controller.Application.SEW_GVL.Interface_Axis2'
_stLocalVar_ErrorBasic	ST_LocalVariables_Basic	
sAdditionalText	STRING(Constants.gc_udiLengthAdditionalText)	''
_stLocalVar_ControlSource	ST_LocalVariables_ControlSource	
eControlSource	E_CONTROLSOURCE	USERINTERFACE
itfAccessControl	SEW_IAccCtrl.IAccessControl	16#00A47BA0
xReset	BOOL	FALSE
xGetAccessControl	BOOL	TRUE
xControlActive	BOOL	TRUE
xInitDone	BOOL	TRUE
Basic	SEW_IAx.ST_Basic	
Inverter	SEW_IAx.ST_Inverter	
In	ST_Inverter_In	
Out	ST_Inverter_Out	
xConnected	BOOL	TRUE
xPowered	BOOL	FALSE
xReady	BOOL	TRUE
xReferenced	BOOL	TRUE
xSetpointActive	BOOL	FALSE
xSafeStop	BOOL	FALSE
xPositionValid	BOOL	FALSE
wDigitalInputs	WORD	1
lrActualTorque	LREAL	0
eActualInverterMode	E_INVERTERMODE	EmergencyStop
usiErrorID	USINT	0
usiErrorSubID	USINT	0

25299799691

25967886/EN – 03/2020

8.1.1 Enable axis

The axis must be enabled before it can move. To enable the axis, set the following variables in the *Basic.In* structure:

- Set *xEnable_EmergencyStop* to "TRUE".
- Set *xEnable_ApplicationStop* to "TRUE".

The *xPowered* variable reports back "TRUE" in the *Inverter.Out* structure.

Interface in the IEC Editor

xError	BOOL	FALSE
xWarning	BOOL	FALSE
udiMessageID	UDINT	16#00000000
sInstancePath	STRING(Constants.gc_udiLengthPathName)	'Controller.Application.SEW_GVL.Interface_Axis2'
_stLocalVar_ErrorBasic	ST_LocalVariables_Basic	
sAdditionalText	STRING(Constants.gc_udiLengthAdditionalText)	''
_stLocalVar_ControlSource	ST_LocalVariables_ControlSource	
eControlSource	E_CONTROLSOURCE	USERINTERFACE
itfAccessControl	SEW_IAccCtrl.IAccessControl	16#00A47BA0
xReset	BOOL	FALSE
xGetAccessControl	BOOL	TRUE
xControlActive	BOOL	TRUE
xInitDone	BOOL	TRUE
Basic	SEW_IAx.ST_Basic	
Inverter	SEW_IAx.ST_Inverter	
In	ST_Inverter_In	
Out	ST_Inverter_Out	
xConnected	BOOL	TRUE
xPowered	BOOL	FALSE
xReady	BOOL	TRUE
xReferenced	BOOL	TRUE
xSetpointActive	BOOL	FALSE
xSafeStop	BOOL	FALSE
xPositionValid	BOOL	FALSE
wDigitalInputs	WORD	1
lrActualTorque	LREAL	0
eActualInverterMode	E_INVERTERMODE	EmergencyStop
usiErrorID	USINT	0
usiErrorSubID	USINT	0

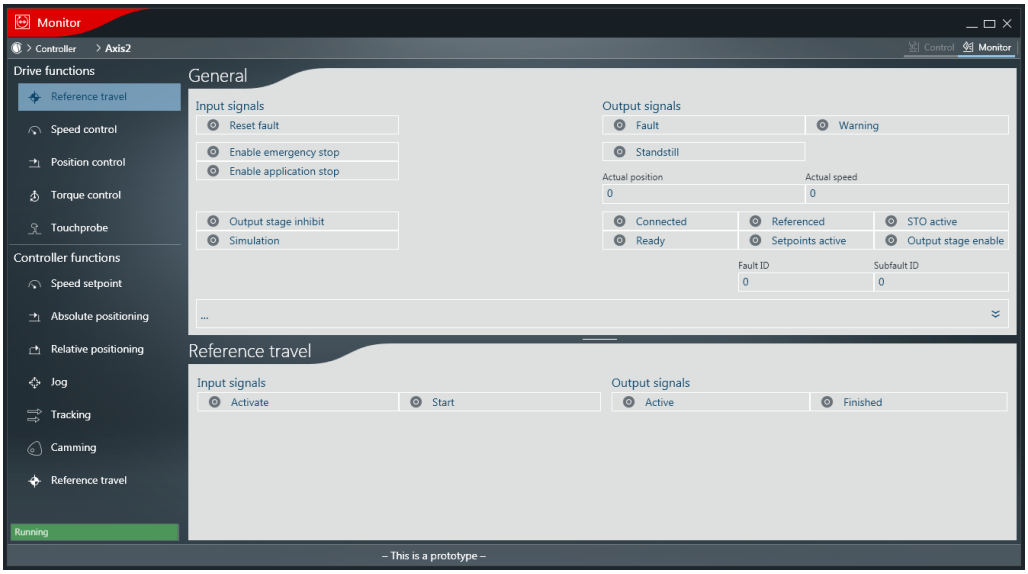
25299799691

25967886/EN – 03/2020

8.2 Controlling the "Reference travel" operating mode

Do the following to control the "Reference travel" operating mode:

1. In the "General" section, activate the "Enable emergency stop" input signal.
2. In the "General" section, activate the "Enable application stop" input signal.
   ⇨ The "Output stage enable" output signal in the "General" section is active.
3. In the "Reference travel" section, activate the "Activate" input signal.
4. In the "Reference travel" section, activate the "Start" input signal.
   ⇨ The output signals "Active" and "Finished" in the "Reference travel" section are activated.
   ⇨ The "Referenced" output signal in the "General" section is activated.



25295431691

8.3 Controlling the "Speed control" operating mode

8.3.1 From the IEC program

The following application example shows how to control the "Velocity" operating mode from the IEC program.

Control

Control the axis in the *Velocity* structure as follows:

- To activate the operating mode, set *xActivate* to "TRUE".
- Define the setpoints for *lrVelocity*, *lrAcceleration*, *lrDeceleration*.
- To start the operating mode, set *xStart* to "TRUE".

Interface in the IEC Editor

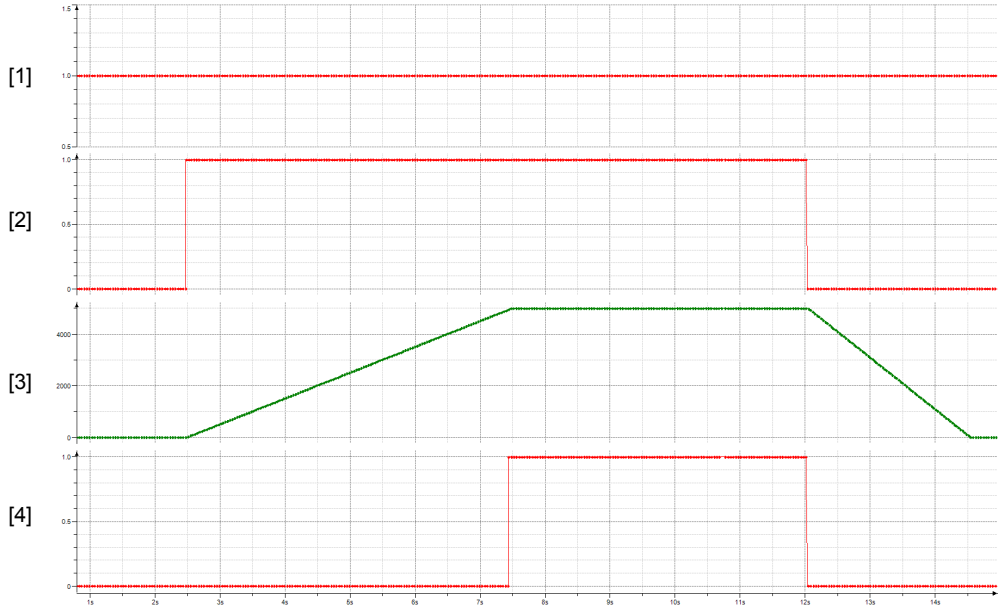
Velocity	SEW_UIDM.ModeVelocity_UI	
In	SEW_IDM.ST_ModeVelocityIn2	
xActivate	BOOL	TRUE
xStart	BOOL	TRUE
lrVelocity	LREAL	5000
lrAcceleration	LREAL	1000
lrDeceleration	LREAL	2000
uiJerkTime	UINT	0
Out	SEW_IDM.ST_ModeVelocityOut	
xActive	BOOL	TRUE
lrActualVelocity	LREAL	5000
xInVelocity	BOOL	TRUE

25300190603

Note:

When *xStart* has a value of "TRUE", the system accelerates with *lrAcceleration*. When the target speed is reached, you will receive the feedback *xInVelocity* = "TRUE" in the *OUT* structure. When *xStart* has a value of "FALSE", the system stops with *lrDeceleration*.

Trace recording



25346435467

[1] Velocity.In.xActivate
[2] Velocity.In.xStart
[3] Velocity.In.lrVelocity
[4] Velocity.Out.xInVelocity

Information:

The speed curve is trapezoidal as the drive was moved without jerk time.

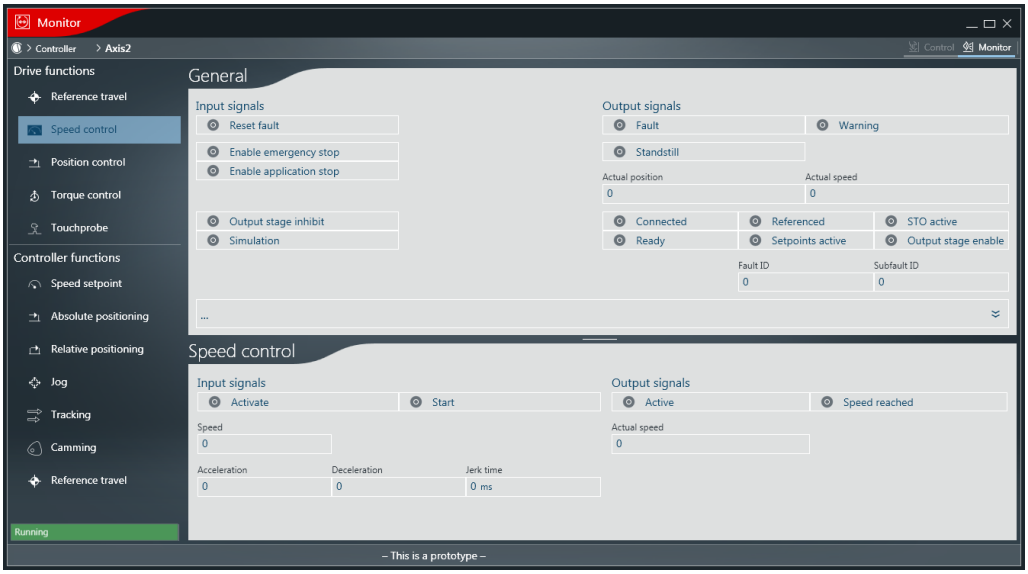
25967886/EN – 03/2020

8.3.2 Via monitor

Do the following to control the "Speed control" operating mode via the monitor:

✓ The "Monitor" (→ 🖹 54) is open.

1. In the "General" section, activate the "Enable emergency stop" input signal.
2. In the "General" section, activate the "Enable application stop" input signal.
   ⇨ The "Output stage enable" output signal in the "General" section is active.
3. Specify values for the input signals "Speed", "Acceleration", "Deceleration", and "Jerk time" in the "Speed control" section.
4. In the "Speed control" section, activate the "Activate" input signal.
5. In the "Speed control" section, activate the "Start" input signal.
   ⇨ The "Active" output signal in the "Speed control" section is active.
   ⇨ When the specified speed is reached, the "Speed reached" output signal is activated in the "Speed control" section.



25295488779

8.4 Controlling the "Position control" operating mode

8.4.1 From the IEC program

The following application example shows how to control the "Positioning" operating mode from the IEC program.

Control

Control the axis in the *Positioning* structure as follows:

- To activate the operating mode, set *xActivate* to "TRUE".
- Define the setpoints for*lrPosition*, *lrVelocity*, *lrAcceleration*, and *lrDeceleration*.
- To start the operating mode, set *xStart* to "TRUE".

Interface in the IEC Editor

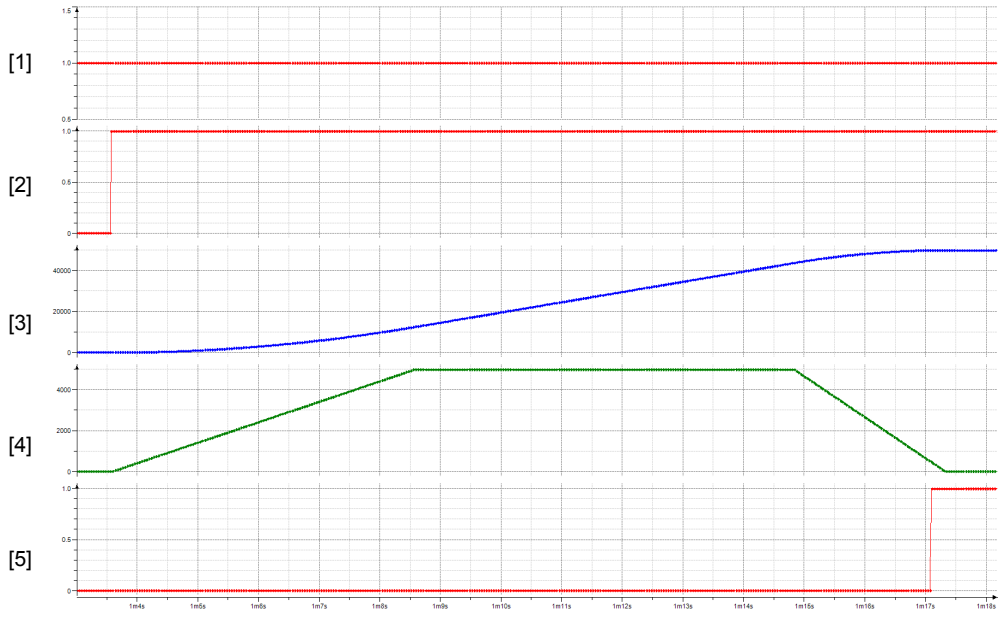
Positioning	SEW_UIDM.ModePositioning_UI	
In	SEW_IDM.ST_ModePositioningIn2	
xActivate	BOOL	TRUE
xStart	BOOL	TRUE
lrPosition	LREAL	50000
lrVelocity	LREAL	5000
lrAcceleration	LREAL	1000
lrDeceleration	LREAL	2000
uiJerkTime	UINT	0
Config	SEW_IDM.ST_ModePositioningConfig	
eMode	E_POSITIONINGMODE	eAbsolute
Out	SEW_IDM.ST_ModePositioningOut	
xActive	BOOL	TRUE
lrActualPosition	LREAL	50000
xInPosition	BOOL	TRUE

25300779787

Note:

When *xStart* has a value of "TRUE", the system accelerates with *lrAcceleration*. When the target position is reached, you will receive the feedback *xInPosition* = "TRUE" in the *OUT* structure. When *xStart* has a value of "FALSE", the system stops with *lrDeceleration*. The speed curve is trapezoidal as the drive was moved without jerk time.

Trace recording



25345085451

[1] Positioning.In.xActivate
[2] Positioning.In.xStart
[3] Positioning.In.lrPosition
[4] Positioning.In.lrVelocity
[5] Positioning.Out.xInPosition

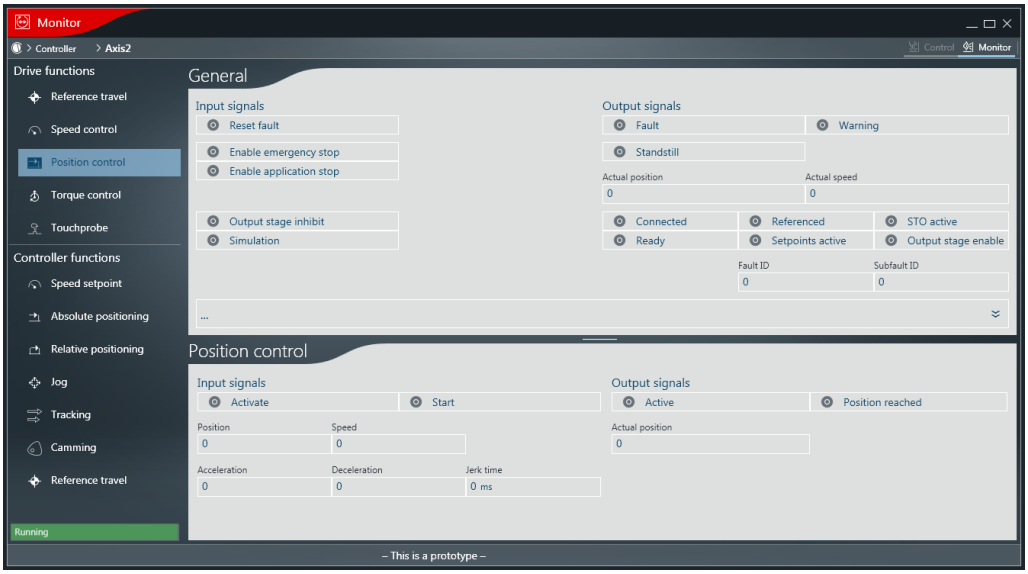
25967886/EN – 03/2020

8.4.2 Via monitor

Do the following to control the "Position control" operating mode via the monitor:

✓ The "Monitor" (→ 🖹 54) is open.

1. In the "General" section, activate the "Enable emergency stop" input signal.
2. In the "General" section, activate the "Enable application stop" input signal.
   ⇨ The "Output stage enable" output signal in the "General" section is active.
3. Specify values for the input signals "Position", "Speed", "Acceleration", "Deceleration", and "Jerk time" in the "Position control" section.
4. In the "Position control" section, activate the "Activate" input signal.
5. In the "Position control" section, activate the "Start" input signal.
   ⇨ The "Active" output signal in the "Position control" section is active.
   ⇨ When the specified position is reached, the "Position reached" output signal is activated in the "Position control" section.



25295569803

8.5 Controlling the "Torque control" operating mode

8.5.1 From the IEC program

Torque > 0

The following application example shows how to control the "Torque control" operating mode in the IEC program.

Control

Control the axis in the *Torque* structure as follows:

- To activate the operating mode, set *xActivate* to "TRUE".
- Define the setpoints for *lrVelocityMax*, *lrVelocityMin*, and *lrTorque*.
- To start the operating mode, set *xStart* to "TRUE".

Interface in the IEC Editor

Torque	SEW_UIDM.ModeTorque_UI	
In	SEW_IDM.ST_ModeTorqueIn2	
xActivate	BOOL	TRUE
xStart	BOOL	TRUE
lrVelocityMax	LREAL	1000
lrVelocityMin	LREAL	1000
lrTorque	LREAL	0.2
uiJerkTime	UINT	0
Out	SEW_IDM.ST_ModeTorqueOut	
xActive	BOOL	TRUE
lrActualTorque	LREAL	0.2
xAtTorqueLimit	BOOL	TRUE

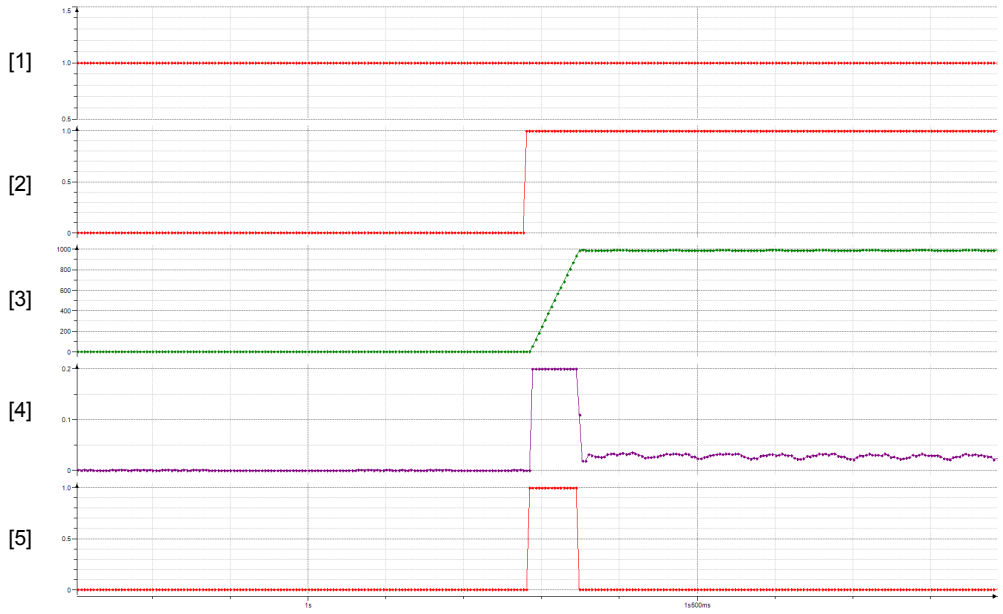
25300827915

Note:

When the setpoint torque is reached, you will receive the feedback *xAtTorqueLimit* = "TRUE" in the *OUT* structure.

Trace recording

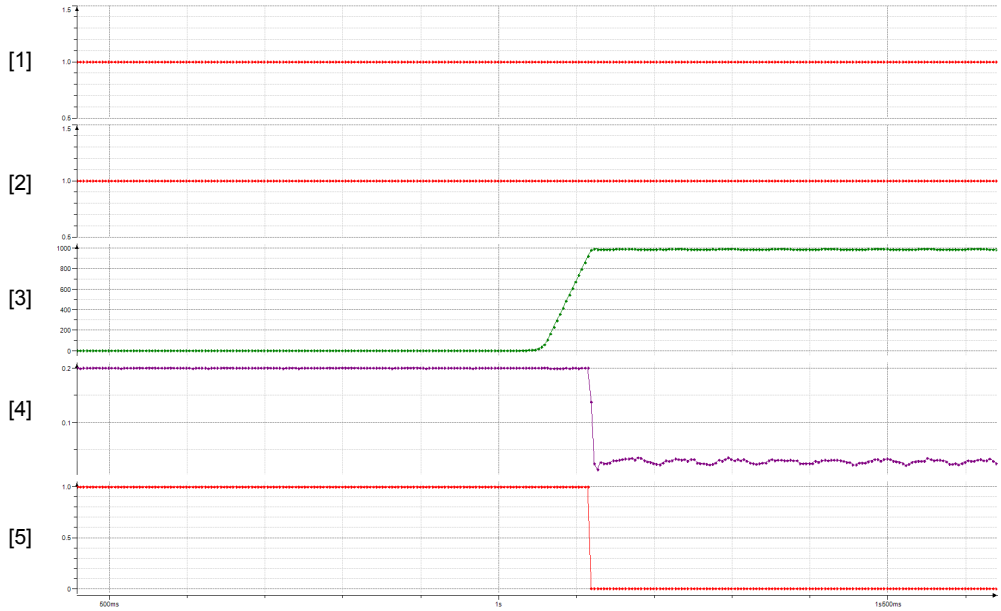
The following trace recording shows the profile of the current speed and the current torque with the values set at start as described above. The current torque is set immediately to the required torque when starting. The *xAtTorqueLimit* feedback is "TRUE". The axis accelerates with the torque in positive direction. When the maximum speed is reached, the current torque is reduced and the *xAtTorqueLimit* feedback is set to "FALSE".



25346425867

[1] Torque.In.xActivate
[2] Torque.In.xStart
[3] Torque.In.lrVelocityMax
[4] Torque.In.lrTorque
[5] Torque.In.xAtTorqueLimit

In the following trace recording, the axis is first held at standstill by the load; the current speed is zero and the *xAtTorqueLimit* feedback is "TRUE". Then the load suddenly disappears and the axis accelerates in positive direction. The current torque is reduced as soon as the maximum speed is reached. The *xAtTorqueLimit* feedback is accordingly set to "FALSE".



25346417035

[1] Torque.In.xActivate
[2] Torque.In.xStart
[3] Torque.In.lrVelocityMax
[4] Torque.In.lrTorque
[5] Torque.In.xAtTorqueLimit

Note:

The scenario under consideration here occurs, for example, in the case of material tearing in a winder operated in torque control mode. It is important that the axis speed does not exceed the *lrVelocityMax* limit.

Torque < 0

The following application example shows how to control the "Torque control" operating mode in the IEC program.

Control

Control the axis in the *Torque* structure as follows:

- To activate the operating mode, set *xActivate* to "TRUE".
- Define the setpoints for *lrVelocityMax*, *lrVelocityMin*, and *lrTorque*.
- To start the operating mode, set *xStart* to "TRUE".

25967886/EN – 03/2020

Interface in the IEC Editor

Torque	SEW_UIDM.ModeTorque_UI	
In	SEW_IDM.ST_ModeTorqueIn2	
xActivate	BOOL	TRUE
xStart	BOOL	TRUE
lrVelocityMax	LREAL	1000
lrVelocityMin	LREAL	1000
lrTorque	LREAL	-0.2
uiJerkTime	UINT	0
Out	SEW_IDM.ST_ModeTorqueOut	
xActive	BOOL	TRUE
lrActualTorque	LREAL	-0.2
xAtTorqueLimit	BOOL	TRUE

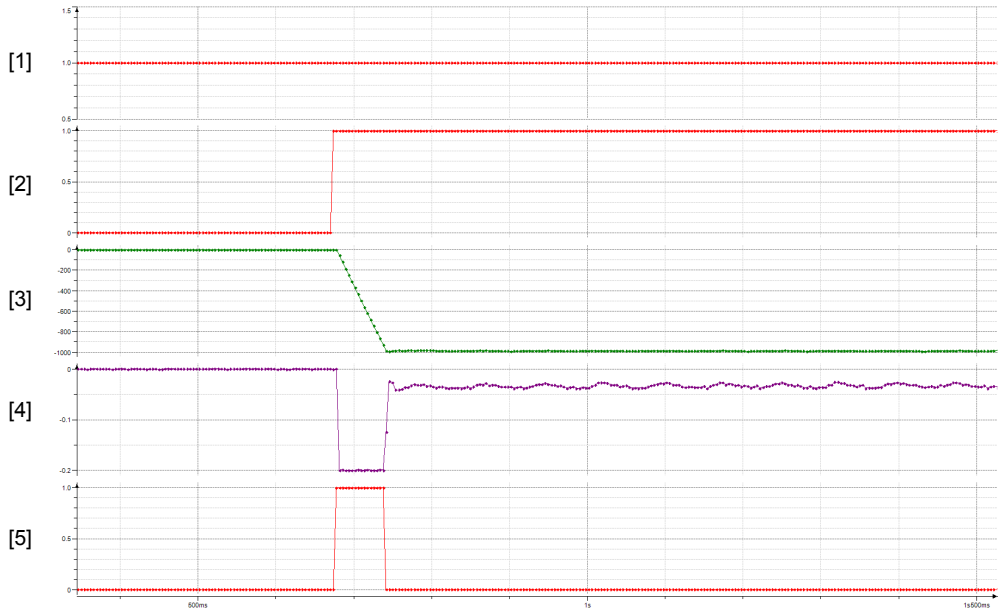
25300879243

Note:

When the setpoint torque is reached, you will receive the feedback *xAtTorqueLimit* = "TRUE" in the *OUT* structure.

Trace recording

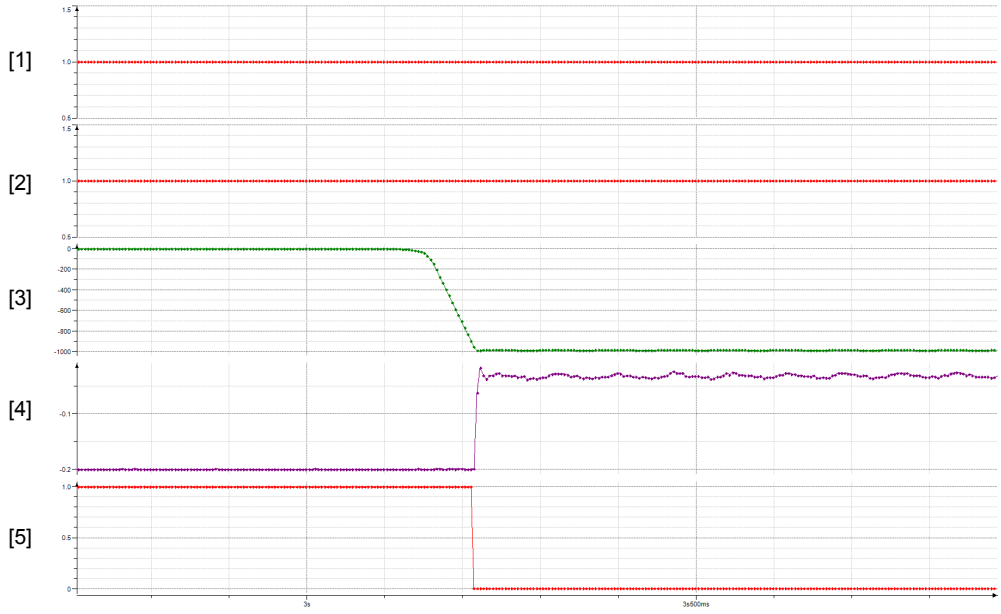
The following trace recording shows the profile of current speed and current torque when starting the operating mode with the values set as described above. The current torque is set immediately to the required torque when starting. The *xAtTorqueLimit* feedback is "TRUE". The axis accelerates with the torque in negative direction. When the maximum speed is reached, the current torque is reduced and the *xAtTorqueLimit* status variable is set to "FALSE".



25346396171

[1] Torque.In.xActivate
[2] Torque.In.xStart
[3] Torque.In.lrVelocityMax
[4] Torque.In.lrTorque
[5] Torque.In.xAtTorqueLimit

In the following trace recording, the axis is initially held at standstill by the load. The current speed is zero and the *xAtTorqueLimit* status variable is set to "TRUE". Then the load suddenly disappears and the axis accelerates in negative direction. The current torque is reduced as soon as the maximum speed is reached. The *xAtTorque-Limit* status variable is accordingly set to "FALSE".



25346388875

[1] Torque.In.xActivate
[2] Torque.In.xStart
[3] Torque.In.lrVelocityMax
[4] Torque.In.lrTorque
[5] Torque.In.xAtTorqueLimit

Note:

The scenario under consideration here occurs, for example, in the case of material tearing in a winder operated in torque control mode. It is important that the axis speed does not exceed the *lrVelocityMin* limit.

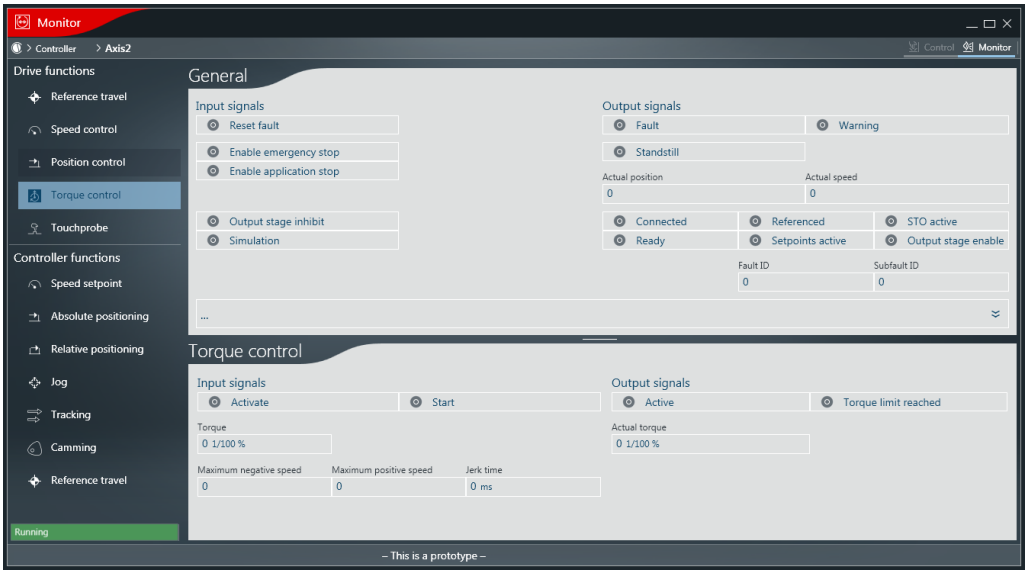
25967886/EN – 03/2020

8.5.2 Via monitor

Do the following to control the "Torque control" operating mode via the monitor:

✓ The "Monitor" (→ 🖹 54) is open.

1. In the "General" section, activate the "Enable emergency stop" input signal.
2. In the "General" section, activate the "Enable application stop" input signal.
   ⇨ The "Output stage enable" output signal in the "General" section is active.
3. Specify values for the input signals "Torque", "Maximum negative speed", "Maximum positive speed", and "Jerk time" in the "Torque control" section.
4. In the "Torque control" section, activate the "Activate" input signal.
5. In the "Torque control" section, activate the "Start" input signal.
   ⇨ The "Active" output signal in the "Position control" section is active.
   ⇨ When the specified torque is reached, the "Torque reached" output signal is activated in the "Torque control" section.



25295574539

8.6 Extended function

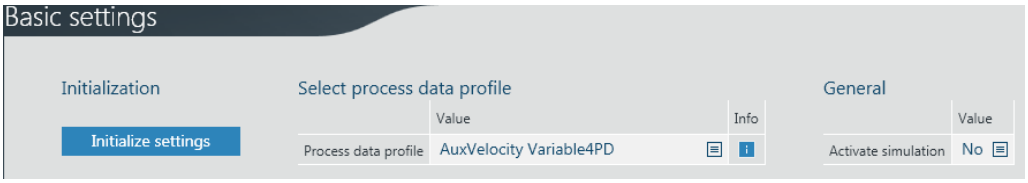
8.6.1 Extending the process data profile

Auxiliary axes use process data profiles that are as lean as possible and that contain all data required for the supported operating modes.

Control information and status information, for example a touchprobe function or jerk time processing are not included in the process data profile. The process data profile must be extended to obtain these functions.

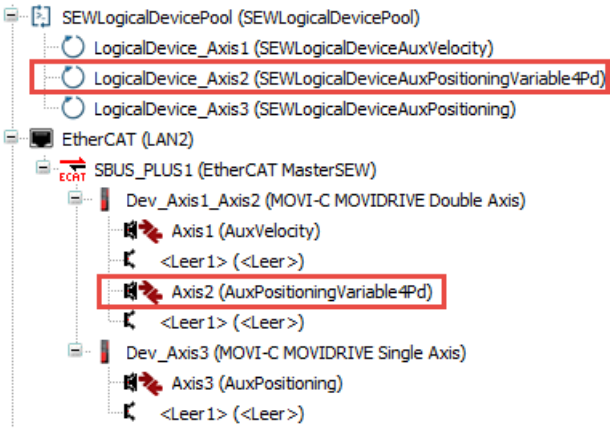
Do the following to extend the process data profile:

1. In MOVISUITE®, click the software module.
   ⇨ The configuration menus of the software module are displayed.
2. Choose the matching option as the value for the process data profile in the "Basic settings" menu under "Process data profile". For 4 additional process data words for exchanging data between MOVI-C® CONTROLLER and inverter, select "AuxVelocity Variable4PD" or "AuxPositioning Variable4PD". For 8 additional process data words for the MultiMotion Auxiliary Positioning software module "AuxPositioning Variable 8PD".



25301420171

3. Click the [Initialize settings] button.
4. Update the IEC project to set the new process data profile for the axis.



25301833739

5. Connect the additionally available process data words accordingly in the inverter. For a detailed description of applications, refer to the chapters "Processing jerk time" (→ 🖹 72) and "Processing touchprobe" (→ 🖹 75).

8.6.2 Processing digital inputs and outputs of the inverter

The state of the digital inputs of the inverter is sent to the MOVI-C® CONTROLLER via status word 3 (bits 0 to 5). The state of the digital outputs of the inverter is controlled via control word 2 (bits 0 to 3). Both control words are not included in the "AuxVelocity" process data profile. All process data profiles with 8 or more process data words transmit this information by default.

25967886/EN – 03/2020

Connection of process data

Proceed as follows:

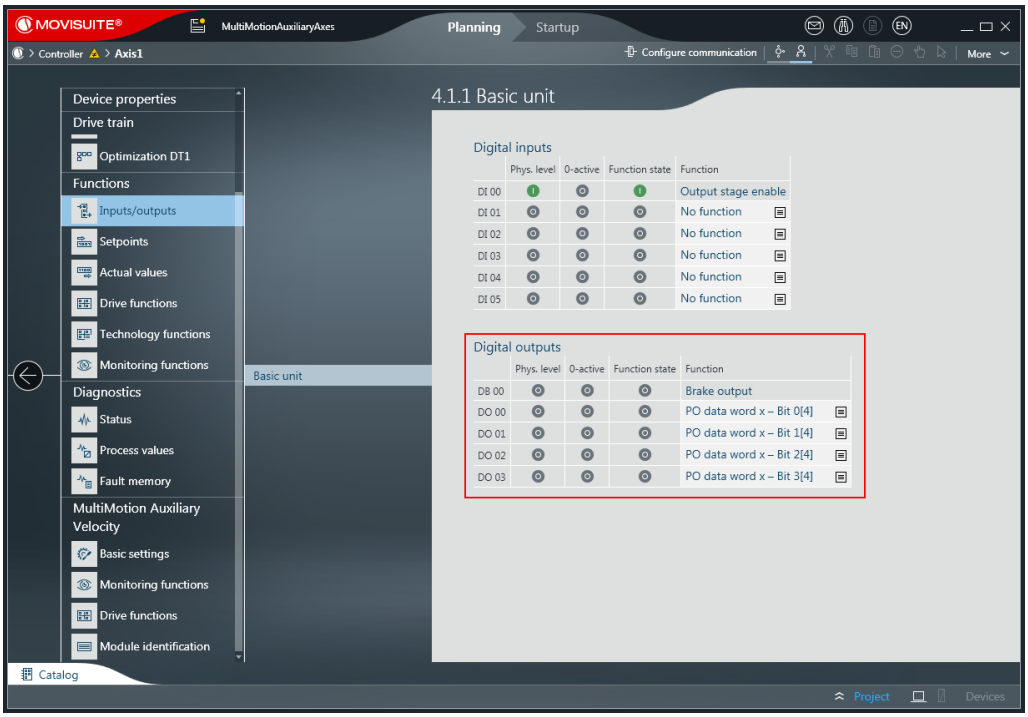
✓ The process data profile is extended to include additional process data words. For instructions, refer to the chapter "Extending the process data profile" (→ 🗎 68).

1. In MOVISUITE®, click the software module.

   ⇨ The configuration menus of the software module are displayed.

2. Under "Functions" in the main menu, open the "Inputs/outputs" menu and its sub-menu "Basic unit".

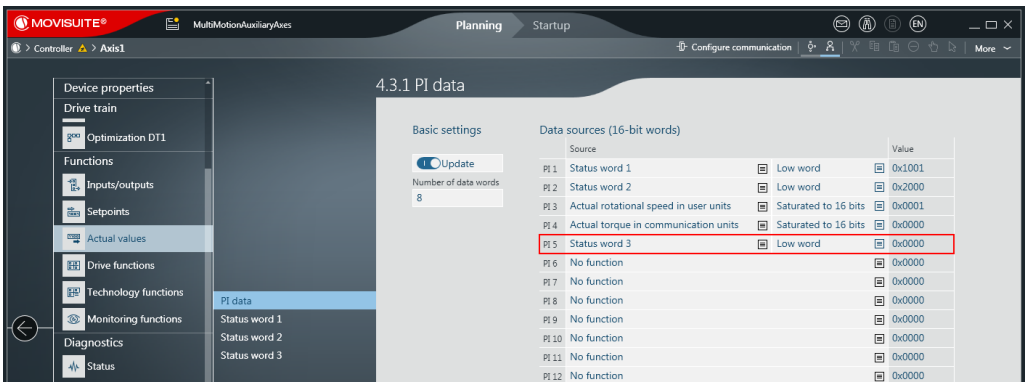
3. To connect the digital outputs with PO data word 5, define the function "PO data word x – bits 0-3[4]" for DO 00-03 (see screenshot). As the process data in the inverter are numbered beginning from 0, the word [4] is accessed during connection.



25327061259

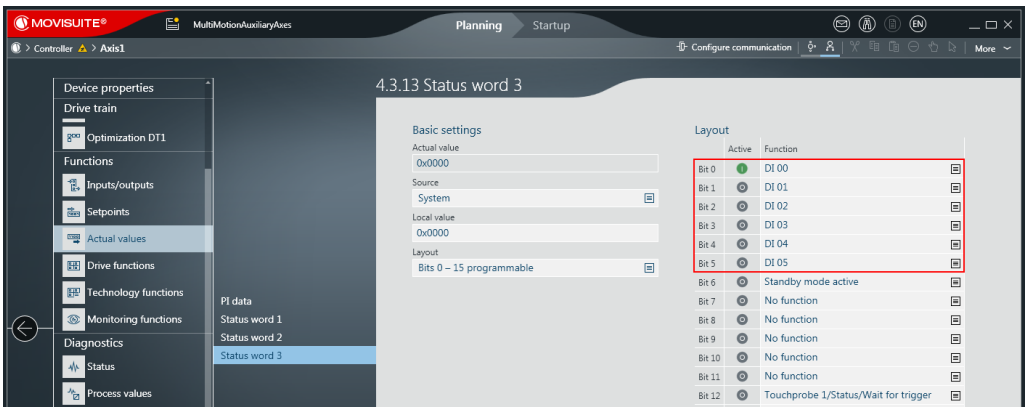
4. Under "Functions" in the main menu, open the "Actual values" menu and its sub-menu "PI data".

5. To connect "Status word 3" with "PE data word 5", define "Status word 3" as "Low word" as the source for "PI 5".



25327064331

6. Now open submenu "Status word 3" of the "Actual values" menu.
7. To connect the digital inputs with "Status word 3", define the functions DI 00-05 for bits 0 to 5 in the "Layout" section.



25327067403

Programming in the IEC program

1. Exit the configuration menu and open the IEC project in the IEC Editor.
2. In the IEC project in action *User_PRG.ReadActualValues*, transfer process data word 5 (byte offset = 8; see screenshot) using the method *GetPdIn* of the logic device assigned to the axis.
   ⇨ The lower 6 bits in the *wReadBuffer* variable in the *User_PRG* program represent the state of the digital inputs.
3. In the IEC project in action *User_PRG.WriteSetpointValues*, transfer process data word 5 using the method *SetPdOut* of the logic device assigned to the axis to the value defined in the *wWriteBuffer* variable.
   ⇨ The lower 4 bits in the *wWriteBuffer* variable in the *User_PRG* program represent the state of the digital outputs.

User_PRG

```
1  PROGRAM User_PRG
2  VAR_OUTPUT
3      xInitDone : BOOL;
4  END_VAR
5  VAR
6
7      eError:         SEW_IDH.E_Error;
8
9      wReadBuffer:    WORD;
10     dwByteToRead:   DWORD := 2;
11
12     wWriteBuffer:   WORD;
13     dwByteToWrite:  DWORD := 2;
14
15 END_VAR
```

User_PRG.ReadActualValues

```
1  eError := LogicalDevice_Axis1.GetPdIn   (
2                                          pbBuffer := ADR(wReadBuffer),
3                                          dwBufferLen := SIZEOF(wReadBuffer),
4                                          dwOffset := 8,
5                                          pdwBytesToRead := ADR(dwByteToRead)
6                                          );
```

User_PRG.WriteSetpointValues

```
1  eError := LogicalDevice_Axis1.SetPdOut  (
2                                          pbBuffer := ADR(wWriteBuffer),
3                                          dwBufferLen := SIZEOF(wWriteBuffer),
4                                          dwOffset := 8,
5                                          pdwBytesToWrite := ADR(dwByteToWrite),
6                                          bMask := 0
7                                          );
```

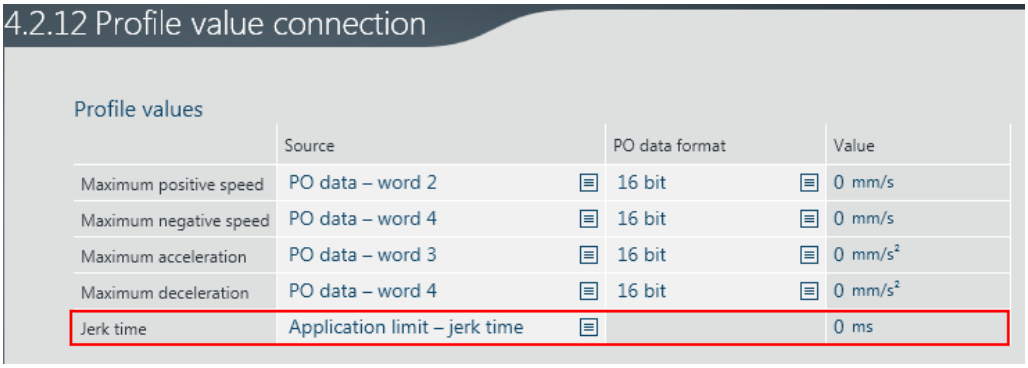
25327070859

8.6.3 Processing jerk time

For processing a jerk time, the jerk time must be mapped to one of the free process data words in the inverter. Proceed as follows:

✓ The process data profile was extended to include additional process data words. For instructions, refer to the chapter "Extending the process data profile" (→ 🗎 68).

1. In MOVISUITE®, click the software module.
   ⇨ The configuration menus of the software module are displayed.
2. Under "Functions" in the main menu, open the "Setpoints" menu and its submenu "Profile value connection".
3. Select the required additional process data word for the "Jerk time".

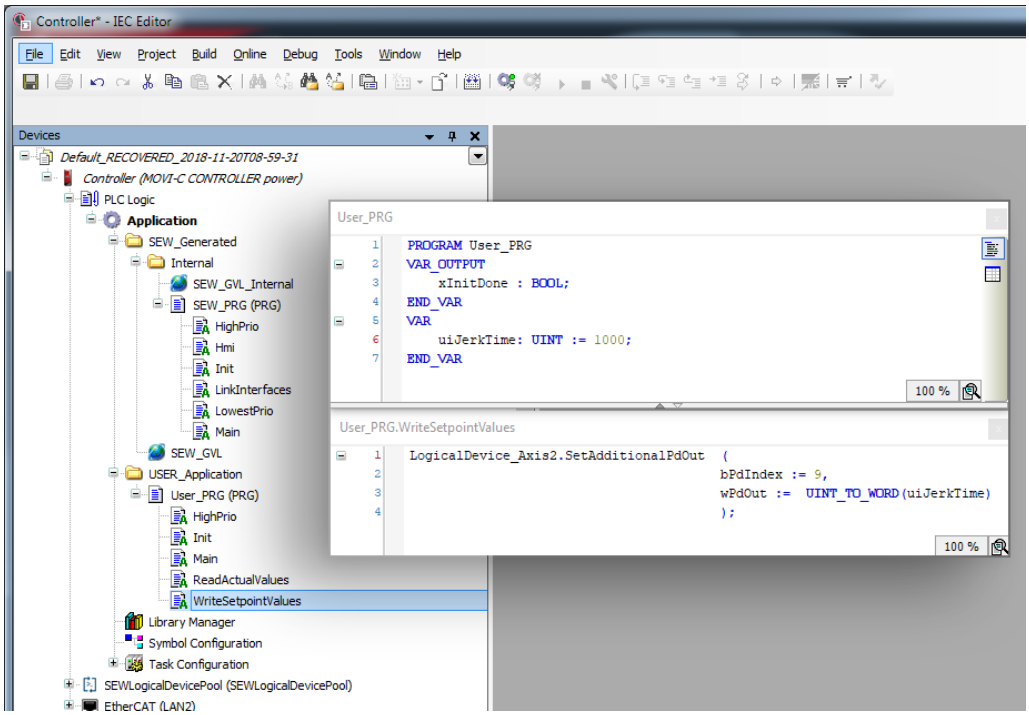


25301837195

4. Exit the configuration menu and open the IEC project in the IEC Editor.

25967886/EN – 03/2020

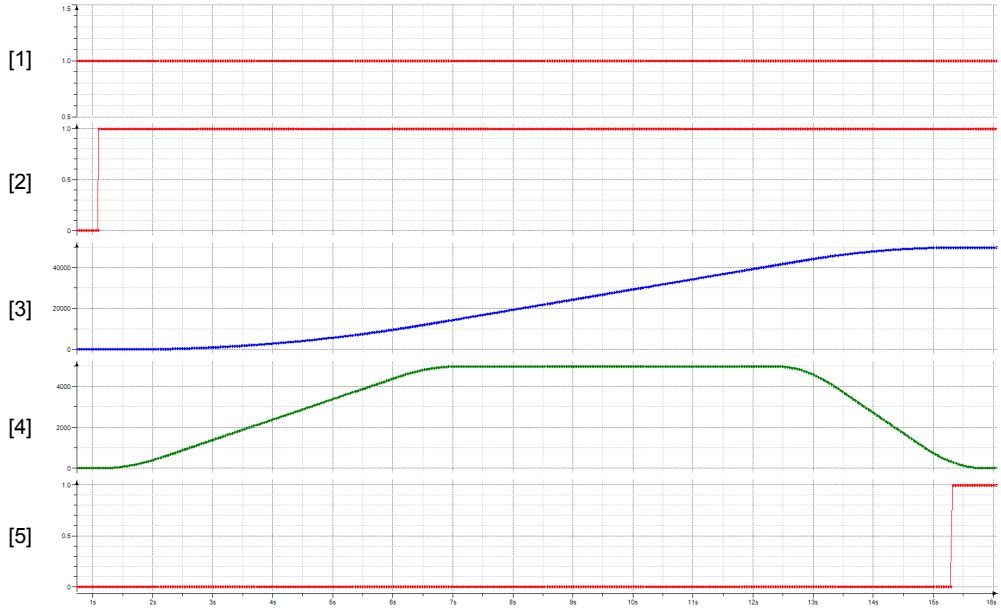
5. In the IEC project, transfer the jerk time to the set process data word using the method *SetAdditionalPdOut* of the logic device assigned to the axis. To do so, first declare a variable *uiJerkTime* in the *User_PRG (PRG)* program and then set the required jerk time in this variable.
6. Write the variable to the required process data word using the action *User_PRG.WriteSetpointValues*.



25301839755

Trace recording

The following trace recording shows an example of the position and speed profiles as described in chapter "Controlling the "Position control" operating mode" (→ 🗎 60) with a jerk time of 1000 ms:



25345089547

[1] Velocity.In.xActivate
[2] Velocity.In.xStart
[3] Velocity.In.lrPosition
[4] Velocity.In.lrVelocity
[5] Velocity.Out.xInPosition

Comment:

The speed profile is no longer trapezoidal. The edges are rounded, which means the acceleration does not "jump" to the set value but gradually reaches the set value via the jerk time. Positioning takes accordingly longer. In this case, it takes twice the jerk time.

25967886/EN – 03/2020

8.6.4 Processing touchprobe

Connection of process data

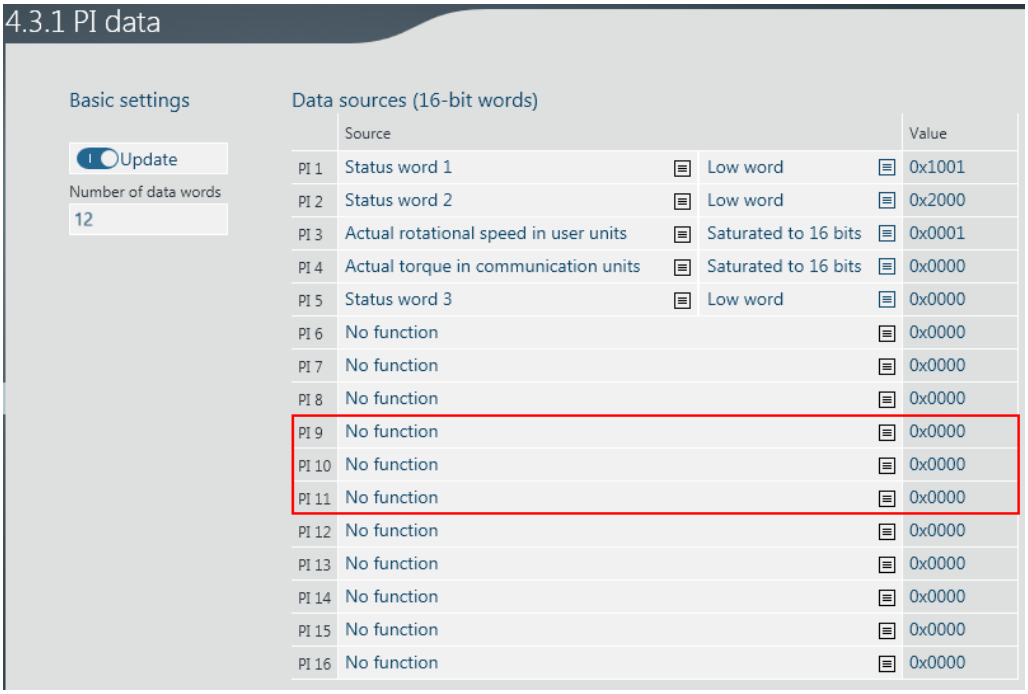
For processing the touchprobe function, the following control and status information must be mapped to the free process data words in the inverter:

- Touchprobe position
- Touchprobe counter
- "Activate" control bit
- "Active" status bit

Proceed as follows:

✓ The process data profile was extended to include additional process data words. For instructions, refer to the chapter "Extending the process data profile" (→ 🖹 68).

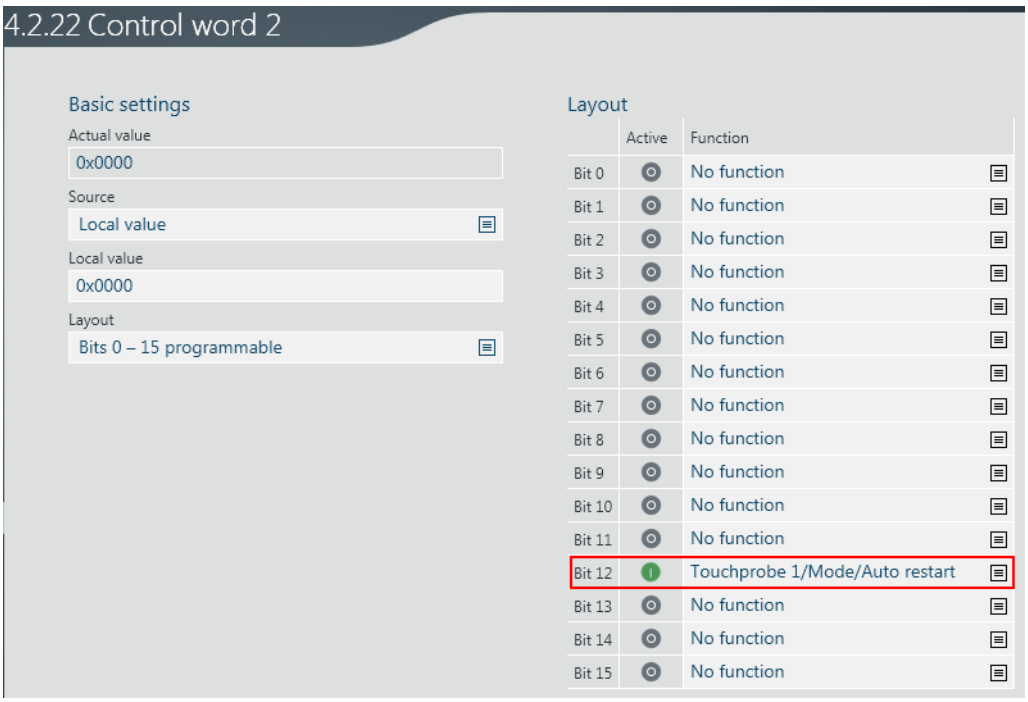
1. In MOVISUITE®, click the software module.
   ⇨ The configuration menus of the software module are displayed.
2. Under "Functions" in the main menu, open the "Actual values" menu and its submenu "PI data".
3. To connect the touchprobe position, define "Low word" of the position for "Touchprobe 1/detected value" as the source for "PI 9".
4. To connect the touchprobe position, define "High word" of the position for "Touchprobe 1/detected value" as the source for "PI 10".
5. To connect the touchprobe counter, define "Saturated to 16 bits" for "Touchprobe 1/trigger – counter" as the source for "PI 11".



25302233995

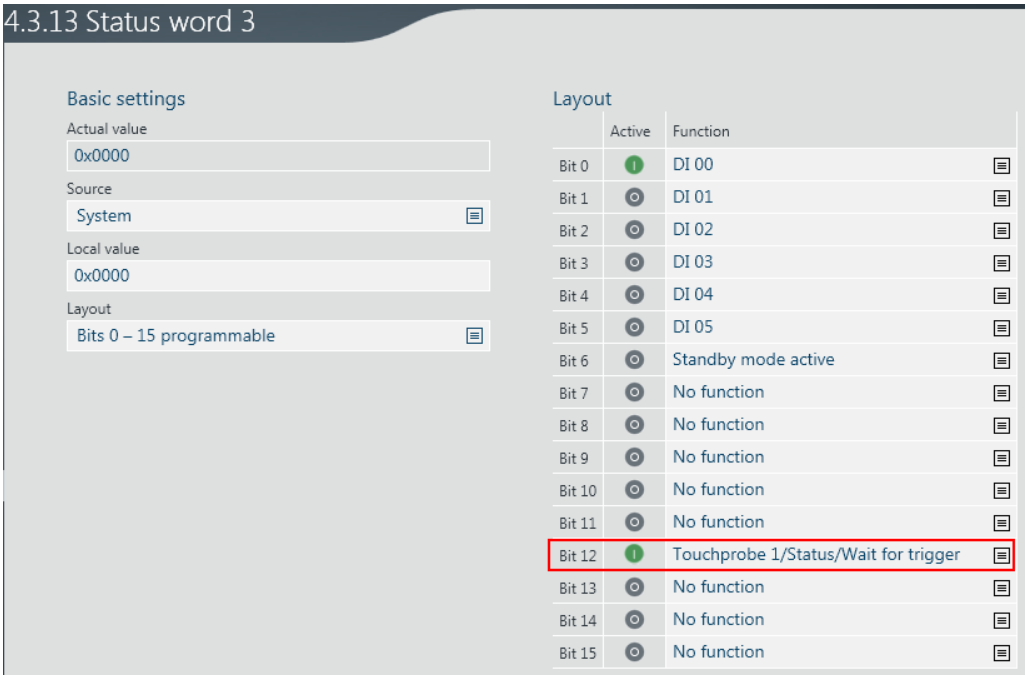
6. Under "Functions" in the main menu, open the "Setpoints" menu and its submenu "Control word 2".

7. To connect "Activate touchprobe" control, define the function "Touchprobe 1/Mode/Auto restart" for "Bit 12".



25302236555

8. Under "Functions" in the main menu, open the "Setpoints" menu and its submenu "Control word 3".
9. To connect the "Touchprobe active" feedback, define the function "Touchprobe 1/Status/Wait for trigger" for "Bit 12".



25302239115

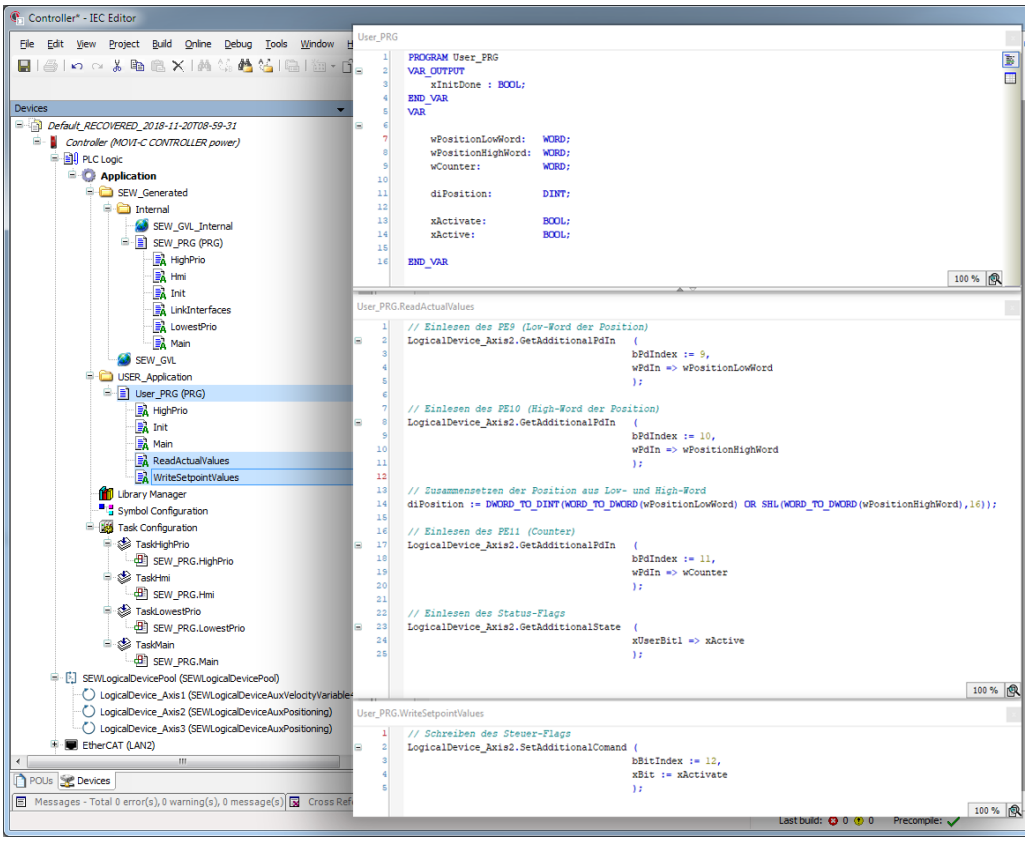
25967886/EN – 03/2020

Programming example in the IEC program

INFORMATION

Access to the additionally configured process data can only be programmed using the methods of the logic device assigned to the axis.

1. Exit the configuration menu and open the IEC project in the IEC Editor.
2. Declare the required local variables as shown.
3. In the action *User_PRG.ReadActualValues*, read process data words 9 to 10 using the *GetAdditionalPdIn()* method. These process data words contain the position value of the touchprobe function, which is generated accordingly.
4. In the action *User_PRG.ReadActualValues*, read the process data word 11 using the *GetAdditionalPdIn()* method. This process data word contains the counter value of the touchprobe function.
5. In the action *User_PRG.ReadActualValues*, read the additionally configured status bit using the *GetAdditionalState()* method. This status bit contains the *xActive* feedback of the touchprobe function.
6. In the action *User_PRG.WriteSetpointValues*, write the additionally configured status bit using the *SetAdditionalCommand()* method. This status bit contains the *xActivate* control signal of the touchprobe function.

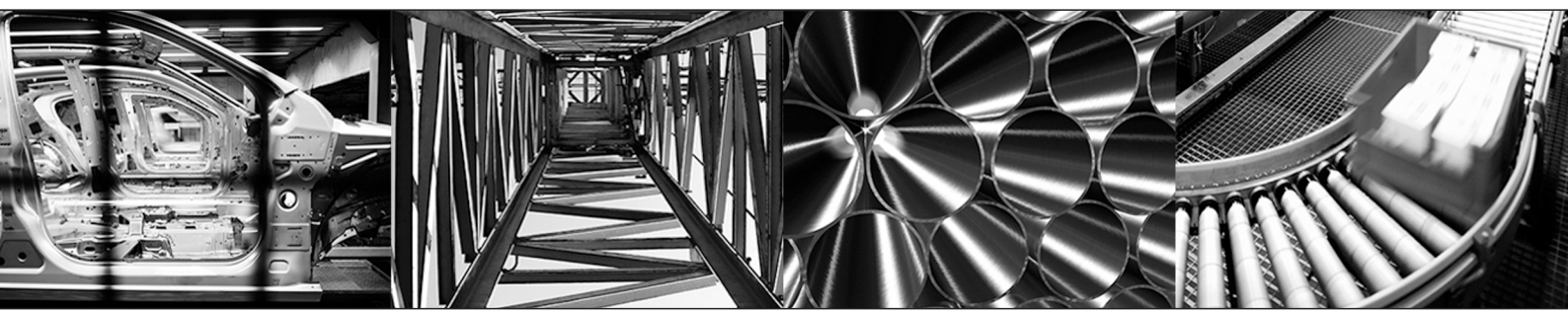


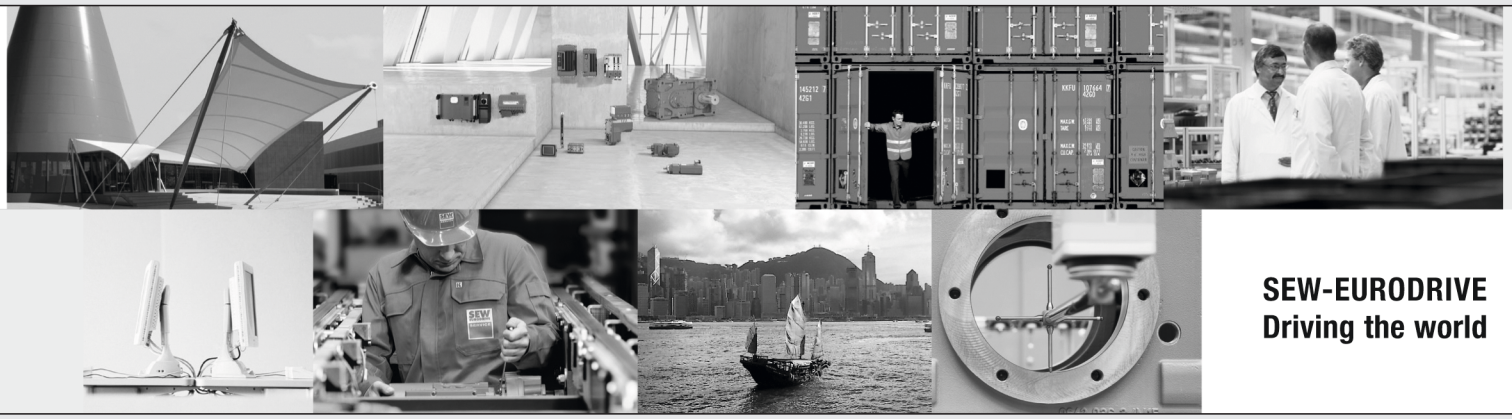
25311918859

⇨ You can activate the touchprobe function using the *xActivate* bit. *xActive* is reported back once the touchprobe function has been activated. The touchprobe counter increments with each touchprobe event. The position value allocated to the event is stored in the local variable *diPosition*.

Index

25967886/EN – 03/20





SEW-EURODRIVE
Driving the world

SEW-EURODRIVE GmbH & Co KG
Ernst-Blickle-Str. 42
76646 BRUCHSAL
GERMANY
Tel. +49 7251 75-0
Fax +49 7251 75-1970
sew@sew-eurodrive.com
➞ **www.sew-eurodrive.com**