



# DOCUMENTATION ISG-kernel

## **Manual** **HLI-Documentation V2.20xx**

Short Description:  
HLI

© Copyright  
ISG Industrielle Steuerungstechnik GmbH  
STEP, Gropiusplatz 10  
D-70563 Stuttgart  
All rights reserved  
[www.isg-stuttgart.de](http://www.isg-stuttgart.de)  
[support@isg-stuttgart.de](mailto:support@isg-stuttgart.de)

Documentation version: 1.20  
13/12/2023

# Preface

## Legal information

---

This documentation was produced with utmost care. The products and scope of functions described are under continuous development. We reserve the right to revise and amend the documentation at any time and without prior notice.

No claims may be made for products which have already been delivered if such claims are based on the specifications, figures and descriptions contained in this documentation.

## Personnel qualifications

---

This description is solely intended for skilled technicians who were trained in control, automation and drive systems and who are familiar with the applicable standards, the relevant documentation and the machining application.

It is absolutely vital to refer to this documentation, the instructions below and the explanations to carry out installation and commissioning work. Skilled technicians are under the obligation to use the documentation duly published for every installation and commissioning operation.

Skilled technicians must ensure that the application or use of the products described fulfil all safety requirements including all applicable laws, regulations, provisions and standards.

## Further information

---

Links below (DE)

<https://www.isg-stuttgart.de/produkte/softwareprodukte/isg-kernel/dokumente-und-downloads>

or (EN)

<https://www.isg-stuttgart.de/en/products/softwareproducts/isg-kernel/documents-and-downloads>

contains further information on messages generated in the NC kernel, online help, PLC libraries, tools, etc. in addition to the current documentation.

## Disclaimer

---

It is forbidden to make any changes to the software configuration which are not contained in the options described in this documentation.

## Trade marks and patents

---

The name ISG®, ISG kernel®, ISG virtuos®, ISG dirigent® and the associated logos are registered and licensed trade marks of ISG Industrielle Steuerungstechnik GmbH.

The use of other trade marks or logos contained in this documentation by third parties may result in a violation of the rights of the respective trade mark owners.

## Copyright

---

© ISG Industrielle Steuerungstechnik GmbH, Stuttgart, Germany.

No parts of this document may be reproduced, transmitted or exploited in any form without prior consent. Non-compliance may result in liability for damages. All rights reserved with regard to the registration of patents, utility models or industrial designs.

# General and safety instructions

## Icons used and their meanings

This documentation uses the following icons next to the safety instruction and the associated text. Please read the (safety) instructions carefully and comply with them at all times.

## Icons in explanatory text

➤ Indicates an action.

⇒ Indicates an action statement.



### **DANGER**

#### **Acute danger to life!**

If you fail to comply with the safety instruction next to this icon, there is immediate danger to human life and health.



### **CAUTION**

#### **Personal injury and damage to machines!**

If you fail to comply with the safety instruction next to this icon, it may result in personal injury or damage to machines.



### **Attention**

#### **Restriction or error**

This icon describes restrictions or warns of errors.



### **Notice**

#### **Tips and other notes**

This icon indicates information to assist in general understanding or to provide additional information.



### **Example**

#### **General example**

Example that clarifies the text.



### **Programing Example**

#### **NC programming example**

Programming example (complete NC program or program sequence) of the described function or NC command.



### **Release Note**

#### **Specific version information**

Optional or restricted function. The availability of this function depends on the configuration and the scope of the version.

# Table of contents

<b>Preface.....</b>	<b>2</b>
<b>General and safety instructions .....</b>	<b>3</b>
<b>1 Introduction.....</b>	<b>9</b>
1.1 Implementation by High-Level Interface (HLI).....	10
1.2 Organisation of the HLI .....	11
1.2.1 Data of a channel\axis .....	13
1.3 Status and display information .....	13
1.4 Control commands .....	14
1.4.1 Control unit with usage check .....	15
1.5 Operation with and without PLC.....	16
<b>2 Axes .....</b>	<b>18</b>
2.1 Definition of axes.....	18
2.2 Definition of coordinate systems .....	18
2.3 Description of the axis-specific interface .....	20
2.3.1 Axis identification .....	20
2.3.2 Axis positions .....	22
2.3.2.1 Axis positions in the PCS.....	22
2.3.2.2 Axis positions in the ACS.....	24
2.3.3 Position of tool centre point in MCS.....	25
2.3.4 Status information of an axis.....	26
2.3.4.1 Drive of an axis .....	34
2.3.4.2 Compensation of an axis .....	39
2.3.4.3 Measurement.....	41
2.3.4.4 Link to ADS.....	42
2.3.5 Axis control commands.....	43
2.3.5.1 Commanding axis couplings.....	60
2.3.5.2 Distance control .....	64
2.3.6 Control commands of a drive .....	70
2.3.7 External axis commanding.....	72
2.3.8 Measuring with external measuring hardware .....	73
<b>3 Spindle.....</b>	<b>76</b>
3.1 Introduction .....	76
3.2 Description of the spindle-specific interface .....	76
3.2.1 Speeds of a spindle .....	76
3.2.2 Positions of a spindle .....	77
3.2.3 Status information of a spindle.....	77
3.2.4 Control commands of a spindle .....	80
3.2.5 External spindle command.....	82
3.2.5.1 Control unit of external spindle command .....	82
3.2.5.2 User data for external spindle commands .....	82
<b>4 Channel .....</b>	<b>86</b>
4.1 Introduction .....	86
4.2 Description of the channel-specific interface.....	87

4.2.1	Status information of a channel .....	87
4.2.1.1	Status information on tool orientation .....	101
4.2.1.2	Diagnosis upload .....	102
4.2.2	Control commands of a channel .....	104
<b>5</b>	<b>PLC .....</b>	<b>128</b>
5.1	Control commands to PLC .....	128
5.1.1	Reset.....	128
5.1.2	Block search .....	130
<b>6</b>	<b>Technology processes.....</b>	<b>133</b>
6.1	Introduction .....	133
6.2	Management of technology functions .....	133
6.3	Elements to manage axis-specific technology control units.....	134
6.3.1	Block-by-block synchronisation (default synchronisation) .....	134
6.3.2	Cross-block synchronisation .....	134
6.4	Elements for managing channel-specific technology control units.....	135
6.4.1	Block-by-block synchronisation (default synchronisation) .....	135
6.4.2	Cross-block synchronisation .....	135
6.5	Data of a technology control unit.....	136
6.5.1	Data of an axis-specific technology control unit.....	136
6.5.2	Data of a channel-specific technology control unit.....	138
6.6	Data of the technology functions.....	141
6.6.1	Data of the M/H function .....	141
6.6.2	Data of the S function .....	143
6.6.3	Data of the T function.....	145
6.6.3.1	User data of tool identification .....	146
<b>7</b>	<b>External variables / V.E variables .....</b>	<b>148</b>
<b>8</b>	<b>Processing types.....</b>	<b>149</b>
8.1	State graph of operation modes.....	149
8.1.1	States of operation mode: Automatic mode.....	151
8.1.2	States of operation mode: Manual block.....	151
8.1.3	States of operation mode: Manual operation mode .....	152
8.1.4	States of operation mode: Homing .....	152
8.2	Control commands/status information for operation modes.....	152
8.2.1	Control Unit.....	153
8.2.2	User data .....	154
8.2.2.1	Requested and commanded user data.....	154
8.2.2.2	Status information .....	158
<b>9</b>	<b>Manual operation mode .....</b>	<b>159</b>
9.1	Status information of manual mode .....	160
9.2	Control commands of manual mode .....	163
9.2.1	Activating control elements for manual mode .....	163
9.2.1.1	User data on activation .....	165
9.2.2	Parametrising of manual mode .....	166
9.2.2.1	Continuous jog mode (by button press).....	167
9.2.2.1.1	Control Unit.....	167

9.2.2.1.2	User data .....	168
9.2.2.2	Incremental jog mode (by button press) .....	169
9.2.2.2.1	Control Unit.....	169
9.2.2.2.2	User data .....	170
9.2.2.3	Handwheel mode.....	172
9.2.2.3.1	Control Unit.....	172
9.2.2.3.2	User data .....	172
9.2.3	Control elements for manual mode.....	173
9.2.3.1	Enforcing a button press.....	173
9.2.3.1.1	Control Unit.....	174
9.2.3.1.2	User data .....	176
9.2.3.2	Rapid traverse velocity during path motion.....	177
9.2.3.2.1	Control Unit.....	178
9.2.3.2.2	User data .....	178
9.2.3.3	Handwheel increments .....	179
<b>10</b>	<b>Safety system .....</b>	<b>181</b>
10.1	Channel-specific interface.....	181
10.1.1	Watchdog mechanism .....	181
<b>11</b>	<b>Management.....</b>	<b>185</b>
11.1	Channel-specific interface.....	185
11.2	Axis-specific interface .....	188
<b>12</b>	<b>Error messages .....</b>	<b>190</b>
12.1	Management data of an error message .....	190
12.2	User data of an error message .....	190
12.2.1	Error message content, body nc program.....	195
12.2.2	Error message content, body machine data .....	196
12.2.3	Error message content, body communication.....	197
12.2.4	Error message content, body RAM disk .....	198
12.2.5	Error message content, body file .....	199
12.2.6	Error message content, body interpreted file list.....	200
12.2.7	Error message content, body binary list.....	201
12.2.8	Error message content, body global channel manager .....	202
12.2.9	Additional error information value 1 - 5.....	204
12.2.9.1	Value range of variable 'Type' of additional error information .....	205
12.2.9.2	Value range of 'Dimension' variable of the additional error information .....	206
12.2.9.3	Value range of 'Significance' variable of additional error information .....	207
<b>13</b>	<b>Messages .....</b>	<b>208</b>
13.1	Control Unit .....	208
13.2	User data.....	209
13.2.1	Requested and commanded user data.....	209
<b>14</b>	<b>Implementation as PLC library.....</b>	<b>210</b>
14.1	Access to HLI .....	210
14.1.1	PLC system TwinCAT.....	210
14.2	Function blocks within the PLC library .....	210
14.2.1	Overview of the PLCopen FBs.....	210
14.2.2	ISG HLI Interface .....	211

<b>15</b>	<b>Programming examples .....</b>	<b>213</b>
15.1	PLC main program frame .....	213
15.1.1	Initialisation function UserInitialisations() .....	213
	<b>Keyword index .....</b>	<b>215</b>
<b>16</b>	<b>Appendix .....</b>	<b>224</b>
16.1	Suggestions, corrections and the latest documentation.....	224

## List of figures

Fig. 1:	CNC-PLC interface .....	9
Fig. 2:	Structuring of the HLI .....	11
Fig. 3:	Channel-specific memory area .....	12
Fig. 4:	Axis-specific/spindle-specific memory area .....	12
Fig. 5:	Transfer of status information .....	13
Fig. 6:	Application of an MC control unit .....	14
Fig. 7:	Interaction, MC control unit and PLC .....	16
Fig. 8:	Coordinate systems .....	19
Fig. 9:	Position values in PC and AC .....	20
Fig. 10:	Interaction of control units and SERCOS control or status words .....	47
Fig. 11:	Example signal profile of measurement with external hardware .....	75
Fig. 12:	Vectors of the tool coordinate system .....	101
Fig. 13:	Interaction between feedhold and NC channel stop .....	110
Fig. 14:	Time sequence of feedhold and NC channel stop .....	111
Fig. 15:	Interaction of input disable and NC channel stop .....	119
Fig. 16:	Time sequence of input disable and NC channel stop .....	120
Fig. 17:	Interaction between BOOLEAN-MC control unit and PLC .....	130
Fig. 18:	Interaction between BOOLEAN-MC control unit and PLC .....	132
Fig. 19:	State graph of an operation mode .....	150
Fig. 20:	Control elements and assignment .....	159
Fig. 21:	Manual mode state transitions .....	160
Fig. 22:	Priorities for watchdog mechanism (example TwinCAT 2) .....	184
Fig. 23:	Management data of the channel-specific interface .....	185
Fig. 24:	FB parameters .....	211



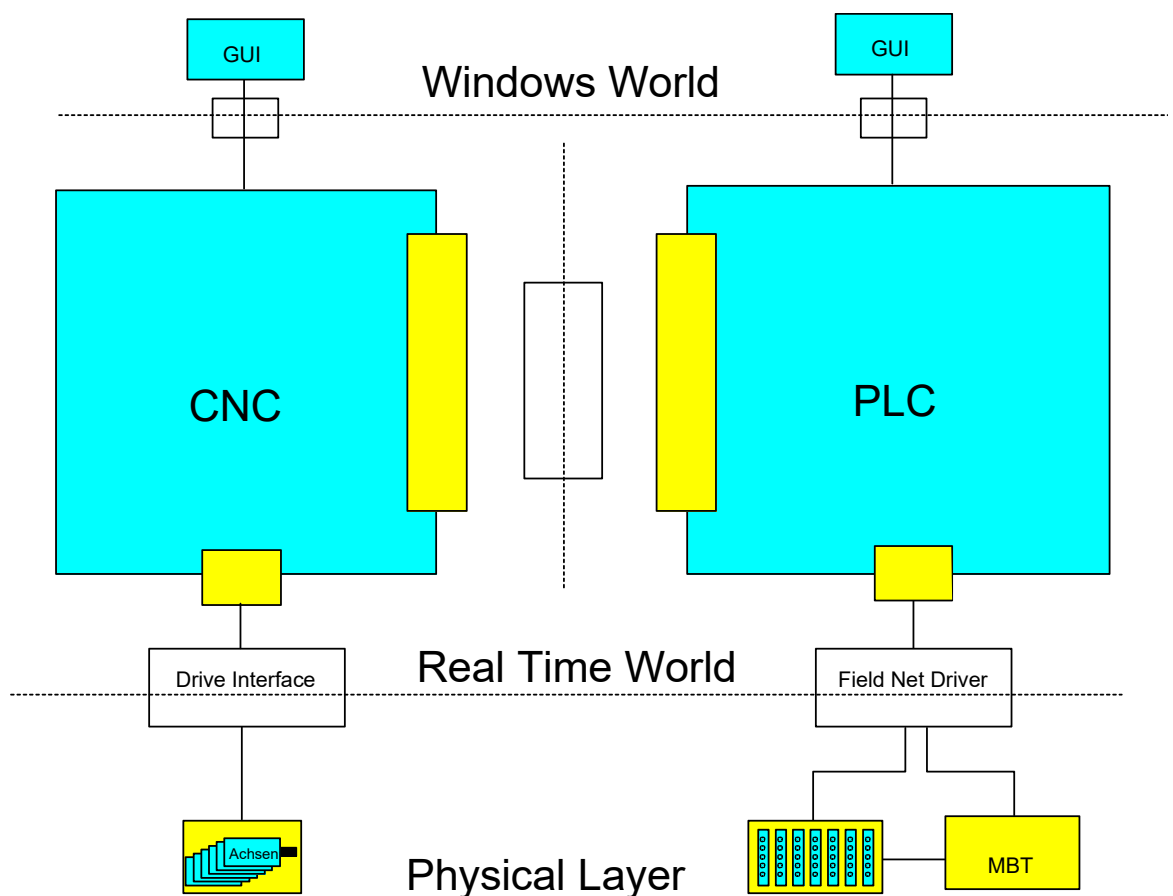
# 1

## Introduction

Extensive data volumes are exchanged between the CNC and the PLC. For example, this comprises

- Commands from the NC program of the CNC to the PLC (e.g. technology commands, such as M, S, T and H commands etc.).
- Acknowledgements of technology commands by the PLC.
- Display data of the CNC (e.g. current axis positions, current and programmed path velocity etc.).
- Jobs from the PLC to the CNC (e.g. mode switchover, set feedhold etc.).
- Jobs from the GUI to the CNC which can be verified and, if applicable, denied by the PLC (e.g. mode switchover, set feedhold etc.).

The interface between CNC and PLC is outlined in the overview diagram below.



**Fig. 1: CNC-PLC interface**

This documentation describes the structure of this interface between CNC and PLC referred to below as High-Level Interface (HLI for short).

The term “High-Level Interface”, in this context, refers to the structure of the interface with complex data structures and handshake variables. This has to be seen in contrast to a data interface, which is considered to be at a low level of abstraction (Low-Level Interface).



### Attention

#### **Safety restriction for the use of the High Level Interface:**

The values transferred between the PLC and TwinCAT CNC can be read and changed by all local users

In order to protect the system against unauthorised access, use the “IPC Security Guide” as a guideline. This will support you in the management of security risks when using Beckhoff products.

Follow the links to the „IPC Security Guide“ for further measures:

**German:** [https://infosys.beckhoff.com/content/1031/ipc\\_security/index.html](https://infosys.beckhoff.com/content/1031/ipc_security/index.html)

**English:** [https://infosys.beckhoff.com/content/1033/ipc\\_security/index.html](https://infosys.beckhoff.com/content/1033/ipc_security/index.html)

## 1.1

### **Implementation by High-Level Interface (HLI)**

A memory area is created as shared memory which both the CNC and the PLC can access in order to implement data exchange between CNC and PLC. This memory area is referred to as High-Level Interface (HLI).

In order to allow this data access, the relevant views from CNC and PLC onto the memory area must be identical. Data consistency is ensured by means of suitable data exchange mechanisms.

A library containing the structure of the HLI in structured text is provided for simple programming at the PLC end in IEC1131-3. This library is valid in each case for the current HLI format and the current status of the CNC and may change if a version update is implemented. Accordingly, the PLC must be recompiled and loaded if there is new library or if a new CNC is used.

## 1.2

### Organisation of the HLI

As opposed to the usual simple structure of the data to be exchanged in input/output data with PLC applications, the HLI contains complex structures. They reflect the logical structuring of the CNC into channels, axes and platform data.

The logical structure of the HLI is shown in the diagram below:

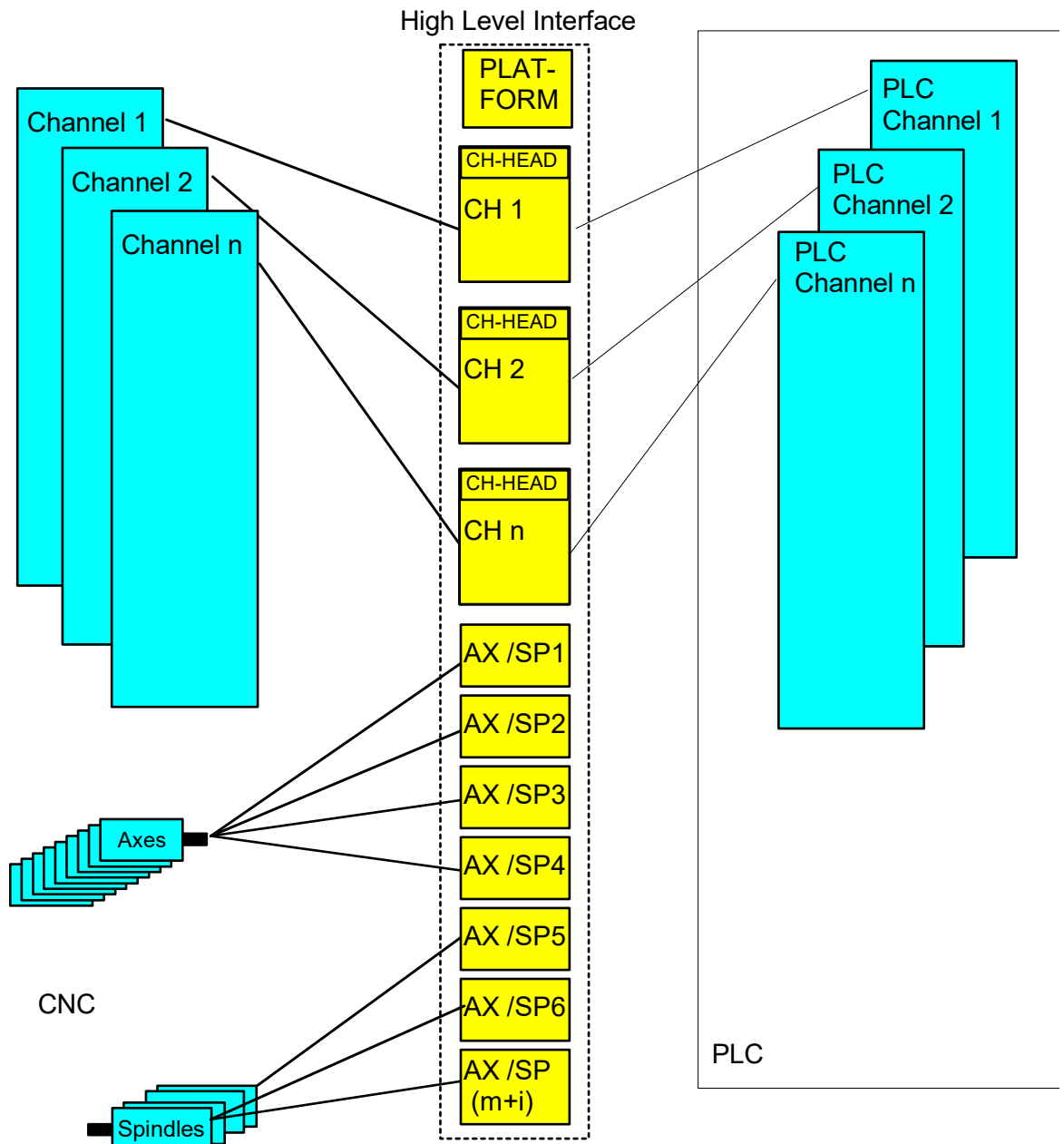


Fig. 2: Structuring of the HLI

The structuring into channel-specific and axis-specific data areas is clearly shown. The channel-specific and axis-specific memory areas on the HLI feature the same logical structure.

The relevant memory areas are subdivided into

- a header area with management data
- and a user data area with status information, control commands and technology data.

The channel/axis-specific memory areas are shown magnified in the diagram below:

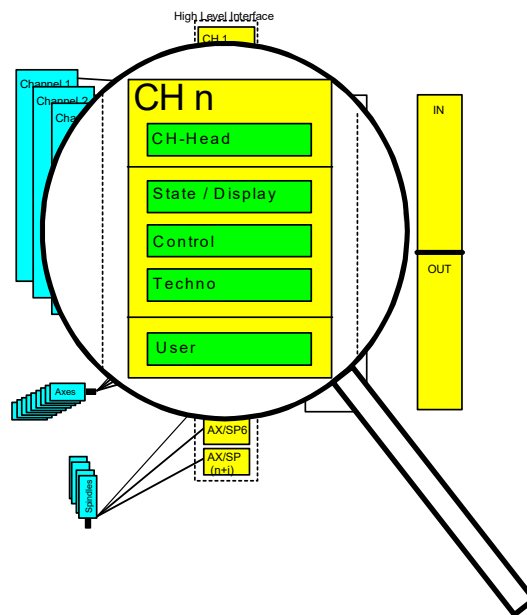


Fig. 3: Channel-specific memory area

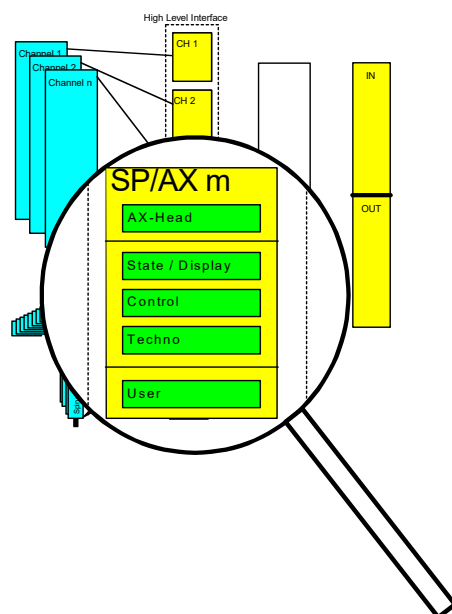


Fig. 4: Axis-specific/spindle-specific memory area

## 1.2.1 Data of a channelaxis

A distinction is made between the following data areas, whereby channels and axes may be considered the same:

The header areas contain:

- management data such as version information and log-on information

The user data areas contain:

- status and display information (CNC → PLC),
- control interfaces (PLC → CNC),
- technology areas (CNC → PLC and PLC → CNC),
- and any application-specific data.

Specific data, such as status information, is refreshed cyclically by the CNC and can be read by the PLC if needed. However, M functions, so-called usage information, must be read by the PLC. For this purpose, the interface contains suitable mechanisms to ensure that no data is lost and that the order of the data is retained.

## 1.3 Status and display information

The status information is transferred unidirectionally from the CNC to the PLC. This data is constantly refreshed by the CNC and can be read by the PLC if needed. The status information is updated without informing the PLC. Updating is not protected by a handshake protocol or semaphore mechanism.

With long PLC cycle times, the PLC does not record each short-term change under certain circumstances but is only informed of the current status.

Status information is exchanged on the HLI via memory. The transfer direction is defined for each status information item. The status data is transferred on the HLI as far as possible as single elementary data items (byte, Boolean, integer etc.).

The status information is combined depending on the CNC-internal structure to enhance structuring.

The diagram below shows the internal structure of the CNC in simplified form:

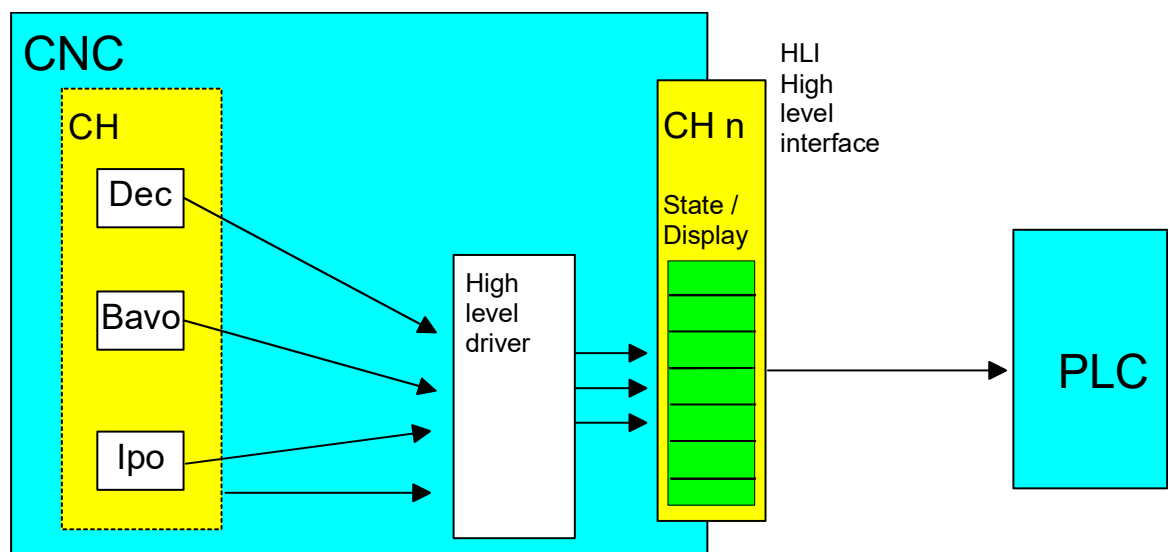


Fig. 5: Transfer of status information

### The CNC is subdivided into the modules:

- **Dec** (Decoder): Module for decoding the NC program. This module converts the CNC program into an internal control data format and executes a number of different calculations, such as parameter calculation etc. Timewise, the decoder runs ahead of actual processing and under certain circumstances may be several hundred NC blocks ahead of current processing.
- **Bavo** (path preparation, look-ahead): Module to calculate dynamic limit values based on the set parameters. It detects corners and triggers geometry changes. The Bavo also runs asynchronously with current processing and under certain circumstances may be several hundred NC blocks ahead of current processing.
- **Ipo** (interpolator): Module to generate individual positions for the current cycle, allowing for programmed and maximum velocities etc. . The interpolator represents the current state of the machine.

## 1.4 Control commands

Control commands are transferred both from the CNC to the PLC and also in the opposite direction.

If a user interface (GUI) communicates via CNC communication objects, there is an option to redirect via the PLC each command which can be operated both by the GUI and the PLC. The PLC then makes the decision regarding the extent to which the GUI command may be transferred to the CNC.

A so-called control unit is created on the HLI for each control command. The control units are differentiated and named based on the target action.

- Control units to influence the CNC are referred to as MC control units (LC acts on MC).
- Control units to influence the PLC are referred to as LC control units (MC acts on LC).

The figure below clearly shows the implemented interaction between users of a control system by means of an MC control unit.

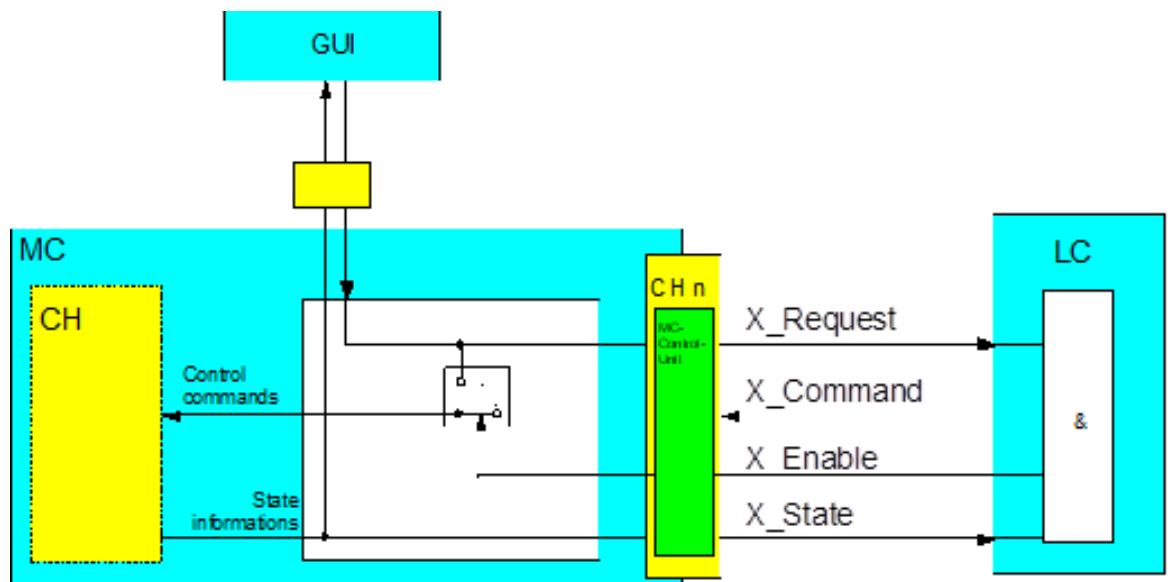


Fig. 6: Application of an MC control unit

If commands from the GUI are redirected via the PLC, the PLC must assign the element **X\_Enable** the value TRUE. The corresponding GUI commands are then set to the element **X\_Request** in the MC control unit. This means that the PLC has the option of allowing or rejecting this request from the GUI. When a GUI command is permitted by the PLC, the PLC must copy the command from the element **X\_Request** to the element **X\_Command**.

Similarly, the PLC can directly send control commands to the CNC even without a prior job from a GUI by writing element **X\_Command**.

The element **X\_State** is used to check the success of a command. The CNC saves the status corresponding to the command in this element.

An **MC control unit** has the following structure:

```
TYPE MCControlUnit:
STRUCT
    X_Request      : < DATATYPE A >; (* data commanded by GUI *)
    X_Enable       : HLI_BOOLEAN;    (* PLC operates this command *)
    X_Command      : < DATATYPE A >; (* data commanded by the PLC *)
    X_State        : < DATATYPE B >; (* CNC feedback *)
END_STRUCT
END_TYPE
```

**Note:**

- < DATA TYPE A > and < DATA TYPE B > may be identical.
- < DATA TYPE A > and < DATA TYPE B > may also be complex data structures besides the standard data types (e.g. BOOL, INT, UINT...).

### 1.4.1

#### Control unit with usage check

If the requested or commanded data is not applied statically and if an individual change should be transferred at this point, the data of the corresponding control unit is provided with an additional usage check.

```
TYPE MCControlUnit:
STRUCT
    X_Request      : < DATATYPE A >; (* data commanded by GUI *)
    X_Enable       : HLI_BOOLEAN;    (* PLC operates this command *)
    X_Command      : < DATATYPE A >; (* data commanded by the PLC *)
    X_RequestSemaphore : HLI_BOOLEAN; (* Request valid *)
    X_CommandSemaphore : HLI_BOOLEAN; (* Command valid *)
    X_State        : < DATATYPE B >; (* CNC feedback *)
END_STRUCT
END_TYPE
```

**Example:**

The CNC accepts the commanded data if the command semaphore has the value TRUE and sets this element to the value FALSE after complete acceptance of the data.

The PLC can write command data if the command semaphore has the value FALSE. The PLC sets this element to TRUE when all commanded data is written.

The CNC writes the data requested by the GUI if the request semaphore is FALSE and then sets this element to TRUE.

The PLC reads the data requested by the GUI if the request semaphore is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.

## 1.5 Operation with and without PLC

Machine start-up can also be performed with partially implemented PLC functionality. The CNC-side driver of the HLI therefore contains mechanisms for the internal simulation of interactions between PLC and CNC.

Example:

The axis should be run in with a rudimentary PLC. The PLC sets only the Axis Enables; other functions are not yet implemented in the PLC.

In order to switch operation modes or perform a reset anyway, the HLI driver simulates the required acknowledgements of the PLC so that the required action can still be performed.

In order to achieve this behaviour and still ensure reliable operation, the PLC issues a corresponding information item to the CNC for each function covered by the PLC, a so-called “present” marker. “Present” means that the PLC claims control of the interface and sends the required acknowledgements.

The PLC must then inform the CNC that it exists, once globally in the HLI. In addition all control commands are enabled individually.

This means that three scenarios are conceivable:

- CNC without PLC
- CNC with PLC which does not use all elements of the HLI
- CNC with PLC which uses all elements of the HLI

The housekeeping sections of each channel or each axis has this “present” element by which the PLC assumes responsibility for this interface.

In addition, each housekeeping unit of a control command, i.e. the control unit, contains an element which the PLC uses to inform the CNC that it supports this command. The **X\_Enable** element means at the same time that, if the corresponding command is issued via the GUI, it is routed via the PLC and reaches the CNC from there. The CNC can then no longer distinguish whether a command arrives from the PLC or the GUI.

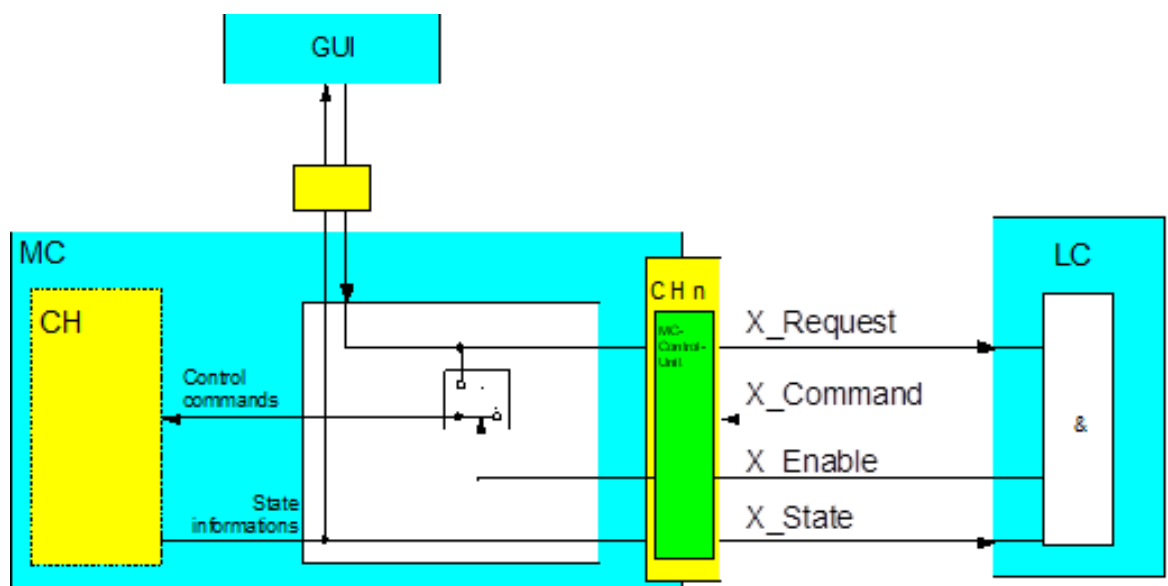


Fig. 7: Interaction, MC control unit and PLC



The connection of the GUI via the CNC is not absolutely necessary. Similarly, an interface can communicate directly with the PLC and the additionally provided communication channel GUI -> CNC -> PLC is not used in such cases.

## 2 Axes

### 2.1 Definition of axes

A distinction can be made between various types of axes in a control system. One practical classification is between programmable, logical and physical axes.

A **physical axis** corresponds to a real axis on the machine (translatory or rotary axis). In a control unit a physical axis is represented by software by means of a **logical axis** (1:1 mapping).

A logical axis is the unified representation of an axis in the axis coordinate system (acs). A logical axis in the control unit provides all the relevant information required for the related physical axis. From the point of view of the CNC, each logical axis has its own parameter set and its own interface to the PLC.

A **programmable axis** is an axis which can be programmed or commanded by the user in the part program or MDI mode. A programmable axis either corresponds directly to a logical/physical axis (1:1 mapping) or is mapped by kinematic transformation onto logical/physical axes (1:n mapping).

In multi-channel operation, a logical axis may be swapped between channels ("axis swap"). In this case, the same logical axis may occur in different attributes on the NC channels from the point of view of the programmer, e.g. as a programmable spindle on one channel (speed preset, speed-controlled) and as a programmable C axis on another channel (position preset, position-controlled).

Programmable, logical and physical axes are generally identical (1: 1 mapping) for simple machines (e.g. a 3-axis milling machine with Cartesian linear axes X, Y and Z). However, with complex machine kinematics or industrial robots, kinematic transformation of programmed to logical/physical axes is required (1:n mapping).

Axes are programmed in the CNC program and moved by the CNC. Alternatively, specific motions may also be triggered directly by the PLC.

The PLC interface for axes has a similar structure as the interface for channels. M functions configured as axis-specific M functions also appear on the axis-specific interface and must be acknowledged by this interface. Similarly, this interface contains axis enables and status information for this axis.

### 2.2 Definition of coordinate systems

Allowance must be made for various coordinate systems owing to the structure of a machine and its machining programs. The totality of a machine is represented by the reference coordinate system in world coordinates. In turn, the individual axes themselves define coordinate systems, the workpiece and the tool.

The two different coordinate systems mainly used are described below.

#### **Axis coordinate system acs**

Each axis has its own coordinate system. Each axis is mounted either on the machine base or on another axis. This means that the machine base or the corresponding axis forms the basis. The axis coordinate system of an axis is fixed with respect to the mounting point of this axis.

#### **Part program coordinate system pcs**

This coordinate system is used in the Geometry Description with the DIN 66025 programming language. The data in a part program constitutes program coordinates. Exceptions are G functions which refer to direct axis coordinates.

Other coordinate system designations are listed for the purpose of completeness.

#### **Machine coordinate system mcs**

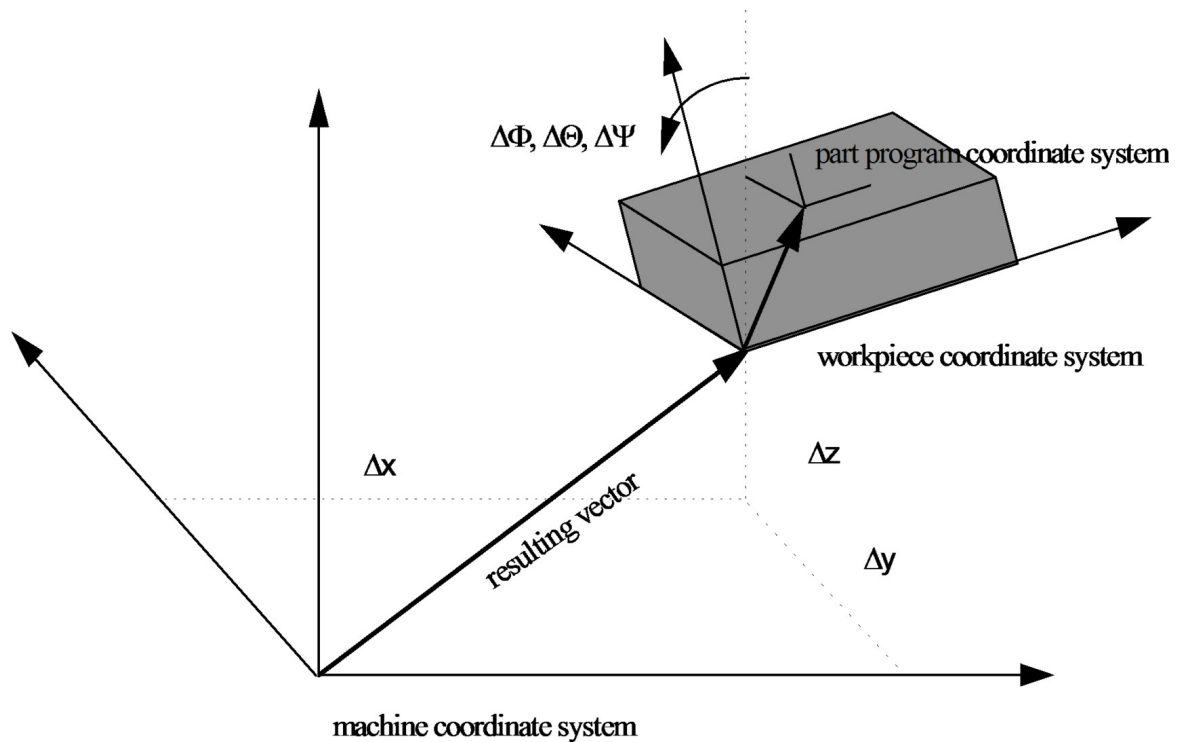
The machine coordinate system represents an abstract coordinate system. It is not bound to a fixed point of the machine. All other coordinate systems refer to this coordinate system.

### Workpiece coordinate system wcs

This coordinate system is fixed to a fixed point of the workpiece. The workpiece description by coordinate information refers to this system.

### Tool coordinate system tcs

The tool coordinate system has its origin at the clamping point of the tool. Tool information (geometry) refers to this system. Length compensation is therefore specified in tool coordinates. On Cartesian machines, the Z axis may coincide with length compensation.



**Fig. 8: Coordinate systems**

Consequently, data or variables described below always refer to a specific coordinate system. Three coordinate systems are shown in the figure above. The offsets depicted  $\Delta x$ ,  $\Delta y$  and  $\Delta z$  and the orientations  $\Delta\Phi$ ,  $\Delta\Theta$  and  $\Delta\Psi$  represent the transformation parameters from mcs to wcs. The orientation angles are Euler angles.

Naming must be supplemented by means of an additional suffix in order to allow for the time aspect. The figure below shows the machining direction of a machining block. The **end position** represents the programmed value; the **active position** is the current value of the interpolator; and the **current position** is the actual position including the control error.

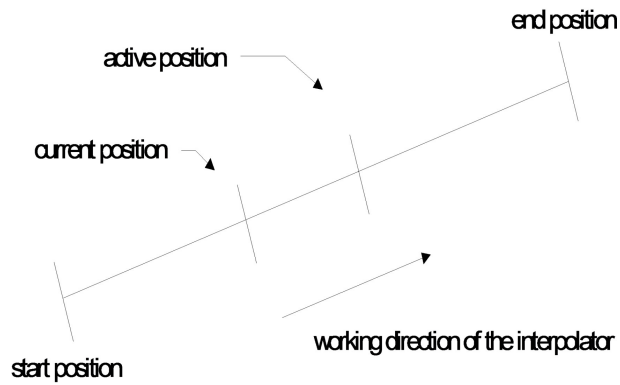


Fig. 9: Position values in PC and AC

## 2.3 Description of the axis-specific interface

### 2.3.1 Axis identification

Axis name (PCS)	
Description	Name of the logical axis used for the current reference in the current automatic program / manual block (e.g. X, Y, Z). This may be changed by default when the channel (SDA-MDS list) is programmed or dynamically in the NC program by means of a swap command.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].HLIAxeName_Data.STR_Zeichen
Data type	ARRAY[1..GCW_250_HLIAchsNameLaenge] OF BYTE
Access	PLC reads

Axis number (PCS)	
Description	If an axis is assigned to a channel, the logical number of this axis is indicated here. The logical axis number is unique in the entire system. The logical axis number is defined arbitrarily in the axis parameter lists and is required to identify the axis in case of an axis swap, among other things (e.g. #CALL AX [X, 1, 0]).
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].W_LogAchsNr pAC[axis_idx]^addr^.StateIpo_Data.W_LogAxeNr
Data type	UINT
Value range	TwinCAT PLC typically [1, gNrAx]
Access	PLC reads

Axis spindle (ACS)	
Description	Indicates whether the axis is currently interpolated by the path interpolator (e.g. thread drilling, tapping) or the BF spindle.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateIpo_Data. <b>X_SpindleAxis</b>
Data type	BOOL
Value range	[TRUE = the axis is traversed by the BF spindle, FALSE]
Access	PLC reads

Axis type (PCS)	
Description	Type of axis
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. <b>W_AxisType</b> pAC[axis_idx]^addr^.StateIpo_Data. <b>W_Type</b>
Data type	UINT
Value range	1 = Translator, 2 = Rotator, 4 = Spindel
Access	PLC reads

Channel number of the axis	
Description	Number of the channel by which the axis is currently commanded. == 0: Axis is currently not on a channel or spindle is not processing any channel command. != 0: Axis belongs to a channel or spindle is executing a channel command.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data. <b>W_NCChannel</b> pAC[axis_idx]^addr^.StateIpo_Data. <b>W_NCChannel</b>
Data type	UINT
Value range	[1, HLI_SYS_CHNMAX]
Access	PLC reads

## 2.3.2 Axis positions

### 2.3.2.1 Axis positions in the PCS

Target position (PCS)	
Description	Target position of the current NC block.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. <b>D_CmdPosition</b>
Data type	DINT
Access	PLC reads

Command position (PCS)	
Description	Position preset in the current cycle as setpoint.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. <b>D_ActPosition</b>
Data type	DINT
Access	PLC reads

Actual position (PCS)	
Description	Actual ACS position converted in the PCS.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. <b>D_CurrentPosition</b>
Data type	DINT
Access	PLC reads

Manual mode offset (PCS)	
Description	Current manual mode offset.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].D_ManOffset
Data type	DINT
Access	PLC reads
Total offset (PCS)	
Description	Sum of all active offsets TotalOffset = ACS - offsets + tool
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].D_TotalOffset
Data type	DINT
Access	PLC reads
Special features	<p>Offsets are included in the total as negative values.</p> <p>For 2.5 D machining: TotalOffset = - Offset_G92 - Offset_NP - ... + WZ_Achsversätze</p> <p>Calculation of PCS coordinates based on ACS coordinates: PCS = ACS + TotalOffset</p> <p>Example program:</p> <pre> %total_offset N00 X100 (ACS = 100mm) N10 G92 X11 (total_offset = -11mm) N20 X200 (ACS = 211mm) N30 D1 (total_offset = 64mm,) ( bei wz[1].ax_ersatz[0] 750000[0.1µm]) N40 V.G.WZ_AKT.V[0]=55 (total_offset = 44mm) N30 X300 (-&gt; ACS coordinate = 256 mm) M30 </pre>

### 2.3.2.2 Axis positions in the ACS

Target position (ACS)	
Description	Target position in the current NC block, ACS. This represents the target position of the program coordinate system referred to the axes. It is valid only as long as no transformation is active. Currently, the target position is not transformed back onto the axes.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_EndPositionACS
Data type	DINT
Access	PLC reads

Actual position (ACS)	
Description	Actual position of the current cycle in the axis coordinate system
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_CurrentPositionACS
Data type	DINT
Access	PLC reads

Command position (ACS)	
Description	Command position of current cycle in the axis coordinate system
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_ActivePositionACS
Data type	DINT
Access	PLC reads



Position offset	
Description	<p>Display of the current offset of the absolute position between the position controller axis (real physical axis) and the interpolator axis (logical axis) currently linked in the axis coordinate system.</p> <p>The display here shows an offset between the position controller position and the interpolator position caused by the use of functions such as "jog of path" (see [FCT-C15]).</p> <p>The offset displayed contains no offsets caused by zero offsets, for example.</p>
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.D_PositionOffsetPhysicalLogicalACS
Data type	DINT
Value range	0.1 µm
Access	PLC is reading

### 2.3.3 Position of tool centre point in MCS

Tool centre point position (MCS)	
Description	Command position of tool centre point in machine coordinate system MCS. The value is refreshed in each interpolation cycle.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].D_W0Position
Data type	DINT
Access	PLC reads

### 2.3.4 Status information of an axis

Axis state (PCS)			
Description	Axis state, PCS		
Signal flow	CNC → PLC		
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].W_State		
Data type	UINT		
Value range	Value	PLC constant	Meaning
	1	GCW_250_HLIAxeReady	The axis is ready and moves according to the specified command values after a command.
	3	GCW_250_HLIAxeActive	The axis is currently moved by the CNC due to an NC command or manual mode.
	5	GCW_250_HLIAxeHold	Axis cannot be moved by the CNC since feed-hold is active.
Access	PLC reads		

Axis state (ACS)			
Description	Axis state, ACS		
Signal flow	CNC → PLC		
ST path	pAC[axis_idx]^addr^.StateLR_Data.W_AxisState		
Data type	UINT		
Value range	Value	PLC constant	Meaning
	1	GCW_250_HLIAxeReady	The axis is ready and moves according to the specified command values after a command.
	3	GCW_250_HLIAxeActive	The axis is currently moved by the CNC due to an NC command or manual mode.
	5	GCW_250_HLIAxeHold	The CNC cannot move the axis because an external signal is set, such as feedhold or tracking mode, or the required drive enables are missing.
	7	GCW_250_HLIAxeError	After an error (in the drive or CNC, e.g. a software limit switch violation) the axis is in error state. Commanding a new motion is only possible after a CNC reset.
Access	PLC reads		
Special features	<b>Even if an axis is not moved in the PCS, a corresponding Cartesian or kinematic transformation may nevertheless execute a motion of the physical axis.</b> <b>Example: 90° rotation about Z; Y is moved if X is programmed.</b>		

### Distance to go (PCS)

Description	Distance to go in the current NC block, difference between target position and command position.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. <b>D_DistToGo</b>
Data type	DINT
Access	PLC reads

### Homing executed (PCS)

Description	The axis completed homing successfully and is now referenced.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. <b>X_HomingDone</b>
Data type	BOOL
Value range	[TRUE = axis is referenced, FALSE]
Access	PLC reads

### Homing executed (ACS)

Description	The axis completed homing successfully and is now referenced.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data. <b>X_HomingDone</b>
Data type	BOOL
Value range	[TRUE = axis is referenced, FALSE]
Access	PLC reads

### Position lag

Description	Current position lag of the axis, difference between command position and actual position.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pAC[axis_idx]^addr^.StateLR_Data. <b>D_FollowingError</b>
Data type	DINT
Access	PLC reads

**Axis in control window**

Description	The axis is located in the control window, i.e. the value of the position lag is less than the currently effective position window (P-AXIS-00236 or P-AXIS-00472).
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_InWindow
Data type	BOOL
Value range	[TRUE = axis is located in control window, FALSE]
Access	PLC reads

**Axis in position**

Description	The axis is located in position, i.e. the control window is reached (see above) and the axis is not interpolated.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_InPosition
Data type	BOOL
Value range	[TRUE = Axis in position, FALSE]
Access	PLC reads

**Axis is moved**

Description	The axis is currently co-used for the programmed path motion.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_IsMoved
Data type	BOOL
Value range	[TRUE = Axis is moved, FALSE]
Access	PLC reads

Axis traverses	
Description	The axis traverses, i.e. a command value is generated for this axis for the current interpolation.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_IsMoving
Data type	BOOL
Value range	[TRUE = Axis is traversed, FALSE]
Access	PLC reads
Special features	<p>If an axis is traversed, e.g. at a very low feed rate in the NC program, the axis is considered as moved for the complete path motion. However, it may not receive a new command position in every cycle due to quantisation. This means that the display “Axis is traversed” is not indicated in each cycle.</p> <p>Similarly, the axis is considered as traversed if override = 0 (traverse at velocity 0). By contrast, no velocity is commanded in the case of a FEEDHOLD, i.e. the axis is considered as not traversed.</p>

Axis moves forwards	
Description	The last output setpoint would lead to a motion in the position direction of motion.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_IsMovingForward
Data type	BOOL
Value range	[TRUE = Axis traverses forwards, FALSE]
Access	PLC reads
Special features	This status flag is unable to detect whether the axis is actually in motion. Then use the status information pAC[axis_idx]^^.addr^.StateLR_Data.X_IsMoving.

Control unit enable, state	
Description	Indicates whether control unit is enabled for the axis.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_ControlLoopEnabled
Data type	BOOL
Value range	[TRUE = Control unit enabled for axis, FALSE = No control unit enable. Axis cannot be moved by the CNC.]
Access	PLC reads

Block number	
Description	Current block number of active NC block
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_BlockNr
Data type	DINT
Access	PLC reads

Axis supply	
Description	The command of the physical axes is executed by the logical axis specified in the NC channel.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.W_LinkToLogicalAxis
Data type	UINT
Value range	[= 0 – not linked, > 0 – logical axis number]
Access	PLC is reading

Cyclic drive values are valid	
Description	The actual values supplied by the drive to the cyclic process data are valid. This signal is reset if the control unit is active and the cyclic transfer of process data is interrupted.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_CyclicDriveDataOk
Data type	BOOL
Value range	[TRUE = Process data is valid, FALSE]
Access	PLC reads
Special features	<p>This signal is dependent on the drive type as follows:</p> <p>SERCOS II: Signal is TRUE if the phase of the SERCOS ring is 4 and at least one of the bits 0x4000 or 0x8000 is set in the drive status word.</p> <p>SERCOS over EtherCAT: Signal is TRUE if the process datum WcState is 0 and at least one of the bits 0x4000 or 0x8000 is set in the drive status word.</p> <p>CANopen: Signal is TRUE if the process datum WcState is 0 and the drive status word contains a valid status according to the CANopen DS402 specification.</p> <p>PROFIBUS: Signal is TRUE if the life counter sign of the PROFIBUS slave is running.</p> <p>For all other drive types, the signal is set to TRUE immediately after CNC start-up.</p>

Configured axis mode	
Description	The axis mode configured in the axis parameter list (P-AXIS-00015) is indicated.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_AxisMode
Data type	UDINT
Value range	See description of axis parameter P-AXIS-00015.
Access	PLC reads

Axis number of gantry master axis	
Description	If the axis is a gantry slave axis, the logical axis number of the master axis in this element (see P-AXIS-00070) is indicated; otherwise 0.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.W_GantryMasterNr
Data type	UINT
Access	PLC reads

Axis assignment	
Description	The display shows the physical axes to which the commanded values of the logical axes are of the channel output.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateIPO_Data.X_LinkToPhysicalAxis
Data type	UINT
Value range	[= 0 – not linked, > 0 – logical axis number]
Access	PLC is reading

Axis-specific interpolator, state	
Description	This element indicates various states of the axis-specific interpolator. The states are bit-encoded and may be active at the same time.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateIPO_Data.D_InternalIpoState
Data type	UDINT
Special features	The element is a bit string. Several states are therefore displayed at the same time.
Value range	The assignment of bits to states is listed in the table below.
Access	PLC reads

PLC constant	Value	Meaning
HLI_AX_INDP_INTERPOLATION	16#00000002	This is an independent axis for which set-points are generated.
HLI_AX_INDP_M_FUNC_PENDING	16#00000004	This is an independent axis which waits for acknowledgement of at least one technology function (M, H, S).
HLI_AX_INDP_ERROR	16#00000008	An error occurred in an independent axis.
HLI_AX_INDP_TIME_INTERPOLATION	16#00000010	Dwell time is active for an independent axis.
HLI_AX_MAN_MV_BACK_TO_START	16#00000020	Axis is moved because it received the command to reduce to 0 the position offset accumulated by motion in manual mode.
HLI_AX_MAN_MV_BACK_WAIT_STOP	16#00000040	The motion to reduce the position offset to 0 and which was generated by a motion of the axis in manual mode was interrupted and the axis is still in the deceleration phase to standstill.
HLI_AX_TRANSM_TO_PLC_IMPOSSIBLE	16#00040000	This is an independent axis where the axis-specific interface is assigned to output technology functions.
HLI_AX_FEEDHOLD	16#001000000	Axis-specific feedhold is active.

Type of axis coupling			
Description	This indicates whether the the axis is a slave axis in an axis coupling; see also [PROG// #AXIS LINK].		
Signal flow	CNC → PLC		
ST path	pAC[axis_idx]^addr^.StateIPO_Data.W_AxLinkMode		
Data type	INT		
Value range	Value	PLC constant	Meaning
	-1	HLI_AXIS_LINK_NONE	No axis coupling active for this axis
	0	HLI_AXIS_LINK_NORMAL	The axis is a slave axis of an axis coupling
	1	HLI_AXIS_LINK_SPDL	The axis is a slave axis of a spindle coupling
	2	HLI_AXIS_LINK_GANTRY	The axis is a slave axis of a soft gantry combination
	3	HLI_AXIS_LINK_MIRROR	The axis is a slave axis of an axis coupling; the axis motion mirrors the master axis
Access	PLC reads		
Special features	The indicated axis coupling in only active when a master axis number is indicated at the same time.		



Axis number of master axis when axis coupling is active	
Description	When the axis is coupled to a master axis (see [PROG//#AXIS LINK]) the logical axis number of the master axis is indicated here.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateIPO_Data.W_AxLinkMasterAxNr
Data type	UINT
Value range	0 : No coupling active > 0: logical axis number of the master axis
Access	PLC reads

Axis type (PCS)		
Description	This indicates the configured axis type (P-AXIS-00018).	
Signal flow	CNC → PLC	
ST path	pAC[axis_idx]^^.addr^.StateIPO_Data.W_Type	
Data type	UINT	
Value range	Value	Meaning
	0x0001	Linear axis
	0x0002	Rotary axis
	0x0004	Spindle
Access	PLC reads	
Special features	Is currently not programmed	

### 2.3.4.1 Drive of an axis

Drive type		
Description	Type of axis drive The drive type is specified for each axis in the entry <code>kenngr.antr_typ</code> .	
Signal flow	CNC → PLC	
ST path	<code>pAC[axis_idx]^addr^.StateLR_Data.W_DriveType</code>	
Data type	UINT	
Value range	Value	Meaning
	1	Conventional drive (+-10 V), not used
	2	SERCOS drive
	3	Profidrive
	4	Drive Simulation
	5	Lightbus
	6	Terminal axis (+-10V) via bus
	7	Real time Ethernet
	8	CANopen drive
	16	Virtual drive
	32	CAN drive (option)
Access	PLC reads	

Drive ready for power connection / drive torque enabled	
Description	The axis drives is ready for power connection / drive torque is enabled.
Signal flow	CNC → PLC
ST path	<code>pAC[axis_idx]^addr^.StateLR_Data.X_ReadyPowerOn</code>
Data type	BOOL
Value range	[TRUE = Ready for power connection, FALSE]
Access	PLC reads
Special features	<p><b>SERCOS drives</b></p> <p>With SERCOS drives, this information is derived from bit 14 of the status word. To determine the drive state the datum <code>ready_for_control_loop_on_r</code> [► 35] must also be considered.</p> <p><b>CANopen drives</b></p> <p>With CANopen drives this signal is set when the drive is in the 'Operation enabled' state; in this state the drive is torque enabled.</p> <p>This drive state is indicated by the state word <code>xxxx xxxx x01x 0111</code> transferred by the drive (binary notation, x = don't care).</p>

Drive ready for operation	
Description	Drive control section and power supply are ready for operation.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_ReadyControlLoopOn
Data type	BOOL
Value range	[TRUE = Drive ready for operation, FALSE]
Access	PLC reads
Special features	<p><b>SERCOS drives</b></p> <p>With SERCOS drives, this information is derived from bit 15 of the status word. In order to evaluate the drive state, the datum ready_for_power_on_r [► 34] (page 32 [► 34]) must also be considered. The various combinations are described below.</p> <p>If</p> <pre>(gpAx[axis_idx]^lr_state.ready_for_power_on_r == FALSE) AND (gpAx[axis_idx]^lr_state.ready_for_control_loop_on_r == FALSE)</pre> <p>this means that the drive is not ready for power connection.</p> <p>If</p> <pre>(gpAx[axis_idx]^lr_state.ready_for_power_on_r == TRUE) AND (gpAx[axis_idx]^lr_state.ready_for_control_loop_on_r == FALSE)</pre> <p>this means that the drive is ready for power connection and the internal drive initialisation procedures are completed.</p> <p>If</p> <pre>(gpAx[axis_idx]^lr_state.ready_for_power_on_r == FALSE) AND (gpAx[axis_idx]^lr_state.ready_for_control_loop_on_r == TRUE)</pre> <p>this means that the drives control section and power supply are ready for operation but the drive has no torque.</p> <p>If</p> <pre>(gpAx[axis_idx]^lr_state.ready_for_power_on_r == TRUE) AND (gpAx[axis_idx]^lr_state.ready_for_control_loop_on_r == TRUE)</pre> <p>this means that the drive is ready for operation, i.e. “Feed rate enable, axis” [► 43] and “Drive on” [► 43] are set to TRUE and active. This means that the drive is torque enabled and follows the position command values in the NC kernel.</p> <p><b>CANopen drives</b></p> <p>This signal is set for CANopen drives when the drive is in valid state unequal to the ‘Not ready to switch on’. befindet.Der The ‘Not ready to switch on’ state is represented in the drive status word by the value xxxx xxxx x0xx 0000 (binary notation, x = don’t care). Drive has completed self-test and initialisation successfully.</p>

Drive error	
Description	An error occurred in the drive. The drive is therefore interlocked.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Error
Data type	BOOL
Value range	[TRUE = Error occurred, FALSE]
Access	PLC reads
Special features	<p><b>SERCOS drives</b></p> <p>With SERCOS drives, this information is derived from bit 13 of the status word.</p> <p><b>CANopen drives</b></p> <p>With CANopen drives this information is derived from bit 3 (bitmask 0x0008) of the state word.</p>

Drive state		
Description	State information delivered by the drive via the fieldbus. This element contains the following data dependent on the drive type:	
	Drive type (P-AXIS-00020)	Datum
	SERCOS	Value of SERCOS parameter S-0-135 (drive state).
	CANopen	Value of CANopen object 0x6041 (drive state)
	PROFIDRIVE	Value of drive state word 1 (ZSW1)
	Lightbus	Value of drive signal 23 from the drive (drive state 'DriveState3')
	For further information refer to the drive documentation or the specification of the drive profile used.	
Signal flow	CNC → PLC	
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.W_NativeDriveState	
Data type	UINT	
Access	PLC reads	

Read drive data	
Description	Data transferred by the drive to the NC kernel. The content is application-specific This data is also provided in parallel in the state of the control unit, see read/write drive data cyclically [► 70]
Special features	<p><b>Currently, data transfer can only be used for SERCOS drives.</b></p> <p>Therefore, enable transfer of the value in the axis parameter list to the drive, e.g.:</p> <pre># Cyclically read the 4 byte PLC value # uns32_3 auf S-0-0819 antr_digital.typ.sercos.at[1].ident_nr      0819 antr_digital.typ.sercos.at[1].ident_len    4 antr_digital.typ.sercos.at[1].nc_ref      LR_VAR3_IN</pre>
Signal flow	CNC → PLC
ST path	<p>pAC[axis_idx]^^.addr^.StateLR_Data.D_Word1</p> <p>pAC[axis_idx]^^.addr^.StateLR_Data.D_Word2</p> <p>pAC[axis_idx]^^.addr^.StateLR_Data.D_Word3</p> <p>pAC[axis_idx]^^.addr^.StateLR_Data.D_Word4</p>
Data type	UDINT
Access	PLC reads

DRIVE_STATE_MODE_0	
Description	present mode OF drive i.e. position loop control DRIVE_STATE_MODE_0 0x00000001
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode0
Data type	BOOL
Value range	
Access	PLC reads

DRIVE_STATE_MODE_1	
Description	DRIVE_STATE_MODE_1 0x00000002
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode1
Data type	BOOL
Value range	
Access	PLC reads

### DRIVE\_STATE\_MODE\_2

Description	DRIVE_STATE_MODE_2 0x00000004
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode2
Data type	BOOL
Value range	
Access	PLC reads

### DRIVE\_STATE\_MODE\_3

Description	DRIVE_STATE_MODE_3 0x00000004
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode3
Data type	BOOL
Value range	
Access	PLC reads
Special features	<b>Is currently not programmed</b>

### DRIVE\_STATE\_MODE\_4

Description	DRIVE_STATE_MODE_4 0x00000005
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode4
Data type	BOOL
Value range	
Access	PLC reads
Special features	<b>Is currently not programmed</b>

### DRIVE\_STATE\_MODE\_5

Description	DRIVE_STATE_MODE_5 0x00000016
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode5
Data type	BOOL
Value range	
Access	PLC reads
Special features	<b>Is currently not programmed</b>

DRIVE_STATE_MODE_6	
Description	DRIVE_STATE_MODE_6 0x00000007
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode6
Data type	BOOL
Value range	
Access	PLC reads
Special features	<b>Is currently not programmed</b>

### 2.3.4.2 Compensation of an axis

Compensation of drive drifts	
Description	State of activation of drive drift compensation
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_DriftErrorComp
Data type	BOOL
Value range	[TRUE, FALSE]
Access	PLC reads
Special features	<b>Is currently not programmed</b>

Compensation of spindle leadscrew error	
Description	State of activation of spindle leadscrew error compensation
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_PitchErrorComp
Data type	BOOL
Value range	[TRUE = Spindle leadscrew error compensation active, FALSE]
Access	PLC reads
Special features	<b>Is currently not programmed</b>

Compensation of temperature influence	
Description	Activation of drive temperature compensation
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_TempErrorComp
Data type	BOOL
Value range	[TRUE, FALSE]
Access	PLC reads
Special features	<b>Is currently not programmed</b>

Compensation of backlash error	
Description	State of activation of compensation of drive backlash error
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_BacklashErrorComp
Data type	BOOL
Value range	[TRUE = Compensation of backlash error active, FALSE]
Access	PLC reads
Special features	<b>Is currently not programmed</b>

Feed forward	
Description	State of activation of feedforward control
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_FeedForward
Data type	BOOL
Value range	[TRUE = Feedforward control active, FALSE]
Access	PLC reads
Special features	<b>Is currently not programmed</b>



### 2.3.4.3 Measurement

Measuring probe state	
Description	State of measuring probe
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_MeasureEquipActive
Data type	BOOL
Value range	[TRUE = Measuring probe active, FALSE]
Access	PLC reads
Special features	<b>Is currently not programmed</b>

Measurement value valid	
Description	A valid measurement value was latched by the drive
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_MeasureValueOk
Data type	BOOL
Value range	[TRUE, FALSE]
Access	PLC reads
Special features	<b>Is currently not programmed</b>

### 2.3.4.4 Link to ADS

Box identification	
Description	Device-dependent data defined when the system is configured (e.g. system manager) can be read out here.
Data type	HLITwincatBox
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.HLITwinCATBoxData
Access	PLC reads
Elements of the data type	
ST element	<b>.AdsNetId</b>
Description	<p>This defines the AmsNetId of the TwinCAT computer to execute the function. An empty string can also be specified for the local computer.</p> <p>The PLC variable of this type is a string containing the AMS network identifier of the target device for which the ADS command is intended. The string consists of six numerical sections separated by dots. Each numerical section includes a number between 0 and 254. Valid AMS network address include: "1.1.1.2.7.1" or "200.5.7.170.1.7". If an empty string is transferred, the AMS network identifier of the local device is automatically accepted.</p>
ST element	<b>.W_AdsPort</b>
Description	ADS devices in a TwinCAT network topology are identified by an AMS network address and a port number. The port number of the ADS device is specified by the system manager during configuration.
ST element	<b>.W_AdsChannel</b>
Description	

Device identification	
Description	The DeviceId specifies the I/O device where the function is to be executed. The device IDs are specified by the TwinCAT system manager during hardware configuration.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.HLITwinCATDeviceData
ST element	<b>.D_Id</b>
Data type	UDINT
Access	PLC reads
Special features	The device ID is currently used for ring global actions (e.g. phase switching or reading).

## 2.3.5 Axis control commands

Feed rate enable, axis	
Description	Axis-specific feed rate enable The feed rate enable must be set for all axes to be moved. If this is not the case, no path motion occurs.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	<b>SERCOS drives</b> With SERCOS drives, the datum is derived from bit 13 of the control word.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. <b>MCControlBoolUnit_ReleaseFeedhold</b>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b>
Data type	BOOL
Value range	[TRUE = Drive enable, Transition TRUE → FALSE: The drive is shut down in compliance with the acceleration parameters. FALSE = Drive STOP]
Return value	
ST element	<b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Drive enable, FALSE = Drive STOP]
Redirection	
ST element	<b>.X_Enable</b>

Feedhold ON/OFF, axis	
Description	<p>Axis-specific feedhold</p> <p>The axis-specific feedhold acts on an axis if the axis is not currently not in the path compound but in manual mode or is an independent axis. Otherwise, the global feedhold of the channel acts on the axis.</p>
Data type	MCControlDistCtrlUnit, see description of Control Unit
Data type	MCControlBoolUnit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. <b>MCControlBoolUnit_Feedhold</b>
Commanded, requested and return value	
ST element	<p><b>.X_Command</b></p> <p><b>.X_Request</b></p> <p><b>.X_State</b></p>
Data type	BOOL
Value range	[TRUE = feedhold on, FALSE = feedhold off]
Redirection	
ST element	<b>.X_Enable</b>

Feed override, axis	
Description	<p>Axis-specific feed override</p> <p>The axis-specific feed override permits weighting the axis path velocity with an additional factor. The axis-specific feed override only acts on an axis if this axis is currently not in the path compound but is moved in manual mode or as an independent axis. Otherwise the global override of the channel acts on the axis.</p> <p>The axis-specific feed override also acts on single axes and spindles.</p> <p>With spindles the feed override also acts on the velocity acknowledgement of programmed M3/ M4 or MC_MoveVelocity commands.</p> <p>Velocity acknowledgement occurs when the weighted feed rate is reached and in extreme cases when the feedrate is 0.</p> <p>(see StateLR_Data.X_RevReached [► 78])</p>
Data type	MCCControlUNS16Unit, see description of control unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. <b>MCCControlUNS16Unit_Override</b>
Commanded, requested and return value	
ST element	<b>.D_Command</b> <b>.D_Request</b> <b>.D_State</b>
Unit	0,1 %
Data type	UINT
Value range	<p>[0 ... P-AXIS-00109]</p> <p>The parameter P-AXIS-00109 is an axis-specific parameter. The typical value is 1000. See [AXIS].</p>
Redirection	
ST element	<b>.X_Enable</b>

Feed override valid, axis	
Description	Axis-specific feed override valid.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. <b>MCControlBoolUnit_OverrideValid</b>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Axis-specific feed override valid, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

Drive ON	
Description	Drive ON
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	<b>SERCOS drives</b> With SERCOS drives, the datum is derived from bit 15 of the control word.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. <b>MCControlBoolUnit_DriveOn</b>
Commanded and requested values	
ST element	<b>.X_Command</b> <b>.X_Request</b>
Data type	BOOL
Value range	[TRUE = Drive ON, Transition TRUE → FALSE: The drive is shut down in the best possible manner. FALSE = Drive OFF]
Return value	
ST element	<b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Drive ON, FALSE = Drive OFF]
Redirection	
ST element	<b>.X_Enable</b>

Control unit enable, command	
Description	Control unit enable ↔ axis-specific torque connection.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	<b>SERCOS drives</b> With SERCOS drives, the datum is derived from bit 14 of the control word.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. <b>MCControlBoolUnit_TorquePermission</b>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Torque connection, FALSE = Drive is torque-free]
Redirection	
ST element	<b>.X_Enable</b>

The diagram below shows the relationship between HLI control units and the SERCOS control word or the SERCOS status word for SERCOS drives.

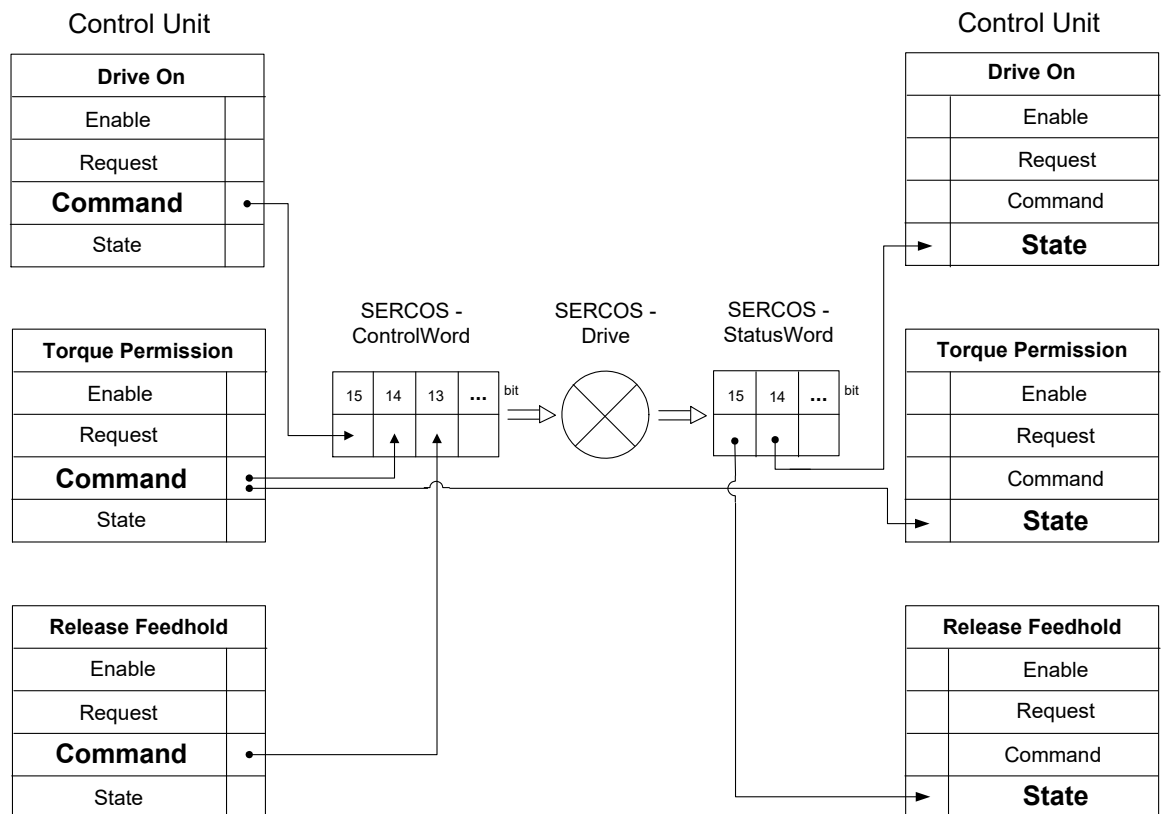


Fig. 10: Interaction of control units and SERCOS control or status words

Reference cam	
Description	Reference cam signal when homing.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	<p>When this signal is commanded, it only acts if the parameter parameter <code>lr_hw[i].cam_direct_access = 0</code> is set in the axis machine data record of the associated axis.</p> <p>When the parameter parameter <code>lr_hw[i].cam_level</code> is used in the axis machine data record, the action of this command can be set from high-active to low-active.</p> <p>The default action is described below.</p>
Access	PLC reads Request + State and writes Command + Enable
ST path	<code>pAC[axis_idx]^addr^.MCControlLr_Data.MCControlBoolUnit_ReferenceCam</code>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Reference cam switched, FALSE = Reference cam not switched]
Redirection	
ST element	<b>.X_Enable</b>

Reduced velocity, axis	
Description	When this signal is set, the axis velocity with G00 and G01 is reduced to the values defined in the axis parameters P-AXIS-00214 or P-AXIS-00155.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	If this axis is in a channel, the limits of the other axes participating in the motion are also considered. The effective value for reduced velocity is determined so that none of the axes participating in the motion overshoots their configured limit. The response is then identical to the command via a channel.
Access	PLC reads Request + State and writes Command + Enable
ST path	<code>pAC[axis_idx]^addr^.MCControlLrpo_Data.MCControlBoolUnit_ReducedFeed</code>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Reduced velocity active, FALSE = Reduced velocity not active]
Redirection	
ST element	<b>.X_Enable</b>



Reduced velocity in zone 1, axis	
Description	By setting this signal, the axis velocity is limited to the velocity defined in the axis parameter P-AXIS-00030 when the axis is located within the range defined by the parameters P-AXIS-00085 and P-AXIS-00093. If necessary the axis is decelerated after entering the zone.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	If the axis is in a channel, the limits of the other axes participating in the motion and also located within their configured zone are considered.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.MCControllpo_Data. <b>MCControlBoolUnit_ReducedFeedZone</b>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Reduced velocity in zone 1 active, FALSE = Reduced velocity in zone 1 not active]
Redirection	
ST element	<b>.X_Enable</b>

Reduced velocity in zone 2, axis	
Description	By setting this signal, the axis velocity is limited to the velocity defined in the axis parameter P-AXIS-00030 when the axis is located within the range defined by the parameters P-AXIS-00097 and P-AXIS-00105. If necessary the axis is decelerated after entering the zone.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	If the axis is in a channel, the limits of the other axes participating in the motion and also located within their configured zone are considered.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.MCControllpo_Data. <b>MCControlBoolUnit_ReducedFeedZone2</b>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Reduced velocity in zone 2 active, FALSE = Reduced velocity in zone 2 not active]
Redirection	
ST element	<b>.X_Enable</b>

### Suppressing Read-In Enable

Description	<p>Suppressing Read-In Enable</p> <p>When the read-in enable (setting NoEfg) is cleared, the interpolator does not read any new previously decoded NC motion information, i.e. the motion is stopped at the end of the current commands in the interpolator.</p>
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlIpo_Data. <b>MCControlBoolUnit_NoEfg</b>
Commanded, requested and return value	
ST element	<p><b>.X_Command</b></p> <p><b>.X_Request</b></p> <p><b>.X_State</b></p>
Data type	BOOL
Value range	[TRUE = No Read-In Enable, FALSE = Read-In Enable]
Redirection	
ST element	<b>.X_Enable</b>

### Machining simulation, axis

Description	<p>Activates and deactivates an axis-specific machining simulation.</p> <p>During machining simulation, all axis-specific technology commands of the NC program are no longer output to the PLC but are acknowledged internally.</p>
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlIpo_Data. <b>MCControlBoolUnit_MachiningSimu</b>
Commanded, requested and return value	
ST element	<p><b>.X_Command</b></p> <p><b>.X_Request</b></p> <p><b>.X_State</b></p>
Data type	BOOL
Value range	[TRUE = Machining simulation active, FALSE = Machining simulation inactive]
Redirection	
ST element	<b>.X_Enable</b>

Ignoring minimum tool velocity	
Description	<p>If a value for minimum tool velocity is configured at tool change, the NC kernel monitors that this limit is not undershot by specifying an override.</p> <p>This control unit can switch this response off and the override acts on the axis as specified.</p>
Data type	MCCControlDistCtrlUnit, see description of Control Unit
Special features	Control unit is only effective if the axis is a spindle.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. <b>MCCControlBoolUnit_IgnoreVbMinTool</b>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Undershooting minimum tool velocity permitted, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

OTC offset	
Description	When this wear offset is set, wear in the direction of this axis can be compensated. When the SURF_NORM_ORI mode is used (Wear in the direction of the surface normal), the offset value must be assigned in the third axis. Unit: 0.1µm
Data type	MCControlSGN32Unit, see description of Control Unit
Peculiarities	The wear offset is distributed by the CNC over several cycles.
Access	PLC reads Request + State and writes Command + Enable
ST Path	pAC[axis_idx]^^.addr^.McControlIpo_Data. <b>MCControlSGN32Unit_OTCOffset</b>
Commanded, requested and return values	
ST Element	<b>.D_Command</b> <b>.D_Request</b> <b>.D_State</b>
Data type	DINT
Value range	[-P-TOOL-00031, P-TOOL-00031]
Redirection	
ST Element	<b>.X_Enable</b>
Move back manual mode offset	
Description	If manual mode is active in the channel and if the commanded axis fails to move, the axis is moved by this command so that afterwards manual mode offset is 0.
Data type	MCControlBoolUnit, see description of Control Unit
Peculiarities	A rising edge (FALSE → TRUE) at X_Command starts the process. The signal is ignored if a manual mode motion is still active or manual mode offset is already 0.
Access	PLC reads Request + State and writes Command + Enable
ST Path	pAC[axis_idx]^^.addr^.McControlIpo_Data. <b>MCControlBoolUnit_ManualMvBackToStart</b>
Commanded, requested and return values	
ST Element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	rising edge (FALSE → TRUE) triggers backward motion
Redirection	
ST Element	<b>.X_Enable</b>

Stop the motion “Move back manual mode offset”	
Description	The motion that was started by the control unit command “Move back manual mode offset” is stopped by the control unit with this command.
Data type	MCControlBoolUnit, see description of Control Unit
Characteristics	A rising edge (FALSE → TRUE) initiates the command. Up to final axis standstill, the datum Axis-specific interpolator [► 31] shows that the stop process is active by the HLI_AX_MAN_MV_BACK_WAIT_STOP [► 32] bit.
Access	PLC reads Request + State and writes Command + Enable
ST Path	pAC[axis_idx]^^.addr^.McControlIpo_Data. <b>MCControlBoolUnit_ManualMvBackStop</b>
Commanded, requested and return values	
ST Element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Valuation	[TRUE = rising edge stops the motion, FALSE]
Redirection	
ST Element	<b>.X_Enable</b>

Tracking operation	
Description	The axis is set to tracking mode, i.e. the commanded setpoint is set identical to the actual value read in. Setting the setpoint and actual value equal is executed until <b>X_Command</b> = TRUE is.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	By setting the setpoint and actual value equal, the current position lag is = 0. This may result in an external force (weight of the axis) slowly changing the axis position (drift).
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. <b>MCControlBoolUnit_FollowUp</b>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = control loop open, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

Running out gantry difference	
Description	If the axis is a gantry slave axis and both the master and slave are referenced, the gantry difference is run out.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. <b>MCControlBoolUnit_GantryOn</b>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Gantry difference runout permitted, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

Accepting reference position		
Description	Accept the reference position and mark the axis as referenced on a rising edge of this control unit. Dependent on the value of the parameter P-AXIS-00278 the actual axis position is set to the following value:	
	P-AXIS-00278	Reference position of the axis
	ABSOLUT	Value of P-AXIS-00152
	OFFSET	Encoder position of the drive + P-AXIS-00279
	PLC	Value of the ReferencePosition [► 56] in the control unit.
	PLC_OFFSET	Encoder position of the drive + value of the ReferencePosition [► 56] in the control unit.
	The X_State variable indicates whether the reference position was set manually and the coordinate system was shifted as a result. The manual setting can be cleared by CNC-controlled homing (G74).  The reference position can also be set manually for an axis with absolute measuring system.	
Data type	MCControlDistCtrlUnit, see description of Control Unit	
Special features	Edge evaluation: The function is triggered on the rising edge at the command input.	
Access	PLC reads Request + State and writes Command + Enable	
ST path	pAC[axis_idx]^.addr^.McControlLr_Data.MCControlBoolUnit_SetReferencePosition	
Commanded, requested and return value		
ST element	.X_Command .X_Request .X_State	
Data type	BOOL	
Value range	[TRUE, FALSE]	
Redirection		
ST element	.X_Enable	

Reference position to be set		
Description	When the parameters P-AXIS-00278 have the value “PLC” or “PLC_OFFSET” in the axis parameter list, the value of the control unit (see also control unit SetReferencePosition [► 55]) is used to calculate the setting position when acceptance of the reference position is triggered via the HLI.  In this case, there are options on how to use the value of this control unit. See below:	
	<b>P-AXIS-00278</b>	<b>Reference position of the axis</b>
	PLC	Value in this control unit.
	PLC_OFFSET	Encoder position of the drive + value in this control unit.
Data type	MCControlSGN32Unit, see description of Control Unit	
Access	PLC reads Request + State and writes Command + Enable	
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. <b>MCControlSGN32Unit_ReferencePosition</b>	
Commanded, requested and return value		
ST element	<b>.D_Command</b> <b>.D_Request</b> <b>.D_State</b>	
Data type	DINT	
Unit	0,1 µm bzw. 10 <sup>-4</sup> °	
Value range	[MIN_SGN32, MAX_SGN32]	
Redirection		
ST element	<b>.X_Enable</b>	



Clearing referencing	
Description	<p>If an axis was homed by setting the reference position or by G74, this state can be cleared by using the actual control unit.</p> <p>If an axis has an absolute measuring system, the axis is subsequently considered as not referenced (it can be referenced again by a G74).</p> <p>This switches off the software limit switches temporarily, for example.</p> <p>Any offset activated by the control unit "Setting the reference position" is not cleared.</p> <p>The X_State variable indicates whether the axis is not currently referenced.</p>
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	As long as the control unit command (command_w) is TRUE, the axis is regarded as not referenced and can not be marked as referenced by triggering acceptance of the reference position (see also control unit SetReferencePosition [► 43]).
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlBoolUnit_ClearReferencePosition
Commanded, requested and return value	
ST element	<p><b>.X_Command</b></p> <p><b>.X_Request</b></p> <p><b>.X_State</b></p>
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

Measurement signal	
Description	This control unit transfers the measurement signal. When this control unit is used, set the entry kenngpr.probing_signal_via_plc to 1 in the parameter list of the appropriate axis.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	Edge evaluation: Use the edge parameterised in the axis parameter list in the entry kenngpr.mess_neg_flanke to accept the measurement value. See also [AXIS].
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlBoolUnit_ProbingSignal
Commanded, requested and return value	
ST element	<p><b>.X_Command</b></p> <p><b>.X_Request</b></p> <p><b>.X_State</b></p>
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

Measurement value, axis	
Description	If the axis parameter P-AXIS-00257 or the NC command #MEAS [...SIGNAL=PLC] changed the measurement signal source to the control unit ProbingSignal [► 57], the measurement value can additionally be transferred via the HLI by activating this control unit.
Data type	MCControlSGN32Unit, see description of Control Unit
Special features	If this control unit is not activated, the axis actual position is used as the measurement value on the rising edge of X_Command of the ProbingSignal [► 57] control unit. The value assigned to the X_Command element of the control unit is transferred to the decoder without modification and is then available for further calculations.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlSGN32Unit_ProbingPosition
Commanded, requested and return values	
ST element	.D_Command .D_Request .D_State
Data type	DINT
Unit	0,1 µm bzw. 10 <sup>-4</sup> °
Value range	[MIN_SGN32 ... MAX_SGN32]
Redirection	
ST element	.X_Enable

Deactivating an axis (park)	
Description	<p>Axis-specific deactivation of an axis.</p> <p>The following actions are not executed inside the CNC when an axis is deactivated:</p> <p>Error monitoring: errors signalled by the drive are not indicated by the CNC.</p> <p>HLI control bits are not transferred to the drive.</p> <p>A CNC reset does not trigger a drive reset.</p> <p>If you try to move a parked axis the error message P-ERR-70265 is issued.</p> <p>No actions are executed in the drive.</p>
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	<p><b>PROFIDRIVE drives:</b></p> <p>When this control unit is active, the bit 0x80 is set in the control word 2 (STW2).</p>
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data.MCControlBoolUnit_DeactivateAxis
Commanded, requested and return value	
ST element	<p><b>.X_Command</b></p> <p><b>.X_Request</b></p> <p><b>.X_State</b></p>
Data type	BOOL
Value range	<p>[TRUE = axis is deactivated,</p> <p>FALSE = axis is active (default mode)]</p>
Redirection	
ST element	<b>.X_Enable</b>

### 2.3.5.1 Commanding axis couplings

Commanding axis couplings	
Description	This control unit defines axis couplings for each axis. The motion of the axis can be influenced additionally or exclusively by the motion of other axes. Further details are contained in the documentation [FCT-A9].
Data type	MCControlDistCtrlUnit, see description of control unit
Access	PLC reads state and writes command + enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. MCControlAxisCouplingUnit_AxisCoupling
Commanded values	
ST element	<b>.AxisCouplingCommand</b>
Signal flow	PLC → CNC
Data type	HLIAxisCouplingCommand [▶ 60]
Access	PLC writes
Return value	
ST element	<b>.AxisCouplingState</b>
Signal flow	CNC → PLC
Data type	HLIAxisCouplingState [▶ 60]
Access	PLC reads
Redirection	
ST element	<b>.X_Enable</b>

Axis couplings, state	
Description	Indicates whether the axis couplings are active for this axis and if so, which ones.
Signal flow	CNC → PLC
Data type	HLIAxisCouplingState
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. MCControlAxisCouplingUnit_AxisCoupling. <b>AxisCouplingState</b>
Access	PLC reads
Elements of the data type	
Element	<b>.desc[ ]</b>
Data type	ARRAY [1..HLI_AxisCouplingMax] OF HLIAxisCouplingDesc [► 60]
Access	PLC reads
Element	<b>.X_State</b>
Data type	BOOL
Access	PLC reads

Axis couplings, command	
Description	This entry defines a coupling all coupling specification for the axis. The maximum number of definable coupling specifications is defined in the HLI_AxisCouplingMax constant. Further details on the definition of the coupling specifications are contained in the documentation [FCT-A9].
Signal flow	PLC → CNC
Data type	HLIAxisCouplingCommand
ST path	pAC[axis_idx]^ .addr^ .McControlLr_Data. MCControlAxisCouplingUnit_AxisCoupling. <b>AxisCouplingCommand</b>
Access	PLC writes
Elements of the data type	
Element	<b>.desc[ ]</b>
Data type	ARRAY [1..HLI_AxisCouplingMax] OF HLIAxisCouplingDesc [► 60]
Access	PLC writes
Element	<b>.X_Semaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	CNC accepts the commanded values when this element has the value TRUE. After acceptance, the CNC sets this value to FALSE.  The PLC sets this element to TRUE when the commanded values are released for acceptance by the CNC. The commanded values can only be updated by the PLC if this element has the value FALSE.

Description of an axis coupling specification			
Description	This entry defines a coupling all coupling specification for the axis. The maximum number of definable coupling specifications is defined in the HLI_AxisCouplingMax constant. Further details on the definition of the coupling specifications are contained in the documentation [FCT-A9].		
Signal flow	PLC → CNC		
ST path	pAC[axis_idx]^addr^.McControlLr_Data. MCCControlAxisCouplingUnit_AxisCoupling.AxisCouplingCommand.desc[idx]  pAC[axis_idx]^addr^.McControlLr_Data. MCCControlAxisCouplingUnit_AxisCoupling.AxisCouplingState.desc[idx]		
Data type	HLIAxisCouplingDesc		
Access	Command for axis coupling: PLC writes Status of axis coupling: PLC reads		
Elements of the data type			
ST element	.CouplingMode		
Data type	UINT		
Value range			
Value range	Constant	Value	Description
	HLI_AXIS_COUP- LING_INACTIVE	0	Coupling is not active.
	HLI_AXIS_COUP- LING_ZERO	1	Coupling factor is zero, used to deactivate an axis.
	HLI_AXIS_COUP- LING_DIRECT	2	Coupling factor is 1.
	HLI_AXIS_COUP- LING_MIRROR	3	Coupling factor is -1.
	HLI_AXIS_COUP- LING_FRACT	4	Coupling factor is a fraction defined by .desc[idx].FractNumerator / .desc[idx].FractDenominator.
ST element	. AxisNumber		
Data type	UINT		
Value range	[UINT_MIN, UINT_MAX]		
Description	Logical axis number of the source axis (the influencing axis). If the axis is to be moved by the NC program when the coupling is active, a coupling specification must be defined with the logical axis number of the axis and the coupling mode HLI_AXIS_COUPLING_DIRECT.		
ST element	.FractNumerator		
Data type	INT		

Value range	The permissible value range is [-32768, 32767]. A value of 0 in this element has the same effect as the coupling mode HLI_AXIS_COUPLING_INACTIVE.
Description	Numerator of coupling factor if coupling mode is specified as HLI_AXIS_COUPLING_FRACT. This element is not evaluated for all other coupling modes. The maximum permissible value for the coupling factor .desc[idx].FractNumerator /.desc[idx].FractDenominator is defined by the constant HLI_AXIS_COUPLING_FACT_MAX. If this value is exceeded, the error message P-ERR-70397 is issued.
ST element	<b>.FractDenominator</b>
Data type	INT
Value range	The permissible range is [-32768, 32767] excluding the 0. A value of 0 in this element results in output of the error message P-ERR-70396.
Description	Denominator of the coupling factor if the coupling mode is specified as HLI_AXIS_COUPLING_FRACT. This element is not evaluated for all other coupling modes. The maximum permissible value for the coupling factor .desc[idx].FractNumerator /.desc[idx].FractDenominator is defined by the constant HLI_AXIS_COUPLING_FACT_MAX. If this value is exceeded, the error message P-ERR-70397 is issued.

### 2.3.5.2 Distance control

Commanding distance control	
Description	This control unit can influence the distance control of the axis. This is dependent on whether it is selected in the axis parameters (see P-AXIS-00328).
Data type	MC_CONTROL_DISTANCE_CONTROL, see description Control Unit with usage check
Access	PLC reads state and writes command + X_enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data. <b>MCControl_DistanceControl</b>
Flow control of commanded values	
ST element	<b>.X_CommandSemaphore</b>
Signal flow	PLC → CNC
Data type	BOOL
Special features	<b>Consumption data item</b>



Access	<p>CNC accepts the commanded values when this element has the value TRUE. After acceptance, the CNC sets this value to FALSE.</p> <p>The PLC sets this element to TRUE when the commanded values are released for acceptance by the CNC. The commanded values can only be updated by the PLC if this element has the value FALSE.</p>
Commanded values	
ST element	<b>.Command</b>
Signal flow	PLC → CNC
Data type	HLIDistanceControlCommand [► 64]
Access	PLC writes
Distance control state	
ST element	<b>.State</b>
Signal flow	CNC → PLC
Data type	HLIDistanceControlState [► 64]
Access	PLC reads
Redirection	
ST element	<b>.X_Enable</b>

Command for distance control		
Description	This entry commands distance control.	
Signal flow	PLC → CNC	
Data type	HLIDistanceControlCommand	
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data.MCControl_DistanceControl. <b>Command</b>	
Access	PLC writes	
Elements of the data type		
ST element	<b>.D_Transition</b>	
Data type	UDINT	
Value range	<b>Value</b>	<b>Constant</b>
	0	HLI_DIST_CTRL_OFF
	1	HLI_DIST_CTRL_ON
	2	HLI_DIST_CTRL_FREEZE
	3	HLI_DIST_CTRL_REF
Description	See table: Transitions to command distance control [► 68]	
ST element	<b>.D_Position</b>	
Data type	DINT	
Value range	[DINT_MIN, DINT_MAX]	
Description	The meaning depends on the commanded transition: HLI_DIST_CTRL_ON: Sollposition der Werkstückoberfläche (SET_POS) HLI_DIST_CTRL_REF: Reference position of the workpiece surface (REF_POS)	

Distance control status		
Description	This entry read the distance control state.	
Signal flow	PLC → CNC	
Data type	HLIDistanceControlState	
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data.MCControl_DistanceControl. <b>State</b>	
Access	PLC reads	
Elements of the data type		
Element	<b>.D_State</b>	
Data type	UDINT	
Access	PLC reads	
Value range	<b>Value</b>	<b>Constant</b>
	0	HLI_DIST_CTRL_STATE_INACTIVE
	1	HLI_DIST_CTRL_STATE_ACTIVE
	2	HLI_DIST_CTRL_STATE_FREEZE
	3	HLI_DIST_CTRL_STATE_TURNING_OFF
	4	HLI_DIST_CTRL_STATE_ERROR
Description	See table: Distance control state [► 68]	
Element	<b>.D_ActualPosition</b>	
Data type	DINT	
Access	PLC reads	
Unit	0,1 µm bzw. 0,0001°	
Description	This datum indicates the current actual position of the workpiece surface detected by the sensing controller.	
Special features	This data item is only entered if distance control is activated in the axis parameters (see P-AXIS-00328).	
Element	<b>.D_ActualOffset</b>	
Data type	DINT	
Access	PLC reads	
Unit	0,1 µm bzw. 0,0001°	
Description	This data item indicates the current position offset of distance control by which the axis was moved due to deviations between the actual workpiece surface and the specified position (SET_POS). The following applies in stationary state (constant workpiece surface and position offset completely run out): Position offset = SET_POS – D_ActualPosition	

Special features	This data item is only entered if distance control is activated in the axis parameters (see P-AXIS-00328).
------------------	--

The tables below list the permissible values to command distance control and the defined states resulting.

Permissible transitions to command the distance control

Transition	Value	Meaning
HLI_DIST_CTRL_OFF	0	Distance control is deactivated. The program switches to the TURNING OFF state in which the position offset is run out. The state then switches automatically to the INACTIVE state.
HLI_DIST_CTRL_ON	1	Distance control is activated. When activated a command position for the workpiece surface must be transferred in the "position" datum. If no absolute encoder is used, distance control must first be referenced.
HLI_DIST_CTRL_FREEZE	2	The current position offset is frozen. Axis adjustment to the actual workpiece surface is ended.
HLI_DIST_CTRL_REF	3	Referencing distance control if no absolute encoder is used. Referencing is only permitted in the INACTIVE state. With this transition a reference position must additionally transferred in the "position" datum.
HLI_DIST_CTRL_ON_CONS T_DIST	4	Activating distance control with continuous distance specification. On activation, a set position must be specified. If the distance sensor supplies no absolute values, distance control must be referenced in advance.
HLI_DIST_CTRL_DRYRUN	5	Activate distance control for pure evaluation of data. No axis tracking in case of changes to the workpiece surface. When activated a command position for the workpiece surface must be transferred in the "position" datum. If no absolute encoder is used, distance control must first be referenced.
HLI_DIST_CTRL_CONST_DI ST	6	Activate distance control for pure evaluation of data. No axis tracking in case of changes to the workpiece surface. On activation, a set position must be specified. If the distance sensor supplies no absolute values, distance control must be referenced in advance.

Defined states of distance control

State	Value	Meaning
HLI_DIST_CTRL_STATE_INACTIVE	0	Distance control is deactivated. The specified offset ("actual_offset") is 0.
HLI_DIST_CTRL_STATE_ACTIVE	1	Distance control is active and adjusts the axis to the workpiece surface.
HLI_DIST_CTRL_STATE_FREEZE	2	Distance control is active. The offset ("actual_offset") is frozen, i.e. the axis is not adjusted to the workpiece surface.
HLI_DIST_CTRL_STATE_TURNING_OFF	3	Distance control was deactivated. The actual active offset ("actual_offset") is run out. As soon as the offset is 0, an automatic switch to INACTIVE state takes place.
HLI_DIST_CTRL_STATE_ACTIVE_CONST_DIST	4	Distance control is active and adjusts the axis to the workpiece surface. Continuous specification of the set distance of the tool to the workpiece surface.
HLI_DIST_CTRL_STATE_ERROR	5	Distance control is in error state, e.g. due to an incorrect state transition or due to an error in the position controller. Only a transition to TURNING OFF is possible to exit this state.
HLI_DIST_CTRL_STATE_DRYRUN_CONST_DIST	6	Distance control is active, however axis is not adjusted to workpiece surface. This allows the evaluation of data, for example filter effect without feedback from the control system. Continuous commanding of the set distance of the tool to the workpiece surface.
HLI_DIST_CTRL_STATE_DRYRUN_SETPOS	7	Distance control is active, however axis is not adjusted to workpiece surface. This allows the evaluation of data, for example filter effect without feedback from the control system.

## 2.3.6 Control commands of a drive

Reading/writing drive data cyclically	
Description	<p>4 elements are provided for writing by the PLC in each interpolation cycle. They can be transferred to the drives via the subordinate drive protocol.</p> <p>The content and action are application-specific (dependent on the drives)</p>
Data type	MCCControlUNS32Unit
Special features	<p><b>Currently, data transfer can only be used for SERCOS drives.</b></p> <p>Therefore, enable transfer of the value in the axis parameter list to the drive, e.g.:</p> <pre># write the 4 byte PLC value cyclically # uns32_1 auf S-0-0815 antr_digital.typ.sercos.mdt[1].ident_nr      0815 antr_digital.typ.sercos.mdt[1].ident_len    4 antr_digital.typ.sercos.mdt[1].nc_ref      LR_VAR1_OUT  # Cyclically read the 4 byte PLC value # uns32_3 auf S-0-0819 antr_digital.typ.sercos.at[1].ident_nr      0819 antr_digital.typ.sercos.at[1].ident_len    4 antr_digital.typ.sercos.at[1].nc_ref      LR_VAR3_IN</pre>
Access	PLC reads Request + State and writes Command + Enable
ST path	<p>pAC[axis_idx]^addr^.McControlLr_Data.MCControlUNS32Unit_D&lt;i&gt;</p> <p>where i = [1, 4]</p>
Commanded and requested value	
ST element	<p><b>.D_Command</b></p> <p><b>.D_Request</b></p>
Data type	UDINT
Return value	
ST path	<b>.D_State</b>
Data type	UDINT
Special features	<p>These values are also provided at the position controller interface. Siehe Antriebsdaten zyklisch lesen [► 34] (pAC[axis_idx]^addr^.StateLR_Data.D_Word1)</p>
Redirection	
ST path	<b>.X_Enable</b>

Operation mode	
Description	<p>Various drive operation modes can be commanded by the PLC (or operator):</p> <p><b>SERCOS drives:</b></p> <p>With SERCOS drives, this information is sent to bit 8 and bit 9 of the control word. This switches over between main and ancillary drive operation modes.</p> <p>Mode0 corresponds to the lowest operation mode control bit of the drive.</p> <p>Currently, variables Mode0 and Mode1 are used only for <b>SERCOS drives</b>.</p> <p><b>PROFIDRIVE drives:</b></p> <p><b>MCControlBoolUnit_Mode0</b></p> <p>This control unit activates the drive function 'parking axis' by setting bit 7 in control word 2.</p> <p>The control unit state element indicates whether the function is active in the drive (value of drive status word 2 bit 7).</p> <p>Tracking mode is activated by the CNC internally for a parked axis.</p> <p><b>MCControlBoolUnit_Mode1:</b></p> <p>This control unit activates the drive function 'parking encoder' by setting bit 14 of the encoder control word.</p> <p>The control unit state element indicates whether the function is active in the drive (value of encoder status word but 14).</p> <p>Tracking mode is activated internally in the CNC for a parked encoder.</p> <p><b>MCControlBoolUnit_Mode2:</b></p> <p>Currently not in use.</p>
Data type	MCControlBoolUnit
Special features	<b>Elements MCControlBoolUnit_Mode3 ... MCControlBoolUnit_Mode6 are currently in use.</b>
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data.MCControlBoolUnit_Mode<i> where i = 0; 6]
Commanded, requested and return values	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Bit set, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

### 2.3.7 External axis commanding

Activating external position or velocity command values, axis	
Description	<p>Specifying velocity or position command values by the SPS effective in addition to the interpolator. No monitoring takes place of transferred values for compliance with the dynamic axis limits.</p> <p>To activate this interface, set the parameter P-AXIS-00732 to 1.</p>
Data type	MC_CONTROL_ADD_CMD_VALUE_UNIT, see description of Control Unit
Special features	<p>When this interface is used, the axis positions in the interpolator and decoder are permanently offset. A repeated synchronisation of axis positions is executed, e.g. at program start, after homing, after measuring or by the NC command #CHANNEL INIT or #SET DEC LR SOLL (old syntax).</p> <p>If synchronisation should not take place (the offset remains as static offset), set the parameter P-AXIS-00322 to the value 1.</p>
Access	PLC writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlAddCmdValueUnit_AddCmdValue
Commanded values	
ST element	.HLIAddCmdValue_Command
Data type	HLIAddCmdValueData [► 72]
Activation	
ST element	.X_Enable

External command value, axis	
Description	Transferring additional position or velocity command values. On activation both values are effective at the same time.
Data type	HLIAddCmdValueData
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlAddCmdValueUnit_AddCmdValue.HLIAddCmdValue_Command
Data structure elements	
ST element	.D_AddPosValue
Special features	Absolute value
Data type	DINT
Unit	0,1 µm
Access	PLC writes
ST element	.D_AddSpeedValue
Data type	DINT
Unit	1 µm/s
Access	PLC writes



### 2.3.8 Measuring with external measuring hardware

Interface for external measuring hardware	
Description	The CNC informs the PLC about the start or end of a measurement run via the external measuring interface so that the PLC can activate or deactivate an external measuring hardware accordingly. The measurement signal source must be set to PLC_EXT_LATCH_CTRL in order to activate this interface (see P-AXIS-00516 or [PROG//Extended programming])
Data type	LcControlExtLatchControl
Special features	This interface is only used to control the activation and deactivation of an external measuring hardware. The CNC signals detection of the measurement value or the detected measuring position via the control units ProbingSignal [► 43] or ProbingPosition [► 43].
Access	PLC writes Please + Done
ST path	pAC[axis_idx]^addr^ <b>ExtLatchControl</b>
Job	
ST element	<b>.X_Please</b>
Description	By setting X_Please the CNC signals to the PLC that the external measuring hardware is to be activated or deactivated since a measurement run is starting or was aborted.
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	<b>Consumption data item</b>
Access	The CNC refreshes the data of the measuring interface only if this element is FALSE. After refreshment, the CNC sets this element to TRUE after the element X_Done was set to FALSE.  The PLC reads the data of the measuring interface if this element has the value TRUE. After the data is transferred, the PLC sets the value to FALSE.
Parameter	
ST element	<b>.ExtLatchOrder</b>
Description	The CNC signals to the PLC in this datum the required measurement parameters such as activate/deactivate the measurement function or the number of the measurement input.
Data type	HLI_EXT_LATCH_ORDER [► 73]
Access	PLC reads
Acknowledgement	
ST element	<b>.X_Done</b>
Description	By setting the element X_Done to TRUE, the PLC signals to the CNC that the measuring command was executed.
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	<b>Consumption data item</b>
Access	The PLC sets the value to TRUE if the measuring command was executed. The CNC sets the value to a FALSE before a new command.

Measuring command data	
Description	The CNC signals to the PLC in this parameter whether the measuring hardware must be activated or deactivated. In addition, the CNC informs the PLC of the measurement input to be used and the relevant measuring edge.
Data type	HLI_EXT_LATCH_ORDER
ST path	pAC[axis_idx]^addr^.ExtLatchControl.ExtLatchOrder
Access	PLC reads
Measuring command identifier	
Description	ID of measuring command to be executed.
ST element	.Order_Id
Data type	UDINT
Value range	HLI_EXT_LATCH_ENABLE_PROBE: Start of measurement run, the measuring hardware must be activated. HLI_EXT_LATCH_DISABLE_PROBE: The measurement run was ended or aborted by reset. The measuring hardware must be deactivated again.
Access	PLC reads
Number of probing input	
Description	The CNC signals to the PLC in this datum the input to be used for measuring (see P-AXIS-00517)
ST element	.D_Input
Data type	DINT
Value range	Measuring input 1 – 255
Access	PLC reads
Relevant measuring edge	
Description	The CNC signals to the PLC in this datum the relevant measuring edge (raising/falling) to be used for measuring (see P-AXIS-00518)
ST element	.Edge
Data type	UDINT
Value range	HLI_MEAS_SIGNAL_LOW_ACTIVE: The measurement value is to be detected on the falling edge. HLI_MEAS_SIGNAL_HIGH_ACTIVE The measurement value is to be detected on the rising edge.
Access	PLC reads

The block diagram below shows an example of the signal profile of the control units ExtLatchControl [► 73], ProbingSignal [► 43] and ProbingPosition [► 43] during the measurement process with external hardware:

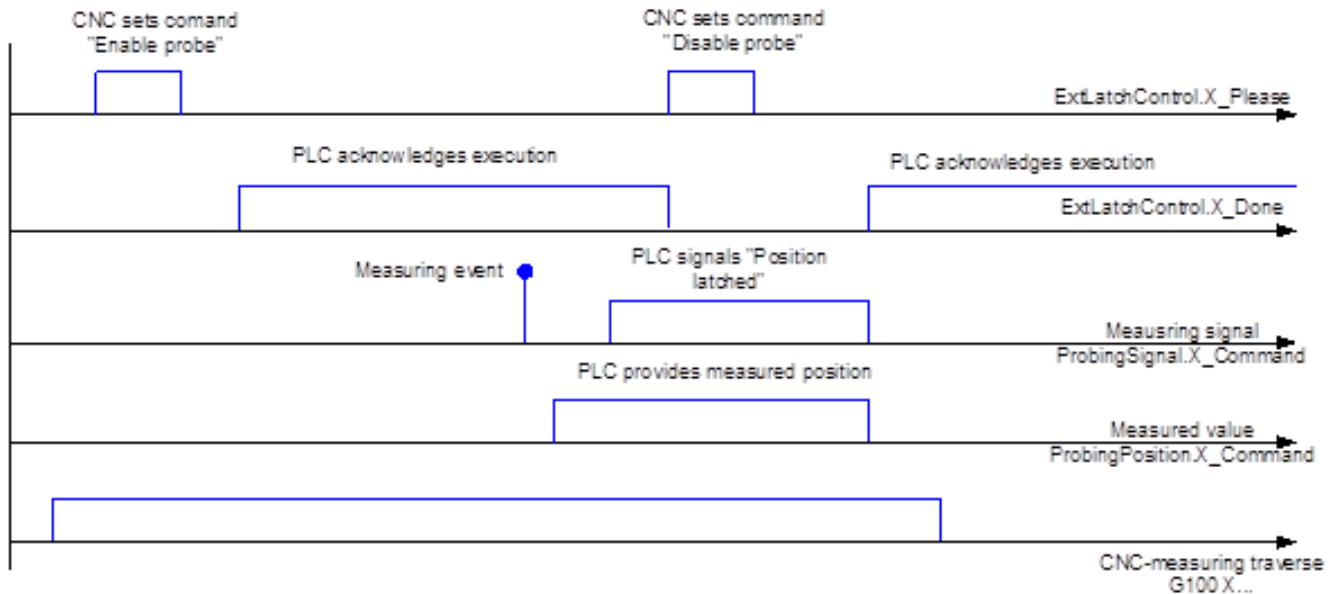


Fig. 11: Example signal profile of measurement with external hardware



#### Notice

When the external measuring interface is used, the probing signal of the control unit ProbingSignal [► 43] is not dependent on the relevant measuring edge P-AXIS-00518. A positive edge always signals the successful detection of a measurement value in the external measuring hardware.



#### Notice

If the control unit ProbingPosition [► 43] is not activated when the measuring event occurs, the measurement value of the current actual value at the time when the probing signal is used.

## 3 Spindle

### 3.1 Introduction

A spindle is an axis with extended properties. At the CNC end, a spindle is mapped by means of a separate motion controller (interpolator).

This means that the spindle axis can be moved not only by the NC program, motion jobs can also be generated at any time by the PLC.

### 3.2 Description of the spindle-specific interface

#### 3.2.1 Speeds of a spindle

Nominal speed	
Description	Current speed setpoint of the spindle
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_ActiveRev
Unit	10 <sup>-3</sup> °/s
Data type	DINT
Access	PLC reads

Actual speed	
Description	Current actual speed of the spindle
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_CurrentRev
Unit	10 <sup>-3</sup> °/s
Data type	DINT
Access	PLC reads

Speed programmed	
Description	Speed setpoint of the spindle programmed via M03, M04 or S in the NC program
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_CmdRev
Unit	10 <sup>-3</sup> °/s
Data type	DINT
Access	PLC reads
Special features	If the axis is not a spindle, the value is undefined.

### 3.2.2 Positions of a spindle

Target position	
Description	Target position when positioning with M19
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.Statelpo_Data.HLIAXeDispData_Spindle.D_CmdPosition
Unit	10 <sup>-4</sup> °
Data type	DINT
Access	PLC reads

Actual position	
Description	Current actual position when positioning with M19
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.Statelpo_Data.HLIAXeDispData_Spindle.D_ActPosition
Unit	10 <sup>-4</sup> °
Data type	DINT
Access	PLC reads

### 3.2.3 Status information of a spindle

Operating state		
Description	Current operating state of the spindle	
Signal flow	CNC → PLC	
ST path	pAC[axis_idx]^^.addr^.Statelpo_Data.HLIAXeDispData_Spindle.D_Mode	
Data type	UDINT	
Value range	<b>Value</b>	<b>Meaning</b>
	1	M05 active
	2	M03 active
	4	M04 active
	8	M19 active
	16	Superimposed motion active (PLCopen)
	32	RPF active
Access	PLC reads	

<b>Speed monitoring active</b>	
Description	Speed monitoring is activated for the spindle.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_RevControlActiv
Data type	BOOL
Value range	[TRUE = Speed monitoring is active, FALSE]
Access	PLC reads
Special features	If the axis is not a spindle, the value is undefined.

<b>Nominal speed reached</b>	
Description	The spindle has reached the programmed speed setpoint.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_RevReached
Data type	BOOL
Value range	[TRUE = Speed setpoint reached, FALSE]
Access	PLC reads
Special features	If the axis is not a spindle, the value is undefined.

<b>Spindle stopped</b>	
Description	The spindle speed is lower than the value of vb_min_null (speed = 0) contained in the axis parameter list.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_RevZero
Data type	BOOL
Value range	[TRUE = Spindle stopped, FALSE]
Access	PLC reads
Special features	If the axis is not a spindle, the value is undefined.

### Speed monitoring invalid

Description	If the speed of rotation of the spindle exceeds the limit speed of the position sensor of the spindle, speed monitoring is deactivated. The limit speed for the position measuring system is defined in the axis parameter list by parameter vb_regelgrenze.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_RevControlInvalid
Data type	BOOL
Value range	[TRUE = Speed monitoring invalid, FALSE]
Access	PLC reads
Special features	If the axis is not a spindle, the value is undefined.

### Distance to go

Description	Distance to go when positioning with M19
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateIpo_Data.HLIAXeDispData_Spindle.D_DistToGo
Unit	10 <sup>-4</sup> °
Data type	DINT
Access	PLC reads

### Spindle command error

Description	A command sent to the spindle was not executed due to error.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateIpo_Data.X_spindle_order_error_r
Data type	BOOL
Value range	[TRUE = Command not executed, FALSE]
Access	PLC reads
Special features	Valid only in conjunction with external spindle command by the PLC

### 3.2.4 Control commands of a spindle

Spindle stop at program end	
Description	If this element is set to TRUE at end of the program, the spindle is stopped.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlIpo_Data.MCControlBoolUnit_SpdIStopAtProgEnd
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Spindle is stopped at end of program, FALSE]
Redirection	
ST element	.X_Enable

Spindle reset	
Description	Commanding a reset for the spindle.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlIpo_Data.MCControlBoolUnit_SpdIReset
Commanded and requested Value	
ST element	.X_Command .X_Request
Data type	BOOL
Value range	[TRUE Spindle reset commanded, FALSE]
Return value	
ST element	.X_State
Data type	BOOL
Value range	[TRUE = Spindle reset is executed, FALSE]
Redirection	
ST element	.X_Enable



Spindle error	
Description	Sets the spindle to error state
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. <b>MCControlBoolUnit_SpdLError</b>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

Emergency stop, spindle	
Description	<b>Commanding this control unit is only effective if the axis is a spindle.</b> If this element is set to active for emergency stop (TRUE), a speed setpoint of zero is output. The axis is stopped at the emergency stop deceleration which is parameterised. This type of motion influence has maximum priority.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. <b>MCControlBoolUnit_EmergencyStop</b>
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Emergency stop active, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

## 3.2.5 External spindle command

### 3.2.5.1 Control unit of external spindle command

External spindle command	
Description	External spindle command. Commands, e.g. spindle stop or spindle positioning, can be issued to the spindle by this control unit over the HLI interface. Enter the other parameters of the command sent to the spindle in the structure HLI_EXT_TO_IPO_DATA.
Data type	MCCControlExtTolpo
Commanded values	
ST path	pAC[axis_idx]^addr^.McControllpo_Data.MCCControlExtTolpo_ExtTolpo.HLIExtTolpoData_Command
Signal flow	PLC → CNC
Data type	HLIExtTolpoData [► 82]
Access	PLC writes
Flow control commanded data	
ST path	pAC[axis_idx]^addr^.McControllpo_Data.MCCControlExtTolpo_ExtTolpo.X_Command-Semaphor
Signal flow	PLC → CNC
Data type	BOOL
Value range	[TRUE = Data is valid, FALSE = Data was accepted]
Special features	<b>Consumption data item</b>
Access	<p>CNC accepts the commanded values when this element has the value TRUE. After acceptance, the CNC sets this value to FALSE.</p> <p>The PLC sets this element to TRUE when the commanded values are released for acceptance by the CNC. The commanded values can only be updated by the PLC if this element has the value FALSE.</p>

### 3.2.5.2 User data for external spindle commands

The parameters for external spindle commands must be saved in the structure elements described below. Not all structure elements need to be completed dependent on the type of command for the spindle.

Programmed block feed	
Description	Programmed block feed
Signal flow	PLC → CNC
ST path	pAC[axis_idx]^addr^.McControllpo_Data.MCCControlExtTolpo_ExtTolpo.HLIExtTolpoData_Command.D_VbProg
Unit	10 <sup>-3</sup> °/s
Data type	DINT
Access	PLC writes

G and M functions	
Description	Bit-encoded spindle parameters
Signal flow	PLC → CNC
ST path	pAC[axis_idx]^addr^.McControlIpo_Data.MCControlExtTolpo_ExtTolpo.HLIExtTolpoData_Command. <b>D_GeoGmFkt</b>
Data type	UDINT
Value range	Only the following 2 bits are currently used: HLI_OPTIM_RICHTEN 0x00000010L Dress optimised rotary axis. When positioning the spindle, the target position is approached by the shortest possible path. HLI_ABSOLUT 0x00000100L Position information absolute
Access	PLC writes

Motion path	
Description	Motion path (relative or absolute) when positioning the spindle with M19.
Signal flow	PLC → CNC
ST path	pAC[axis_idx]^addr^.McControlIpo_Data.MCControlExtTolpo_ExtTolpo.HLIExtTolpoData_Command. <b>D_Fahrweg</b>
Unit	10 <sup>-4</sup> °
Data type	DINT
Access	PLC writes

Type of spindle commands			
Description	The type of spindle commands is defined with this element.		
Signal flow	PLC → CNC		
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data.MCControlExtTolpo_ExtTolpo.HLIExtTolpoData_Command. <b>W_SatzTyp</b>		
Data type	UINT		
Value range	<b>Value</b>	<b>Constant</b>	<b>Meaning</b>
	1	HLI_NC_MOVE_LIN	Linear interpolation
	7	HLI_NC_MOVREF	Homing
	16	HLI_NC_MOVE_ENDLOS	Endless rotation
	20	HLI_NC_GETRIEBE	Spindle gear changes
	29	HLI_NC_SPINDEL_STOP	Stopping the spindle from endless rotation
	30	HLI_NC_SUPER_IMPOSED	accordingly PLCopen MC_SuperImposed
	31	HLI_NC_TABLE_SELECT	Selection of a table according to PLCopen MC_CamTableSelect
	32	HLI_NC_CAM_IN	accordingly PLCopen MC_CamIn
	33	HLI_NC_CAM_OUT	accordingly PLCopen MC_CamOut
	34	HLI_NC_GEAR_IN	accordingly PLCopen MC_GearIn
	35	HLI_NC_GEAR_OUT	accordingly PLCopen MC_GearOut
	36	HLI_NC_PHASING	accordingly PLCopen MC_Phasing
	38	HLI_NC_TOUCH_PROBE	accordingly PLCopen MC_TouchProbe
	39	HLI_NC_ABORT_TRIGGER	accordingly PLCopen MC_AbortTrigger
Access	PLC writes		

Direction of spindle rotation			
Description	Defining the direction of spindle rotation		
Signal flow	PLC → CNC		
ST path	pAC[axis_idx]^addr^.McControlIpo_Data.MCControlExtToIpo_ExtToIpo.HLIExtToIpoData_Command. <b>W_DrehInfo</b>		
Data type	UINT		
Value range	The following values apply when the spindle moves:		
	<b>Value</b>	<b>Constant</b>	<b>Meaning</b>
	0x0000	---	Negative direction of rotation
	0x0001	HLI_SPDL_POS_DREHR	Positive direction of rotation
Access	PLC writes		

## 4 Channel

### 4.1 Introduction

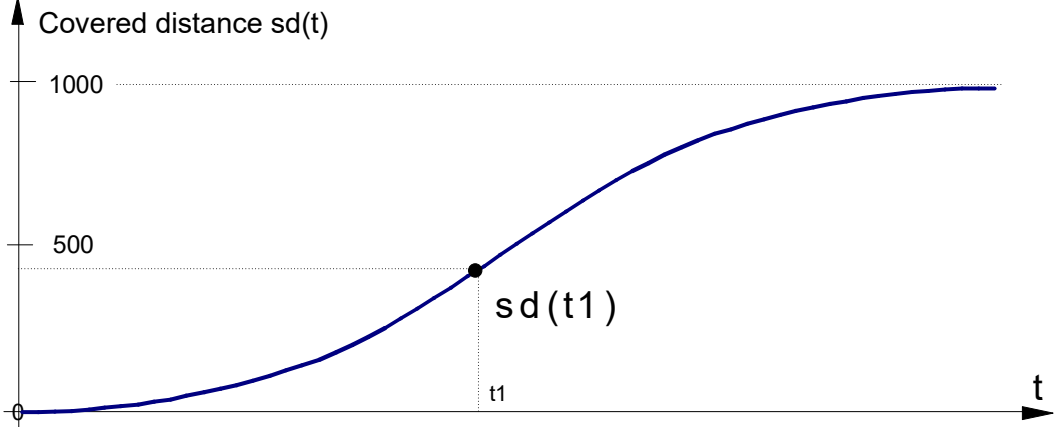
The command variables required for a machine motion are generated within a channel. In this case, input information which may be an instruction from an NC program, a manual motion block or an incremental motion command is converted into command positions for the machine drives. In order to generate a defined relative motion between tool and workpiece, command variables must be generated and coordinated for the machine axes involved. In this context, we refer to axes with a channel relationship. The number of axes controlled in a channel and the type of axes – translatory or rotary – are dependent on the machine concept.

For example, in the case of five-axis milling, three translatory and two rotary axes are controlled in one channel. Today, some machines offer the function of controlled their axes in several channels; an automatic multi-slide lathe is a typical example of this. The specific path motions on these machines are specified in the program for the individual channels and the corresponding command variables are generated independently of each other. However, it is possible to incorporate synchronisation points in each of NC programs to coordinate the channels.

Controls allow these machine concepts so that the generation of command variables is organised in separate channels. Accordingly, a separate NC program is started on each channel. Cross-channel synchronisation is executed either using NC commands or via the PLC.

## 4.2 Description of the channel-specific interface

### 4.2.1 Status information of a channel

Covered block motion path	
Description	<p>Part of the path motion traversed in the current block in relation to the total path. This status datum contains the current block position referred to the path distance in space in the motion block in per mil <math>sd(t)</math>.</p> 
Signal flow	CNC → PLC
Unit	0.1 %
ST path	<code>pMC[channel_idx]^addr^.StateBahn_Data.D_CoveredDistance</code>
Data type	DINT
Access	PLC is reading
Special features	<p>If a main axis participates in the motion, the covered path motion is in relation to the block path of the first three axes. If no main axis participates in the motion, the covered path motion is the position lag with the longest motion time in relation to the block path.</p>

Currently covered path in the NC program (PCS)	
Description	Reads the currently covered path in the NC program since program start or since the last #DISTANCE PROG START CLEAR NC command. The calculation is based on the current position in the current NC block.
Signal flow	CNC → PLC
Unit	0.1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_DistProgStartHigh pMC[channel_idx]^addr^.StateBahn_Data.D_DistProgStartLow
Data type	UDINT
Access	PLC is reading
Special features	In the NC this is an integer number which occupies 8 bytes in the memory. At the HLI the number is provided in the form of two 4-byte wide values. The value in D_DistProgStartLow contains the 4 lower bytes 0 to 3 and the value in D_DistProgStartHigh contains the higher bytes 4 to 7 of the 8-byte value present in the NC kernel.  The read value can be used to command the block search to define the covered path in the NC program from where actual machining should effectively start.

Line counter, NC program	
Description	The datum indicates the NC program line which is the source of the command just processed by the interpolator.  The value is derived from the number of NC program lines which the decoder has read since the NC program started. All the lines read the decoder are counted, i.e. repeatedly read lines, empty and comment lines. All commands to the interpolator resulting from decoding a NC program line are assigned to the associated line counter.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_BlockCount
Data type	UDINT
Access	PLC is reading

Programmed path feed	
Description	Path feed was programmed by the F<value> in the NC program.
Signal flow	CNC → PLC
Unit	1 µm/s
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_CommandFeed
Data type	DINT
Access	PLC reads



Path feed programmed taking into account real-time influences	
Description	Path feed which was programmed by the F<value> in the NC program weighted with the current real-time influences, e.g. override.
Signal flow	CNC → PLC
Unit	1 µm/s
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_CommandFeedActive
Data type	DINT
Access	PLC reads

Current path feed	
Description	Current path feed during interpolation.
Signal flow	CNC → PLC
Unit	1 µm/s
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_ActiveFeed
Data type	DINT
Access	PLC reads

Due to the architecture of the NC kernel in which various components operate asynchronously to each other, certain status information may be provided several times by the various control components.

For example, the signal X\_ProgramEnd in the status flag of the decoder means that the decoder has completed decoding the program while the actual interpolation by the path interpolator may not yet be completed. The end of interpolation of the path axes is indicated by the signal X\_ProgramEnd in the status data of the path interpolator.

End of program reached	
Description	Interpolator has reached end of program
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_ProgramEnd
Data type	BOOL
Value range	[TRUE = End of program reached, FALSE]
Access	PLC reads

End of program reached	
Description	Decoder has reached end of program.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^.addr^.StateDecoder_Data. <b>X_ProgramEnd</b>
Data type	BOOL
Value range	[TRUE = End of program reached, FALSE]
Access	PLC reads
Special features	<b>Caution:</b> The machine cannot move any further since the interpolation signal is relevant for machine motion.

End of program reached	
Description	The look ahead function has reached the end of program
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^.addr^.StateBavo_Data. <b>X_ProgramEnd</b>
Data type	BOOL
Value range	[TRUE = End of program reached, FALSE]
Access	PLC reads
Special features	Required for diagnosis only.

Stop condition	
Description	Displays the condition why the current motion was stopped.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^.addr^.StateBahn_Data. <b>D_StopConditions</b>
Data type	DINT
Value range	See table: Value range of stop conditions [► 90]
Access	PLC is reading

Value range of stop conditions

Constant in PLC	Value	Explanation
SC_BIT_FEEDHOLD	0x0001	Path feed stop
SC_BIT_VFG	0x0002	No axis-specific feed enable.
SC_BIT_SINGLE_BLOCK	0x0004	Single step mode active.
SC_BIT_M00_OR_M01	0x0010	M00 (programmed stop), M01 (optional stop) is active.
SC_BIT_PLC_ACKNOWLEDGE	0x0020	Stop occurs due to waiting for an acknowledgement from the SPS. This may occur as a result of the output of M or H technology functions but is not restricted to them alone.
SC_BIT_OVERRIDE_ZERO	0x0040	Override = 0.
SC_BIT_DELAY_TIME	0x0200	Dwell time.
SC_BIT_CHANNEL_SYNC	0x0800	Channel synchronisation is active.
SC_BIT_IPO_INPUT_EMPTY	0x1000	Input FIFO of the interpolation is empty.
SC_BIT_IPO_INPUT_DISABLED	0x2000	Input of function blocks (e.g. motion blocks etc.) disabled.
SC_BIT_WAIT_FOR_AXES	0x8000	Stop occurs due to waiting until a commanded axis swap is completed.
SC_BIT_CHANNEL_ERROR	0x00010000	An error occurred in the channel.
SC_BIT_WAIT_TECHNO_ACK	0x00020000	Waiting for acknowledgement of M/H/ST technology functions.
SC_BIT_W_C_AFTER_COLLISION	0x00040000	After a detected collision, waiting for motion resumption.
SC_BIT_SLOPE_SUPPLY_PROBLEM	0x00080000	Block supply problem (only occurs in conjunction with HSC slope).
SC_BIT_BACK_INTERPOLATION	0x00100000	Back interpolation after tracking mode is active.
SC_BIT_BREAKPOINT_STOP	0x00400000	Stop because a breakpoint (stop point) is reached; available as of V2.11.2024.03, V2.11.2807.01, V3.1.3039.01.
SC_BIT_M0_STOP	0x02000000	Stop after an M00 function is reached
SC_BIT_M1_STOP	0x04000000	Stop after an M01 function is reached
SC_BIT_DEC_SYN_CHAN_EMPTY	0x10000000	Decoder is waiting for synchronisation. NC channel has no job.

**Error occurred – awaiting clearance**

Description	An internal error has occurred. The interpolator waits for the error to be cleared.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_WaitErrorRemoval
Data type	BOOL
Value range	[TRUE = Error occurred - interpolator waiting, FALSE]
Access	PLC reads

**Error occurred– wait for external input**

Description	In syntax check mode, the decoder waits after an error for an external input (continue, abort)
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateDecoder_Data.X_WaitAfterError
Data type	BOOL
Value range	[TRUE = Error occurred – decoder waiting, FALSE]
Access	PLC reads

**Interpolator active**

Description	Machine is to be/is moved.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_InterpolationActive
Data type	BOOL
Value range	[TRUE = Machine is to be/is moved, FALSE]
Access	PLC reads

**Technology function acknowledgement**

Description	The interpolator waits for the acknowledgement of a technology function from the PLC.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_WaitTechnoAcknowledge
Data type	BOOL
Value range	[TRUE = Waiting for acknowledgement of technology function, FALSE]
Access	PLC reads

### Enable continuation of motion

Description	The interpolator waits for an Enable to continue the motion after a stop in single-step mode.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_WaitContinue
Data type	BOOL
Value range	[TRUE = Interpolator waiting motion to continue, FALSE]
Access	PLC reads

### Dwell time active

Description	The interpolator waits due to a programmed dwell time (G04).
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_DwellTimeActive
Data type	BOOL
Value range	[TRUE = Interpolator waiting, FALSE]
Access	PLC reads

### Axis group in position

Description	All axes in the axis group have reached their programmed end positions.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_AxesInPosition
Data type	BOOL
Value range	[TRUE, FALSE]
Access	PLC reads

### Waiting for axis group in position

Description	The interpolator waits in single-block mode or for all axes to be in position with programmed stop (M00) or optional stop (M01).
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_WaitAxesInPosition
Data type	BOOL
Value range	[TRUE, FALSE]
Access	PLC reads

### Waiting for requested axis

Description	The look ahead function waits for a programmed axis request (see [PROG//#CALL AX]) that it contains the axis.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBavo_Data.X_WaitForAxis
Data type	BOOL
Value range	[TRUE = Waiting for requested axis, FALSE]
Access	PLC reads

### Block search active

Description	The interpolator works in block search mode.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_BlockSearchActive
Data type	BOOL
Value range	[TRUE = active - Interpolator works in block search mode., FALSE]
Access	PLC is reading

### Block search active

Description	Look ahead works in block search mode
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBavo_Data.X_BlockSearchActive
Data type	BOOL
Value range	[TRUE = active – Look ahead works in block search mode, FALSE]
Access	PLC reads

### Block search active

Description	The decoder works in block search mode
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateDecoder_Data.X_BlockSearchActive
Data type	BOOL
Value range	[TRUE = active - Decoder works in block search mode, FALSE]
Access	PLC reads

### Block search, distance from continuation position

Description	If a NC program is started with block search mode, the NC program is processed in simulative mode (with no path motion) up to the specified continuation position. Block search is then in the HLI_BS_WAIT_FOR_PLC_OFF state and calculates the distance between the actual positions of the axis and the continuation position. If block search is in the HLI_BS_RETURNING_TO_CONTOUR state, this value is refreshed cyclically.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_BlockSearchPathDeviation
Unit	0.1 µm
Data type	UDINT
Value range	[0, MAX_SGN32]
Access	PLC is reading

### Block search, state

Description	Indicates the current state of the block search mode in the interpolator.	
Signal flow	CNC → PLC	
ST path	pMC[channel_idx]^addr^.StateBahn_Data.W_BlockSearchState	
Data type	INT	
Value range	<b>Constant</b>	<b>Value</b>
	HLI_BS_INACTIVE	0
	HLI_BS_WAIT_FOR_PLC_ON	1
	HLI_BS_ACTIVE	2
	HLI_BS_WAIT_FOR_PLC_OFF	3
	HLI_BS_WAIT_RETURN_TO_CONTOUR	4
	HLI_BS_RETURNING_TO_CONTOUR	5
	HLI_BS_WAIT_FOR_CONTINUE_CONTOUR	6
Access	PLC is reading	

### Path velocity below limit

Description	The path velocity undershoots the parametrised limit.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_SpeedLimitDetect
Data type	BOOL
Value range	[TRUE = active – Path velocity undershoots parametrised limit, FALSE]
Access	PLC reads

### Rapid traverse velocity, axes move in the channel

Description	If the value is TRUE, the path axes move on the programmed path and the rapid traverse velocity was specified as motion velocity. TRUE is only indicated if at least one axis actually moves.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_RapidMode
Data type	BOOL
Value range	[TRUE = at least one path axis moves and rapid traverse velocity is specified, FALSE]
Access	PLC reads

### Collision detected, wait to continue motion

Description	Indicates that after a collision is detected, the axis waits for a command to continue the motion.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_WaitContinueAfterCollision
Data type	BOOL
Value range	[TRUE = after a collision is detected, wait to continue the path, FALSE]
Access	PLC reads

### Block number, current path motion

Description	If the N function [PROG//N function] is used in the active NC program to program NC block numbers, this datum indicates the NC block number of the NC block currently processed in the interpolator in this datum.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_BlockNumber
Data type	UDINT
Value range	[0, MAX_UN32]
Access	PLC reads



NC block, inserted	
Description	Indicates whether an additional NC block was inserted by the control unit during interpolation. Additional NC blocks may be created by functions such as polynomial contouring or tool radius compensation.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_BlockInserted
Data type	BOOL
Value range	[TRUE = the control unit inserted an NC block, FALSE]
Access	PLC reads

Manual mode is active, without parallel interpolation	
Description	Indicates whether exclusive manual mode is active. No interpolation is active at the same time.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_G200Active
Data type	BOOL
Value range	[TRUE = Manual mode is active without parallel interpolation, FALSE]
Access	PLC reads

Manual mode active, with parallel interpolation	
Description	Indicates whether superimposed manual mode is active. This means that the setpoints for the axes to be moved are calculated by superimposing the defaults for path interpolation and the mode interface of a particular axis.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_G201Active
Data type	BOOL
Value range	[TRUE = Manual mode is active with parallel interpolation, FALSE]
Access	PLC reads

**Measurement process, active**

Description	Indicates that a measurement process was commanded by G100 or G108. The rising edge of the datum indicates that the measurement process was started. The value then remains TRUE until a measurement event was triggered for all the axes participating in the measurement run and for which measurement was activated.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_MeasureActive
Data type	BOOL
Value range	[TRUE = A measurement process is commanded but the measurement results have not yet been received for all measuring axes, FALSE]
Access	PLC reads

**Delete distance to go, state**

Description	If a command is sent via the DeleteDistanceToGo [► 104] control unit, this value is TRUE as long as the NC block which exerts a linear motion to the target position of the next motion block is executed (short cut). It also remains TRUE if the active short cut is itself shortened by a new command.  The signal is again reset when the active NC block is no longer related to the command of the DeleteDistanceToGo [► 104] control unit. See functional description [FCT-C28].
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_DeleteDistanceToGoActive
Data type	BOOL
Value range	[TRUE = linear motion is executed to the target position of the next motion block (short cut), FALSE]
Access	PLC reads

**Time to next motion command containing G01, G02**

Description	Is rapid traverse velocity is specified for the current path motion, this datum indicates the time until the next motion block containing G01 or G02 is active.  This time is only calculated and indicated if it is enabled in the start-up list in the parameter P-STUP-00070 by specifying FCT_LOOK_AHEAD_STANDARD   FCT_CALC_TIME
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_TimeToNextFeedBlock
Data type	UDINT
Value range	[0, MAX_UN32]
Access	PLC reads

### Distance to edge below limit

Description	This signals that the distance to the edge defined by the channel parameter P-CHAN-00222 (edge_machining.pre_dist) was undershot. As from this time, motion on the programmed path is effective at the path feed defined by the channel parameter P-CHAN-00223 (edge_machining.pre_feed).
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIEdgeFunction_Data.X_Signal_1
Data type	BOOL
Value range	[TRUE = active – Distance to edge undershoots parametrised limit, FALSE]
Access	PLC reads

### Wait time at edge

Description	The signal indicates that the motion was stopped at the edge and the wait time specified by the channel parameter P-CHAN-00224 (edge_maching.wait_time) is expiring.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIEdgeFunction_Data.X_Signal_2
Data type	BOOL
Value range	[TRUE = active – Wait time active, FALSE]
Access	PLC reads

### Distance after edge below limit

Description	This signals that the tool moves away from the edge but the distance to the edge is still smaller than the value defined by the channel parameter P-CHAN-00225 (edge_machining.post_dist). The path feed valid for this section is defined by the channel parameter P-CHAN-00226 (edge_machining.post_feed).
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIEdgeFunction_Data.X_Signal_3
Data type	BOOL
Value range	[TRUE = active - Distance after edge not reached, FALSE]
Access	PLC reads

### Angle between active and next NC block

Description	Indicates the angle between 2 consecutive NC blocks if both NC blocks are motion blocks containing programmed feed. The indicated value is in the range of [0, 1800000] which corresponds to [0°, 180°].
Signal flow	CNC → PLC
Unit	0.0001°
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIEdgeFunction_Data.D_AngleEnd
Data type	DINT
Value range	[0, 10000000]
Access	PLC reads
Special features	<p>If the the next motion block is a rapid traverse block, the value value is 5000000</p> <p>NC block leading to motion stop, the output value is 6000000</p> <p>.</p> <p>In all other cases the default value 10000000 is indicated.</p>

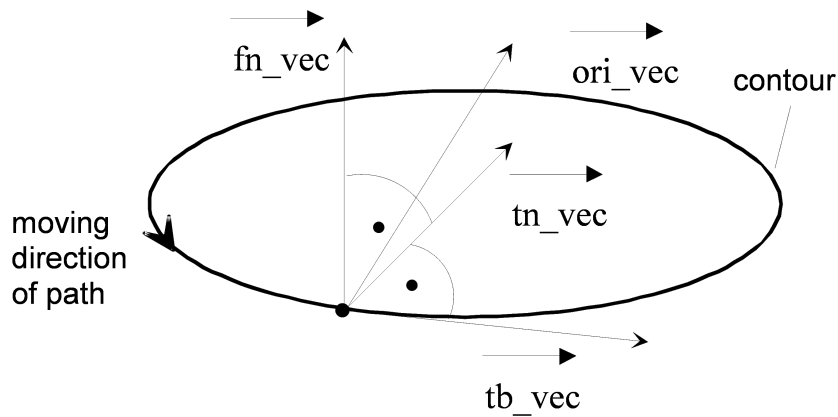
### Velocity at end of current NC block

Description	<p>Indicates the velocity at transition from currently interpolated to next motion block which results from evaluating the geometrical situation and the velocity of the next block.</p> <p>If the dwell time is programmed or a predictable motion stop is programmed at the transition between the NC blocks caused by the output of a technology function with corresponding synchronisation start in the next block, the value 0 is indicated.</p>
Signal flow	CNC → PLC
Unit	1 µm/s
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBlockDynamic_Data.D_VelEndGeo
Data type	DINT
Value range	[0, MAX_SGN32]
Access	PLC reads

### 4.2.1.1

### Status information on tool orientation

The 3 vectors `tb_vec` (path tangent), `tn_vec` and `fn_vec` form a complete right-handed space coordinate system (moving trihedron).



**Fig. 12: Vectors of the tool coordinate system**

**NOTE:**

When tool geometry compensation is active and when machining with the face of the tool, the result is the tool direction vector `ori_vec` obtained from the face normal vector `fn_vec`, the path tangent vector `tb_vec` and the advance and lateral angle.

(END)

The vectors of the tool coordinate system are integrated in the data structure `HLIToolPathDisp-Data`. The tables below contain the description of the elements for this data structure:

Tool direction vector	
Description	Components of the tool direction vector or recalculation from face normal vector and path tangent vector (see Figure)
ST path	<code>pMC[channel_idx]^addr^.StateBahn_Data.HLIToolPath_Data.ori_vec[vec_idx]</code>
Data type	ARRAY [1..HLI_CS_AXES_NR] OF DINT
Special features	The direction vector is normalised to length $10^6$ .

Path tangent vector	
Description	Components of the path tangent vector (see Figure).
Unit	Direction vector normalised to length $10^6$
ST path	<code>pMC[channel_idx]^addr^.StateBahn_Data.HLIToolPath_Data.tb_vec[vec_idx]</code>
Data type	ARRAY [1..HLI_CS_AXES_NR] OF DINT
Special features	The direction vector is normalised to length $10^6$ . The last valid motion direction is retained.

Normal vector to the path tangent	
Description	Components of the resulting vector, cross product of <code>ori_vec</code> $\times$ <code>tb_vec</code> , or cross product of <code>fn_vec</code> $\times$ <code>tb_vec</code> (see Figure).
ST path	<code>pMC[channel_idx]^addr^.StateBahn_Data.HLIToolPath_Data.tn_vec[vec_idx]</code>
Data type	ARRAY [1..HLI_CS_AXES_NR] OF DINT
Special features	The direction vector is normalised to length $10^6$ .

Auxiliary vector of the complete trihedron, face normal vector	
Description	Components of the resulting vector, cross product of <code>tb_vec</code> $\times$ <code>tn_vec</code> , or face normal vector (see Figure).
ST path	<code>pMC[channel_idx]^addr^.StateBahn_Data.HLIToolPath_Data.fn_vec[vec_idx]</code>
Data type	ARRAY [1..HLI_CS_AXES_NR] OF DINT
Special features	The direction vector is normalised to length $10^6$ .

#### 4.2.1.2 Diagnosis upload

Diagnosis upload	
Description	While the CNC is running, the PLC can command an upload of diagnosis data using this control unit. The control unit is enabled by X_Enable = TRUE.
Data type	MCControlBoolUnit, see description of Control Unit
Data type	MCControlBoolUnit
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCControlBahn_Data. <b>MCControlBoolUnit_DiagnosisUpload</b>
Commanded, requested and return values	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = diagnosis upload activated, FALSE = diagnosis upload off]
Redirection	
ST element	<b>.X_Enable</b>
Special feature	<b>Note:</b> The data item X_Command must remain at TRUE until X_State reverts to FALSE. Otherwise, the data is not complete since the diagnosis data upload is aborted.

## 4.2.2 Control commands of a channel

Skip mode, NC block	
Description	Activates/deactivates skip mode at interpreter level for the NC program. The status of skip mode is only evaluated at the start of the NC program. Switchover during execution of an NC program has no effect.
Data type	MCControlBoolUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCControlDecoder_Data.MCControlBoolUnit_ProgramBlockIgnore
Commanded, requested and return values	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Skip mode NC block ON, FALSE = Skip mode NC block OFF, FALSE]
Redirection	
ST element	.X_Enable

Single-block mode at interpreter level NC block	
Description	Activates/deactivates single-block mode at interpreter level. A restart must be present for each block
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	<b>Is currently not supported</b>
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlDecoder_Data.MCControlBoolUnit_SingleBlock
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = single-block mode ON, FALSE]
Redirection	
ST element	.X_Enable



Enable processing of next NC block	
Description	Enable to process the next NC block. This element serves to synchronise the geometry chain with the PLC. For this purpose, the PLC blocks the single-block step enable and thus stops the interpreter.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	<b>Is currently not supported.</b>
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlDecoder_Data.MCControlBoolUnit_ContinueMa- chining
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Continuation NC program decoding	
Description	Continues decoding an NC program after an error has occurred if syntax check (simulation mode) and interactive step enabling of decoding is activated (decoder parameter, characteristic parameter: syn_chk.interaktiv = 1).
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	<b>Is currently not supported.</b>
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlDecoder_Data.MCControlBoolUnit_ReleaseStop
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Stop NC program decoding	
Description	Stops decoding an NC program if interactive enabling of further processing of the NC program was activated in Syntax Check Simulation mode.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	<b>Is currently not supported.</b>
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlDecoder_Data.MCControlBoolUnit_DecStop
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Channel mode			
Description	Selection of a special channel mode such as syntax check or machining time calculation		
Data type	MCCControlSGN32Unit, see description of Control Unit		
Access	PLC reads Request + State and writes Command + Enable		
ST Path	pMC[channel_idx]^ .addr^ .MCCControlDecoder_Data. <b>MCCControlSGN32Unit_Execution-Mode</b>		
Commanded, requested and return values			
ST Element	<b>.D_Command</b> <b>.D_Request</b> <b>.D_State</b>		
Data type	DINT		
Value range	<b>Value</b>	<b>Constant</b>	<b>Meaning</b>
	0x0000	ISG_STANDARD	Normal mode
	0x0001	SOLLKON	Block search
	0x0002	SOLLKON	Nominal contour visualisation simulation with output of visualisation data
	0x0802	SOLLKON_SUPRESS_O UTPUT & SOLLKON	Nominal contour visualisation simulation without output of visualisation data
	0x0004	ON_LINE	Online visualisation simulation
	0x0008	SYNCHK	Syntax check simulation
	0x0010	PROD_TIME	Simulation machining time calculation (No function with TwinCAT)
	0x0020	ONLINE_PROD_TIME	Simulation online machining time calculation
	0x0040	MACHINE_LOCK	Dry run without axis motion
	0x0080	ADD_MDI_BLOCK	Extended manual block mode: the end of a manual block is not evaluated as a program end. It permits the commanding of further manual blocks.
	0x0100	KIN_TRAFO_OFF	Overwrites automatic enable for kinematic transformations by a characteristic parameter defined in the channel parameters (sda_mds*.lis).
	0x1000	BEARB_MODE_SCENE	When SCENE mode is enabled, the output of #SCENE commands is activated on the interface (see also [FCT-C17// Scene contour visualisation]).  An additional client is linked to this output via DataFactory / CORBA.
	0x2000	SUP- PRESS_TECHNO_OUT- PUT	Without output of technology functions (M/H/T). Set implicitly in connection with syntax check

Redirection	
ST element	<b>.X_Enable</b>

Index of position offset group	
Description	<p>A position offset defines an additional offset. For instance, this takes into consideration different clamping positions of a workpiece in a machine workspace.</p> <p>Position offsets are defined in the position offset data. The position offsets for each axis are specified in a position offset group. A position offset group is selected by the group index. Position offset data is evaluated by the CNC at program start.</p>
Data type	MCCControlSGN16Unit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCCControlDecoder_Data. <b>MCCControlSGN16Unit_ClampPosition</b>
Commanded, requested and return value	
ST element	<b>.D_Command</b> <b>.D_Request</b> <b>.D_State</b>
Data type	INT
Value range	[0, 67]
Redirection	
ST element	<b>.X_Enable</b>

Feedhold ON/OFF	
Description	Channel-specific feedhold. When this element is set to TRUE , the feed velocity is immediately ramped down to feed = 0 during interpolation based on the acceleration settings.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	The channel-specific parameter P-CHAN-00097 can be used to exert additional influence on the parameterised accelerations to be used.  Feedhold ON/OFF also be commanded by the control unit for switching operation modes (see Sec. 8.2). A HOLD command results in stopping the channel; a RESUME command cancels the stop command.  If the PLC is registered at both control units, pay attention to the following safety note:
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_FeedHold
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Feedhold ON, FALSE]
Redirection	
ST element	.X_Enable



### CAUTION

**Stop command to CNC is not executed.**

Possible damage to machine.

Similar to other control units, a command from an external application, e.g. an HMI, is displayed in **X\_Request** . If the PLC operates this control unit and therefore sets **X\_Enable** to TRUE, the command from the external application only becomes effective if **X\_Request** to **X\_Command** . However, the semaphores must be operated as usual.

This must also be considered if the PLC operates the control unit in order to switch operation modes (see sec. 8.2) and therefore sets **X\_McmEnable** to TRUE. A HOLD command sent by this control unit triggers the NC kernel to send a request to activate feedhold. This is then displayed in the **X\_Request** of the control unit described here and only becomes effective when the PLC copies **X\_Request** to **X\_Command** . This also applies to cancelling feedhold.

The figures below depict these functions.

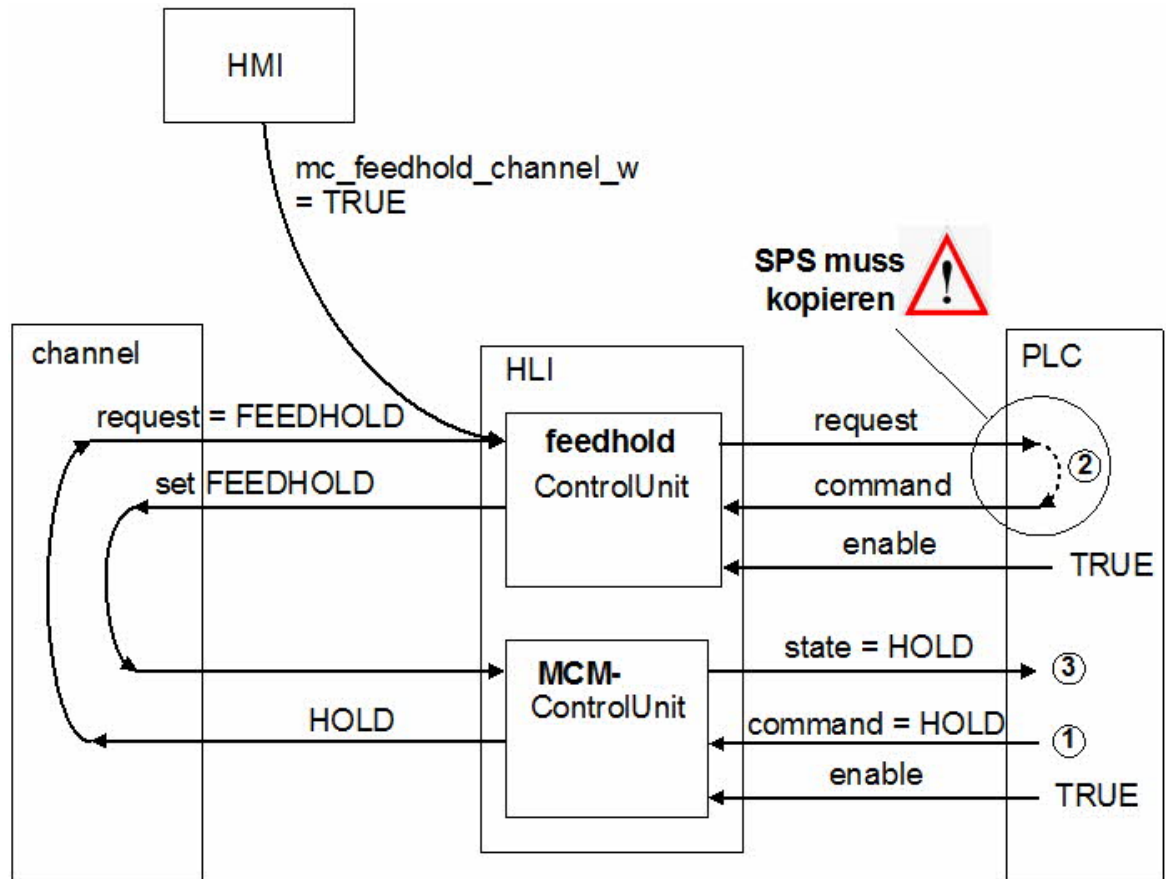


Fig. 13: Interaction between feedhold and NC channel stop

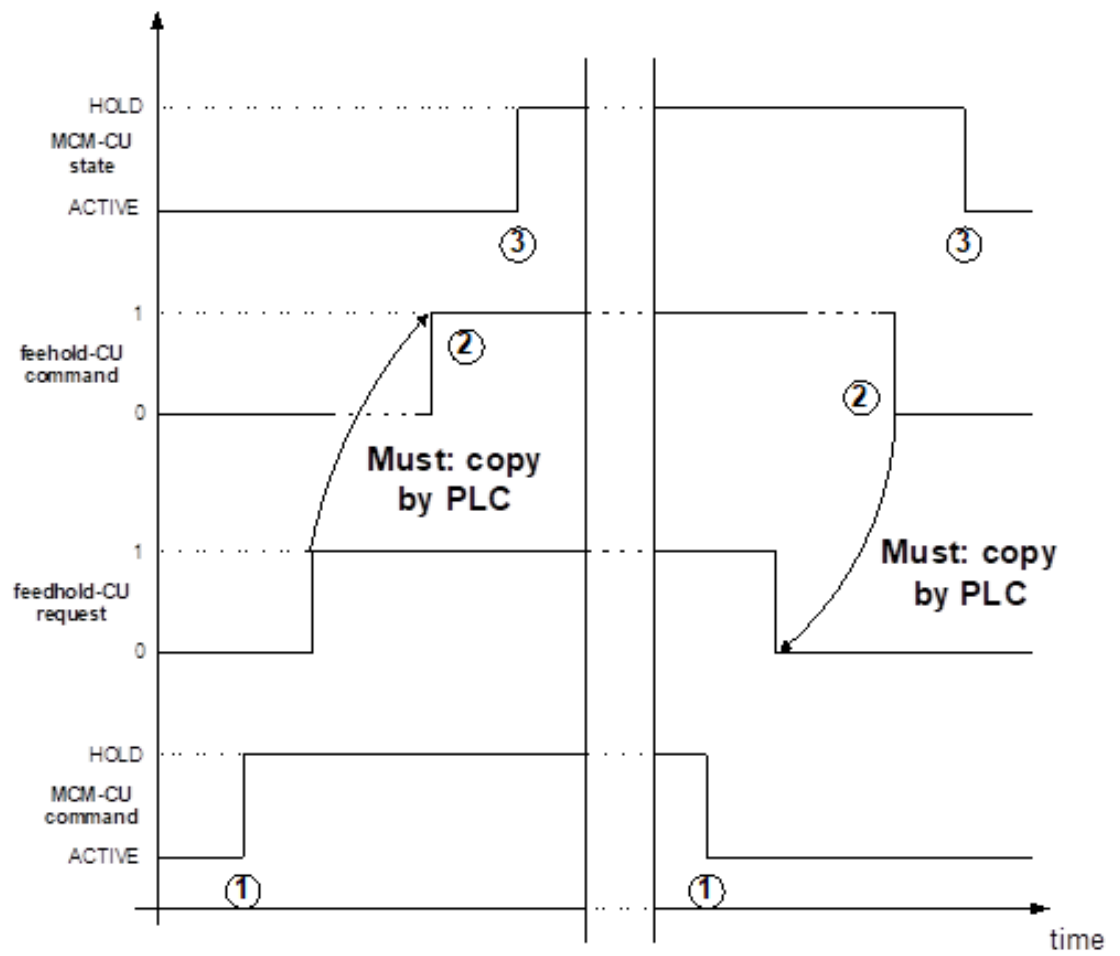


Fig. 14: Time sequence of feedhold and NC channel stop

Emergency stop, channel	
Description	<p>If this emergency stop element is set active (TRUE), interpolation is aborted immediately by output of the zero setpoint or by a deceleration as specified in the emergency deceleration. The NC kernel reverts to error state.</p> <p>This type of motion influence has maximum priority.</p>
Data type	MCCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlBoolUnit_EmergencyStop
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Emergency stop active, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

Single-block mode	
Description	<p>Activating/deactivating single-block mode.</p> <p>Single-block mode refers only to motion blocks. As long as single-block mode is activated, the system is decelerated to feed rate = 0 at the end of each motion block. The following blocks can then only be executed by setting the element "continue motion" if all axes are located in the control window.</p>
Data type	MCCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlBoolUnit_SingleBlock
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	<b>.X_Enable</b>



Optional stop	
Description	<p>Activating/deactivating optional stop.</p> <p>If the function <b>M01</b> (optional stop) is programmed in the current block of the NC program, set this element to the value TRUE to stop at block end (ramped-down deceleration complying with the permissible accelerations).</p> <p>The following block can be enabled by activating the element "continue machining" if the NC kernel indicates that all axes are in the control window by resetting status flag X_WaitAxesInPositiont.</p>
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlBahn_Data.MCControlBoolUnit_M01StopEnable
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE = Optional stop active, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

Resumption of motion	
Description	<p>If program execution is interrupted by selecting "Single block mode" [► 104] or "Optional stop" [► 104] or by M00, this control unit can resume NC program execution.</p> <p>A falling edge for the command value (command_w) of the control unit "Continue motion", i.e. a transition from TRUE to FALSE, leads to a resumption of NC program execution. The condition for this is that all axes are located in the control window.</p>
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	<b>Falling</b> edge of the command resumes NC program execution.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlBahn_Data.MCControlBoolUnit_ContinueMotion
Commanded, requested and return value	
ST element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

Machining simulation ON/OFF	
Description	Activating/deactivating machining simulation. During machining simulation, all technology functions of the NC program are not output to the PLC but are acknowledged internally.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlBahn_Data.MCControlBoolUnit_MachiningSimulation
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Machining simulation active, FALSE = Machining simulation inactive]
Redirection	
ST element	.X_Enable

Feed override	
Description	Feed override allows the programmed path velocity to be weighted with an additional factor
Data type	MCControlUNS16Unit, see description of control unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlBahn_Data.MCControlUNS16Unit_OverrideFeedrate
Commanded, requested and return value	
ST element	.D_Command .D_Request .D_State
Unit	0,1 %
Data type	UINT
Value range	[0, P-CHAN-00056] The parameter P-CHAN-00056 is a channel-specific parameter. The typical value is 1000. See [CHAN].
Redirection	
ST element	.X_Enable

Rapid traversing override	
Description	The rapid traverse override can weight G0 path motions by an additional factor. See also special features.
Data type	MCCControlUNS16Unit, see description of control unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^ .addr^ .MCCControlBahn_Data.MCCControlUNS16Unit_OverrideRapid-Move
Commanded, requested and return value	
ST element	.D_Command .D_Request .D_State
Unit	0,1 %
Data type	UINT
Value range	[0, 1000]
Redirection	
ST element	.X_Enable
Special features	
Parameterisation/ mode of operation	<p>Rapid traverse override is only active if this function is also activated in the channel parameter list. Otherwise, there is no distinction made between feed and rapid traverse blocks.</p> <p>Setting options in the channel parameter P-CHAN-00181:</p> <p>Rapid traversing override is inactive. Rapid traversing override acts on feed and rapid traversing blocks.</p> <p>Rapid traversing override is active. Feed override acts on feed blocks and the minimum of feed and rapid traversing override acts on rapid traversing blocks.</p> <p>Rapid traversing override is active. Feed override acts on feed blocks and only rapid traversing override acts on rapid traversing blocks.</p>

Interrupt output of command values to real axes	
Description	<p>This interrupts the output of NC channel command values to physical axes. The NC channel is stopped and assignment to the real axes is disabled.</p> <p>Physical axes can then be requested and moved by another channel. A different logical axis can be linked to a physical axis here.</p> <p>After clearing this interruption, the axes can be requested again and the original channel continues its motion.</p>
Data type	MCControlBoolUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_SuspendAxisOutput
Commanded, requested and return values	
ST Element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	<b>.X_Enable</b>

External path velocity specified	
Description	External path velocity specified. The set path velocity is activated by the control unit MC-ControlBoolUnit_ExtCommandSpeedValid.
Data type	MCCControlUNS32Unit, see description of control unit
Special features	The path velocity transferred by this interface is automatically restricted to the limits defined in the axis parameters.
Unit	1 µm/s
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCCControlBahn_Data.MCCControlUNS32Unit_ExtCommand-Speed
Commanded and requested Value	
ST element	.D_Command .D_Request
Data type	UDINT
Return value	
ST element	.D_State
Data type	UDINT
Special features	<p>The D-State indicates the actual path velocity used in the interpolate, including any influence by override.</p> <p>By default the externally specified velocity only acts on machining motions (G01, G02, G03). The channel parameter P-CHAN-00102 (plc_command_rapid_feed) sets that the externally specified velocity also acts on rapid traverse motions (G00).</p>
Redirection	
ST element	.X_Enable

Activation of external path velocity	
Description	Activating the velocity commanded in the control unit MCControlUNS32Unit_ExtCommandSpeed. To reach the commanded velocity, all axes involved in the motion are accelerated or decelerated.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_ExtCommandSpeedValid
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Input disable, interpolator	
Description	Disabling input of blocks in the interpolator. When the control unit is enabled, the interpolator stops after it has processed the blocks already input. The channel parameter P-CHAN-00267 can define the event when an activated input disable becomes effective, e.g. effective as of the next rapid traverse block.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	See safety not below.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_InputDisable
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable



### ⚠ CAUTION

**Stop command to CNC not executed.**

Possible damage to machine.

Similar to other control units, a command from an external application, e.g. an HMI, is displayed in **X\_Request**. If the PLC operates this control unit and therefore sets **X\_Enable** to TRUE, the command from the external application only becomes effective if **X\_Request** to **X\_Command**. However, the semaphores must be operated as usual.

This must also be considered if the PLC operates the control unit in order to switch operation modes (see sec. 8.2) and therefore sets **X\_McmEnable** to TRUE. A HOLD command given by this control unit triggers the NC kernel to request input disable. This is then displayed in the **X\_Request** of the control unit described here and only becomes effective when the PLC copies **X\_Request** to **X\_Command**. The same also applies to cancelling input disable.

The figures below depict these functions.

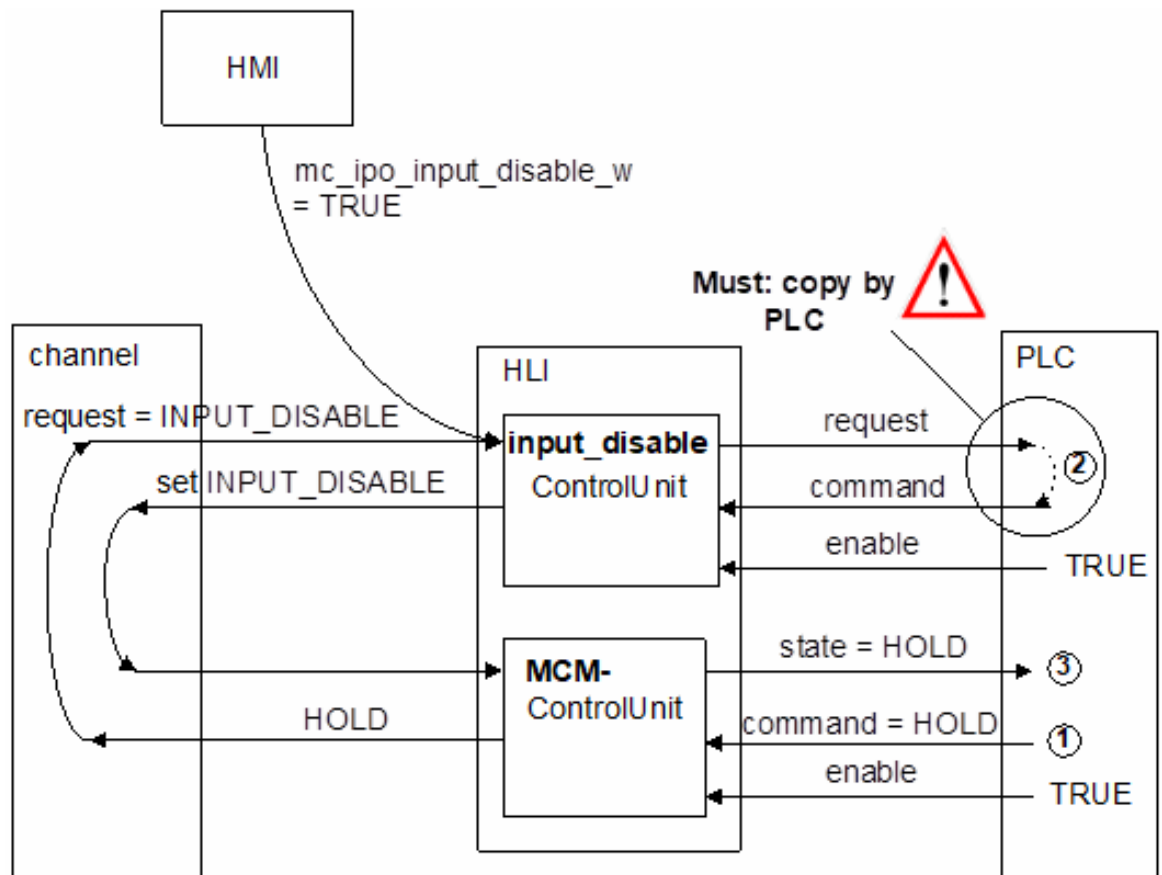


Fig. 15: Interaction of input disable and NC channel stop

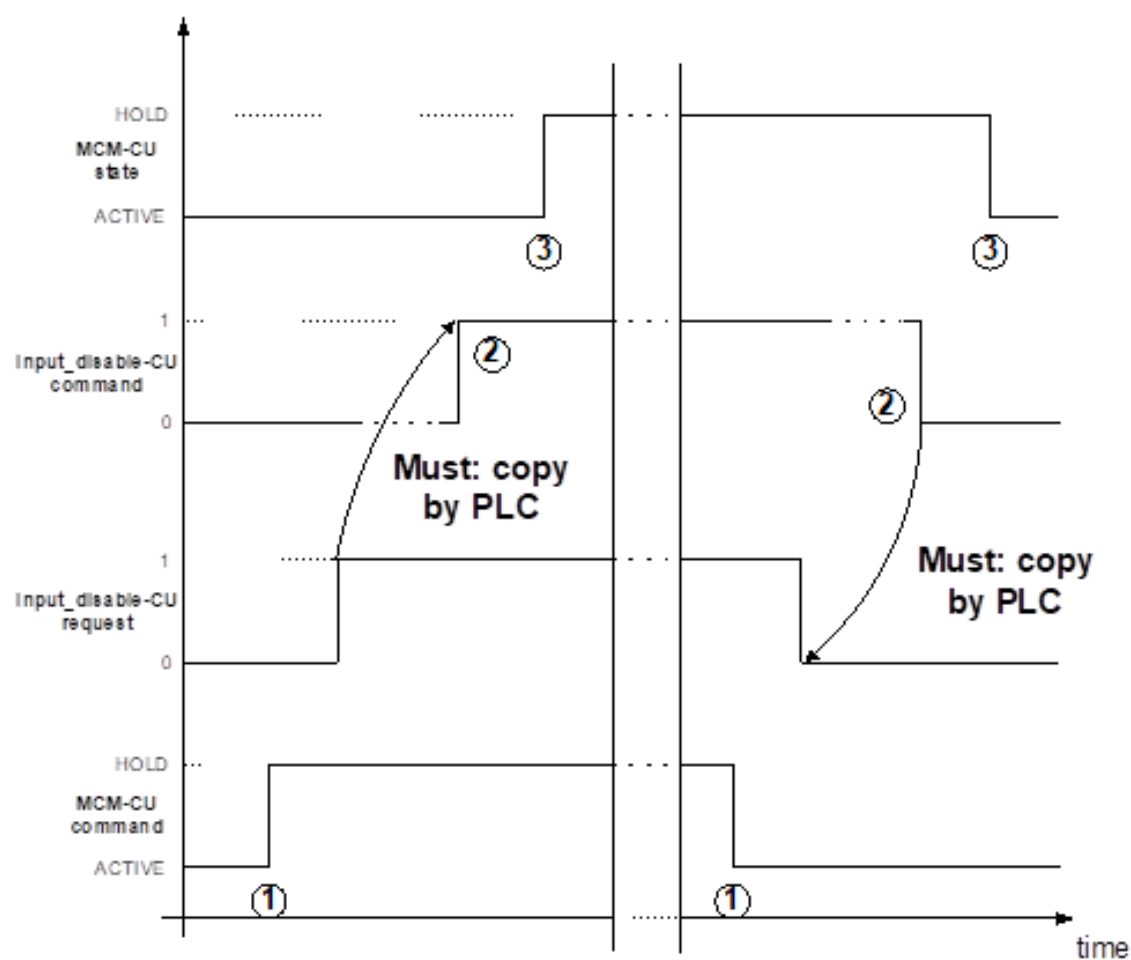


Fig. 16: Time sequence of input disable and NC channel stop



Reduced velocity, channel	
Description	When this signal is set, the path velocity is reduced to the values defined in the axis parameters P-AXIS-00214 and P-AXIS-00155.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	The limits of the axes participating in the motion are taken into consideration. The effective value for reduced velocity is determined so that none of the axes participating in the motion overshoots its configured limit.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_ReducedFeed
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Reduced velocity active, FALSE = Reduced velocity not active]
Redirection	
ST element	.X_Enable

Reduced velocity in zone 1, channel	
Description	When this signal is set, the path velocity is limited to the velocity defined in the axis parameter P-AXIS-00030 if the axis is located within the area defined by the parameters P-AXIS-00085 and P-AXIS-00093. If necessary the axis is decelerated after entering the zone.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_ReducedFeed-Zone
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Reduced velocity in zone 1 active, FALSE = Reduced velocity in zone 1 not active]
Redirection	
ST element	.X_Enable

Reduced velocity in zone 2, channel	
Description	When this signal is set, the path velocity is limited to the velocity defined in the axis parameter P-AXIS-00030 if the axis is located within the area defined by the parameters P-AXIS-00097 and P-AXIS-00105. If necessary the axis is decelerated after entering the zone.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_ReducedFeed-Zone2
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Reduced velocity in zone 2 active, FALSE = Reduced velocity in zone 2 not active]
Redirection	
ST element	.X_Enable

Time override valid	
Description	Time override is activated
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_Override-TimeValid
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Time override is activated, time override is not activated]
Redirection	
ST element	.X_Enable

Time override	
Description	<p>Time override can influence the internal CNC time base for motions. Its effect is similar to slow motion.</p> <p>Time override affects path velocity and acceleration differently.</p> <p><u>Example:</u> If time override is 50% (command_w = 500) the velocity is reduced by a factor of 2 and acceleration by a factor of 4.</p>
Data type	MCCControlUNS16Unit, see description of control unit
Special features	<p>The channel parameter P-CHAN-00111 can be used to affect the effect of time override on the dwell time function.</p> <p>See safety not below.</p>
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlUNS16Unit_OverrideTime
Commanded, requested and return value	
ST element	<p><b>.D_Command</b></p> <p><b>.D_Request</b></p> <p><b>.D_State</b></p>
Unit	0,1 %
Data type	UINT
Value range	[100, 1000]
Redirection	
ST element	<b>.X_Enable</b>



### Attention

#### Time override affects real-time and safety functions.

Delays in the use of these functions may lead to longer reaction times for safety functions under certain circumstances.

**Time override affects path feed and acceleration independent of real-time functions such as feedhold or safety functions such as reduced speed. The user must consider this when using this function.**

Disabling the detection of tool life data	
Description	Detecting tool life data is suppressed when this control unit is activated.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_ToolLifeSup-pressCapture
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Tool life data is not detected, FALSE = Tool life data is detected]
Redirection	
ST element	.X_Enable

Delete distance to go, command	
Description	<p>The rising edge of the commanded value has the effect that the CNC channel is decelerated to feed velocity 0. Then a linear motion is executed to the target position of the next motion block (short cut).</p> <p>The command only affects motion blocks.</p> <p>The functional description [FCT-C28] deals with the topic of "Delete distance to go" in detail.</p>
Data type	MCControlBoolUnit, see description of Control Unit
Peculiarities	See state data X_DeleteDistanceToGoActive [► 87]
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_DeleteDistanceToGo
Commanded, requested and return values	
ST Element	.X_Command .X_Request .X_State (TRUE indicates that the command was detected by the CNC)
Data type	BOOL
Value range	[TRUE = distance to go to be deleted, FALSE = no impacts on motion blocks]
Redirection	
ST Element	.X_Enable

Delete distance to go, end marker	
Description	<p>This control unit sets the end marker defined in the NC program online to valid. A bit mask is defined for this in the NC program.</p> <p>If at least one single bit in the end marker is set on the NC interface, this marker is valid as a jump target.</p> <p>Example:</p> <pre>N10 #DEL DIST2GO [END = '16#0014']</pre> <p>The end marker can be set to valid by bit3 (hexadecimal 4) or by bit5 (hexadecimal 0x10).</p>
Data type	MC_CONTROL_UN32_UNIT, see description of control unit
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlUNS32Unit_DeleteDistanceTo-GoActivation
Commanded, requested and return values	
ST Element	<p><b>.X_Command</b></p> <p><b>.X_Request</b></p> <p><b>.X_State</b></p>
Data type	UDINT
Value range	32-bit
Redirection	
ST Element	<b>.X_Enable</b>

Online tool compensation (OTC)	
Description	<p>The tool radius can be adapted by setting the wear offset depending on wear. [0,1µm] See also Wear compensation of tool radius.</p> <p>It is only possible to used the control unit in the RADIUS or TOOL_DIR modes and in the DISC or AUTO mode.</p> <p>See NC command #OTC.</p> <p>See also functional description [FCT-C20].</p>
Data type	MCCControlSGN32Unit, see description of Control Unit
Peculiarities	The wear offset is not output in a cycle in the CNC. Instead it is output over several cycles.
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlSGN32Unit_OTCRadiusOff-set
Commanded, requested and return values	
ST Element	<b>.D_Command</b> <b>.D_Request</b> <b>.D_State</b>
Data type	DINT
Value range	[-P-TOOL-00031, P-TOOL-00031]
Redirection	
ST Element	<b>.X_Enable</b>

Backward motion	
Description	Select/deselect backward motion on the path
Data type	MCCControlBoolUnit, see description of Control Unit
Access	SPS reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlBoolUnit_BackwardMotion
Commanded, requested and return values	
ST Element	<b>.X_Command</b> <b>.X_Request</b> <b>.X_State</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST Element	<b>.X_Enable</b>

Simulated forward motion	
Description	Select/deselect simulated forward motion on the path For example, M function synchronisations are treated differently.
Data type	MCCControlBoolUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCCControlBahn_Data.MCCControlBoolUnit_SimulateMotion
Commanded, requested and return values	
ST Element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST Element	.X_Enable

Reset backward motion memory	
Description	Deselects backward motion memory No further NC block is saved in the memory. The memory is deleted.
Data type	MCCControlBoolUnit, see description of Control Unit
Access	SPS reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCCControlBahn_Data.MCCControlBoolUnit_ResetBackward-Storage
Commanded, requested and return values	
ST Element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST Element	.X_Enable

## 5 PLC

### 5.1 Control commands to PLC

#### 5.1.1 Reset

PLC reset, axis	
Description	The PLC can be requested to perform a reset via this axis-specific interface. Here, the PLC must indicate that it want to be informed of reset requests by the CNC kernel by setting the <b>X_Enable</b> element.
Data type	MC_CONTROL_BOOL_UNIT, see description of control unit
Special features	<b>Consumption data item</b>
Access	CNC sets <b>X_Command</b> to TRUE to command a reset for the PLC. The CNC sets <b>X_Command</b> to FALSE after the PLC acknowledges execution of the rest by the <b>X_State</b> element.
ST path	pAC[axis_idx]^^.addr^.LcControllpo_Data.LCControlBoolUnit_PLCReset
Commanded value	
ST element	<b>.X_Command</b>
Signal flow	CNC → PLC
Data type	BOOL
Value range	[TRUE = Reset request from CNC to PLC, FALSE]
Return value	
ST element	<b>.X_State</b>
Signal flow	PLC → CNC
Data type	BOOL
Value range	[TRUE = PLC executed reset, FALSE]
Requirement	
ST element	<b>.X_Enable</b>
Signal flow	PLC → CNC
Data type	BOOL
Value range	[TRUE = PLC wants to be notified about requests by CNC, FALSE]



PLC reset, channel	
Description	The PLC can be requested to execute a reset via this channel-specific interface. Here, the PLC must indicate that it wants to be informed of reset requests by the CNC kernel by setting the <b>X_Enable</b> element.
Data type	MC_CONTROL_BOOL_UNIT, see description of control unit
Peculiarities	<b>Consumption data item</b>
Access	CNC sets <b>X_Command</b> to TRUE to command a reset for the PLC. The CNC sets <b>X_Command</b> to FALSE after the PLC acknowledges the execution of the reset via the <b>X_State</b> element.
ST Path	pMC[channel_idx]^addr^.LCControlBahn_Data.LCControlBoolUnit_PLCReset
Commanded value	
ST Element	<b>.X_Command</b>
Signal flow	NCK → PLC
Data type	BOOL
Value range	[TRUE = Reset request from NCK to PLC, FALSE]
Return value	
ST Element	<b>.X_State</b>
Signal flow	PLC → NCK
Data type	BOOL
Value range	[TRUE = PLC executed reset, FALSE]
Request	
ST element	<b>.X_Enable</b>
Signal flow	PLC → NCK
Data type	BOOL
Value range	[TRUE = PLC wants to be notified about requests by CNC, FALSE]

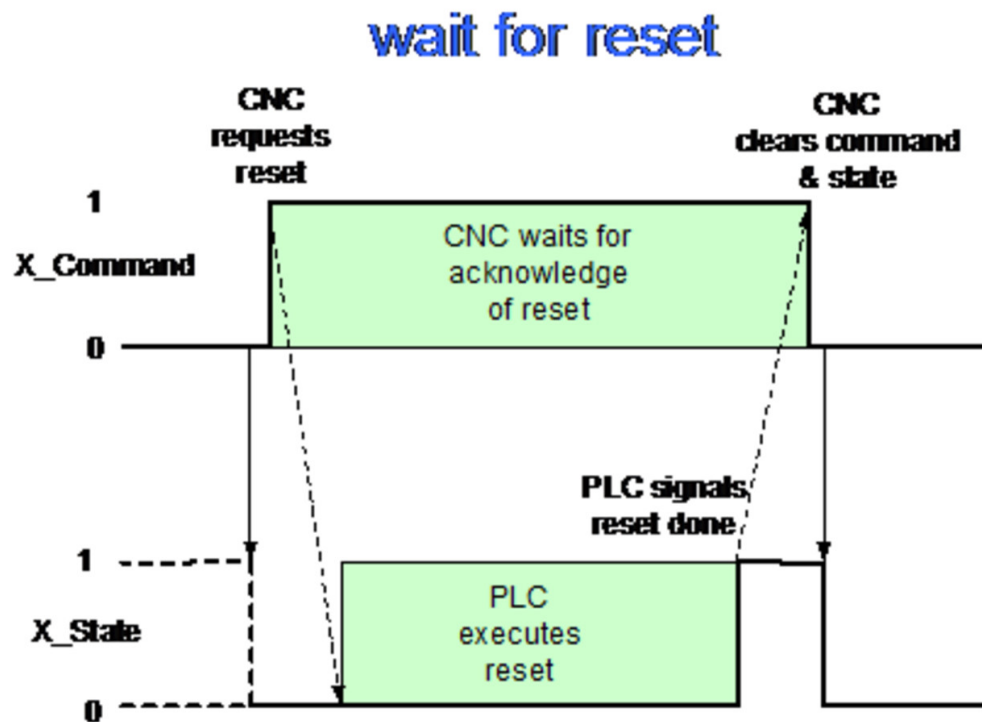


Fig. 17: Interaction between BOOLEAN-MC control unit and PLC



#### Notice

Before a new request and after detecting the reset acknowledgement, the CNC clears the **X\_State** signal.

### 5.1.2

### Block search

Block search on/off to PLC	
Description	
Data type	MC_CONTROL_BOOL_UNIT, see description of control unit
Special features	<b>Consumption data item</b>
Access	<p>The NC kernel sets <b>X_Command</b> to TRUE when block search is to be enabled and then waits for acknowledgement from the PLC. The PLC acknowledges block search enable by setting the element <b>X_State</b>.</p> <p>Consequently, the NC kernel sets the <b>X_Command</b> to FALSE when block search is disabled and then waits for acknowledgement from the PLC which indicates this by cancelling the element <b>X_State</b>.</p>
ST path	pMC[channel_idx]^addr^.LCControlBahn_Data.LCControlBoolUnit_BlockSearch
Commanded value	
ST element	<b>.X_Command</b>
Signal flow	CNC → PLC
Data type	BOOL
Value range	[TRUE = Block search is to be enabled, FALSE]
Return value	
ST element	<b>.X_State</b>
Signal flow	PLC → NCK
Data type	BOOL
Value range	[TRUE = PLC enabled block search, FALSE]
Requirement	
ST element	<b>.X_Enable</b>

## wait for block skip on / off

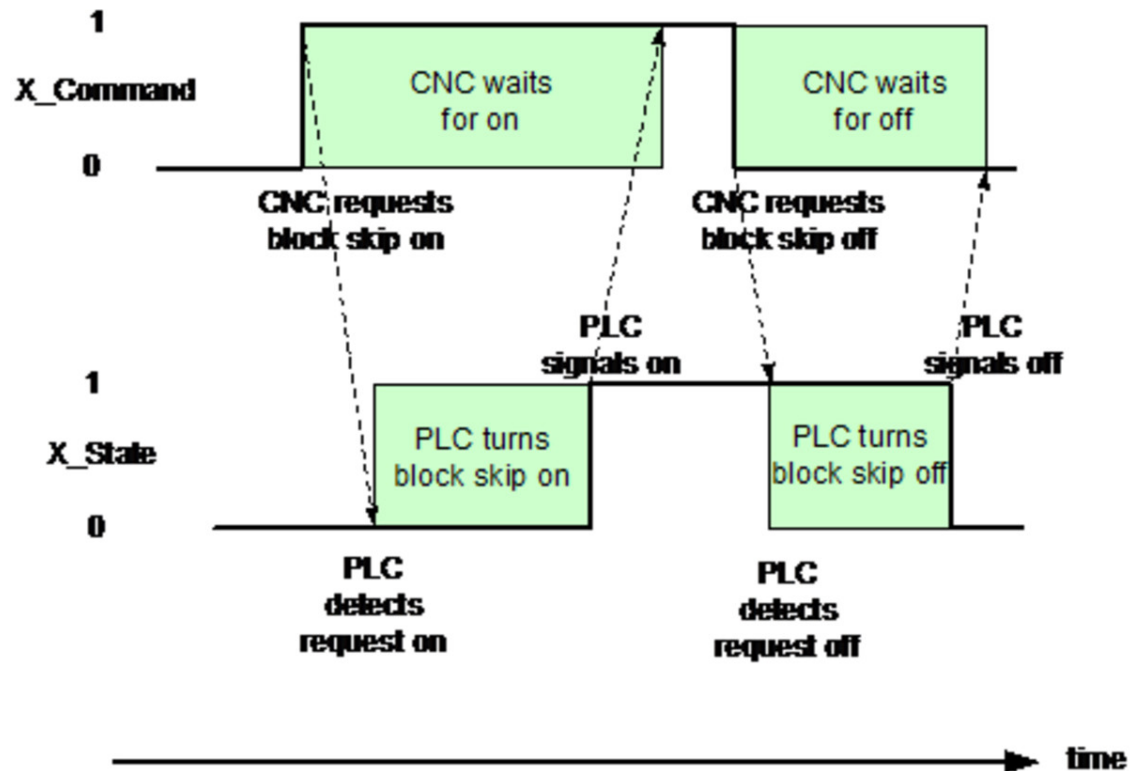


Fig. 18: Interaction between BOOLEAN-MC control unit and PLC



### Notice

When the CNC is reset, the CNC resets **X\_Command** and **X\_State**.

## 6 Technology processes

### 6.1 Introduction

Technology functions can be defined for each channel and each axis.

Channel-specific technology functions are defined in the channel parameters; axis-specific technology functions are defined in the axis parameters of each axis.

This definition also includes the synchronisation mechanism of the technology function. A distinction is made between two basic types of synchronisation:

- Block-by-block synchronisation (default synchronisation),
- Cross-block synchronisation.

M functions not defined are indicated as unknown M functions by an error message after NC program start and decoding is aborted.

### 6.2 Management of technology functions

A basic distinction is made between two types of technology functions:

technology functions synchronised block-by-block and cross-block synchronised technology functions. This subdivision is also reflected on the High-Level Interface.

All technology functions to be synchronised block-by-block must be acknowledged at the latest at the end of the NC block in which they were programmed. They are therefore saved in consecutive sequence in the corresponding management field.

This is not the case with technology functions with cross-block synchronisation. In this case, the non-acknowledged technology functions yet to be executed may be distributed over the entire field (with gaps). In addition, it may occur that several identical technology functions are present in this field since, owing to cross-block synchronisation, the individual technology commands do not need to be acknowledged at the end of the NC block in which they were programmed. This must be allowed for at the PLC end to generate the acknowledgement of technology functions.

The number of technology functions programmed in an NC block is made available on the HLI for the array of the technology functions synchronised block-by-block. The number of non-acknowledged technology functions is specified for the array of cross-block synchronised technology functions.

## 6.3 Elements to manage axis-specific technology control units

### 6.3.1 Block-by-block synchronisation (default synchronisation)

Array of technology functions with block-by-block synchronisation	
Description	Array of M/H/S/T technology functions with block-by-block synchronisation. Technology functions are saved contiguously in this array.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.ATechnoUnitAxe_Std[tech_unit_idx]
Data type	ARRAY [1..HLI_MaxTechnoUnitsAxeStdSync] OF TechnoUnitAxe

Number of technology functions with block-by-block synchronisation	
Description	Number of valid entries in the array ATechnoUnitAxe_Std( = number of technology functions to be acknowledged in this block)
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.W_UsedUnitsStdSync
Data type	UINT
Value range	[0, HLI_MaxTechnoUnitsAxeStdSync]
Access	PLC reads

### 6.3.2 Cross-block synchronisation

Technology functions with cross-block synchronisation	
Description	Array of M/H/S/T technology functions with cross-block synchronisation. There may be entries of already acknowledged technology functions among the entries for non-acknowledged M functions.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.ATechnoUnitAxe_Late[tech_unit_idx]
Data type	ARRAY [1..HLI_MaxTechnoUnitsAxeLateSync] OF TechnoUnitAxe

Number of technology functions with cross-block synchronisation	
Description	Number of non-acknowledged technology functions in the array AtechnoUnitAxe_Late
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.W_UsedUnitsLateSync
Data type	UINT
Value range	[0, HLI_MaxTechnoUnitsAxeLateSync]
Access	PLC reads

## 6.4 Elements for managing channel-specific technology control units

### 6.4.1 Block-by-block synchronisation (default synchronisation)

Array of technology functions with block-by-block synchronisation	
Description	Array of M/H/S/T technology functions with block-by-block synchronisation. Technology functions are saved contiguously in this array.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.ATechnoUnitChannel_Std[tech_unit_idx]
Data type	ARRAY [1..HLI_MaxTechnoUnitsChStdSync] OF TechnoUnitChannel

Number of technology functions with block-by-block synchronisation	
Description	Number of entries in the array ATechnoUnitChannel_Std ( = number of technology functions to be acknowledged in this block)
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.W_UsedUnitsStdSync
Data type	UINT
Value range	[0, HLI_MaxTechnoUnitsChStdSync]
Access	PLC reads

### 6.4.2 Cross-block synchronisation

Technology functions with cross-block synchronisation	
Description	Array of M/H/S/T technology functions with cross-block synchronisation. There may be entries of already acknowledged technology functions among the entries for non-acknowledged M functions.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.ATechnoUnitChannel_Late[tech_unit_idx]
Data type	ARRAY [1.. HLI_MaxTechnoUnitsChLateSync] OF TechnoUnitChannel

Number of technology functions with cross-block synchronisation	
Description	Number of entries in the array ATechnoUnitChannel_Late.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.W_UsedUnitsLateSync
Data type	UINT
Value range	[0, HLI_MaxTechnoUnitsChLateSync]
Access	PLC reads

## 6.5 Data of a technology control unit

### 6.5.1 Data of an axis-specific technology control unit



Data of a technology function, axis			
Description	A technology control unit contains elements for commanding, acknowledging and transferring any required parameters.		
Data type	TechnoUnitAxe		
ST path	<b>Default synchronisation:</b> pAC[axis_idx]^addr^.ATechnoUnitAxe_Std[tech_unit_idx] <b>Cross-block synchronisation:</b> pAC[axis_idx]^addr^.ATechnoUnitAxe_Late[tech_unit_idx]		
Job			
ST element	<b>.X_Please</b>		
Description	By setting X_Please, the CNC signals to the PLC that the technology control unit is to be executed.		
Data type	BOOL		
Value range	[TRUE, FALSE]		
Special features	<b>Consumption data item</b>		
Access	The CNC refreshes the data of the technology function only if this element is FALSE. After refreshment, the CNC sets this element to TRUE after the element X_Done was set to FALSE.  The PLC reads the data of the technology function if this element has the value TRUE. After the data is transferred, the PLC sets the value to FALSE.		
Function type			
ST element	<b>.W_FktCtrl</b>		
Description	The technology function type is transferred in W_FktCtrl.		
Data type	UINT		
Value range	Value	Constant	Meaning
	1	GCW_250_HLIIntfMFkt	M function
	2	GCW_250_HLIIntfHFkt	H function
	3	GCW_250_HLIIntfSpindel	S function
Access	PLC reads		
Parameter			
ST element	<b>.MSTHProsessAxe_Attribut</b>		
Description	Depending on the contents of the element W_FktCtrl, this element contains the parameters of an  M function/H function if the technology function type is GCW_250_HLIIntfMFkt or GCW_250_HLIIntfHFkt S function (spindle) if the technology function type is GCW_250_HLIIntfSpindel		
Data type	ARRAY [1.. HLI_TechnoUnionByteCountAxe] OF BYTE		
Access	PLC reads		
Acknowledgement			

ST element	<b>.X_Done</b>
Description	By setting the element X_Done to TRUE, the PLC signals to the CNC that the technology information was executed.
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	<b>Consumption data item</b>
Access	PLC sets the value to TRUE when the technology function was executed. The CNC sets the value to a FALSE before a new command.

### 6.5.2 Data of a channel-specific technology control unit

Data of a technology function, channel			
Description	A technology control unit contains elements for commanding, acknowledging and transferring any required parameters.		
Data type	TechnoUnitChannel		
ST path	<b>Default synchronisation:</b> pMC[channel_idx]^addr^.ATechnoUnitChannel_Std[tech_unit_idx] <b>Cross-block synchronisation:</b> pMC[channel_idx]^addr^.ATechnoUnitChannel_Late[tech_unit_idx]		
Job			
ST element	<b>.X_Please</b>		
Description	By setting X_Please, the CNC signals to the PLC that the technology control unit is to be executed.		
Data type	BOOL		
Value range	[TRUE, FALSE]		
Special features	<b>Consumption data item</b>		
Access	The CNC refreshes the data of the technology function only if this element is FALSE. After refreshment, the CNC sets this element to TRUE after the element X_Done was set to FALSE.  The PLC reads the data of the technology function if this element has the value TRUE. After the data is transferred, the PLC sets the value to FALSE.		
Function type			
ST element	<b>.W_FktCtrl</b>		
Description	The technology function type is transferred in W_FktCtrl.		
Data type	UINT		
Value range	<b>Value</b>	<b>Constant</b>	<b>Function</b>
	1	HLI_INTF_M_FKT	M function
	2	HLI_INTF_H_FKT	H function
	3	HLI_INTF_SPINDEL	S function
	4	HLI_INTF_TOOL	T function
Access	PLC reads		
Parameter			
ST element	<b>.MSTHProsessChannel_Attribut</b>		
Description	Depending on the contents of the element W_FktCtrl, this element contains the parameters of an  M function/H function if the technology function type is GCW_250_HLIIntfMFkt or GCW_250_HLIIntfHFkt  S function (spindle) if the technology function type is GCW_250_HLIIntfSpindel  T function if the technology function type is GCW_250_HLIIntfTool		

Data type	ARRAY [1.. HLI_TechnoUnionByteCountChannel] OF BYTE
Access	PLC reads
Acknowledgement	
ST element	<b>.X_Done</b>
Description	By setting the element X_Done to TRUE, the PLC signals to the CNC that the technology information was executed.
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	<b>Consumption data item</b>
Access	PLC sets the value to TRUE when the technology function was executed. The CNC sets the value to a FALSE before a new command.

## **6.6 Data of the technology functions**

### **6.6.1 Data of the M/H function**

Data of the M function/H function	
Description	The function number and the execution time are transferred as additional parameters of an M or H function.
Data type	MHProzess
ST path	<p>pMHProcess : POINTER TO MHProzess;</p> <p><b>Channel-specific, default synchronisation:</b></p> <p><b>pMHProcess</b> := ADR(pMC[channel_idx]^addr^.ATechnoUnitChannel_Std[tech_unit_idx].MSTHProzessChannel_Attribut.AB_Data[1]);</p> <p><b>Channel-specific, cross-block synchronisation:</b></p> <p><b>pMHProcess</b> := ADR(pMC[channel_idx]^addr^.ATechnoUnitChannel_Late[tech_unit_idx].MSTHProzessChannel_Attribut.AB_Data[1]);</p>
Access	PLC reads
Number of the M function/H function	
Description	<p>Number of the M or H function. This corresponds to the number for an M or H function programmed in the NC program.</p> <p>Example: For instance, if M4711 was programmed.</p>
ST element	<b>.D_Number</b>
Data type	UDINT
Access	<p>FktNr : UDINT;</p> <p>FktNr := pMHProcess^.D_Number</p>
Probable execution time M/H function	
Description	<p>Contains values configured for the channel parameters in the entry P-CHAN-00040 or P-CHAN-00026 (m_prozess_zeit[]). These values can be used to implement timeout monitoring for technology functions in the PLC application.</p> <p>or</p> <p>If synchronisation type MOS_TS is configured, the sample time offset of the M/H technology function is indicated in this element. It is calculated and output by the NC kernel.</p>
ST element	<b>.D_Time</b>
Data type	UDINT
Access	<p>Time : UDINT;</p> <p>Time := pMHProcess^.D_Time</p>
Special features	<p>Re. 1.): For spindle-specific M functions the corresponding channel parameter is called mX_prozess_zeit where [X = 3, 4, 5, 19].</p> <p>Example: The parameter for the M3 function of a spindle is:</p> <p>spindel[index].m3_prozess_zeit. Bsp: The characteristic parameter for an M function is:</p> <p>spindel[index].mX_prozess_zeit.</p>
Block number of the M/H function	
Description	Block number from the NC program to specify the program line in the NC program in which the M or H function was programmed.
ST element	<b>.D_BlockNumber</b>

Data type	UDINT
Access	BlockNr : UDINT; BlockNr := pMHProcess^.D_BlockNumber
<b>Program line number of the M/H function</b>	
Description	Number of the NC program line in which the M/H function was programmed.
ST element	.D_PrgLineNumber
Data type	UDINT
Access	PrgLineNr : UDINT; PrgLineNr := pMHProcess^.D_PrgLineNumber
<b>Number as additional information</b>	
Description	A number assigned to an M or H function in the NC program and transferred when the M/H function is output to the interface. See [PROG//M/H functions with additional information].
ST element	.D_AddNumber
Data type	DINT
Access	AddNr : DINT; AddNr := pMHProcess^.D_AddNumber
<b>Number of cross-block synchronised technology functions not acknowledged by the PLC</b>	
Description	Number of technology functions not yet acknowledged by the PLC but with cross-block synchronisation and waiting at the interface. The number contains all types of technology functions.
ST element	.W_NrLateSync
Data type	UINT
Access	NrLateSync : UINT; NrLateSync := pMHProcess^.W_NrLateSync

## 6.6.2 Data of the S function

When M functions (M03, M04, M05, M19) associated with a spindle are programmed, the technology function data is saved as an S function in the associated **axis-specific** HLI area.

Data of the M function/H function	
Description	The parameters of an S function are contained in the SProzess structure.
Data type	Sprozess
ST path	<p>pSProcess : POINTER TO SProzess;</p> <p><b>Axis-specific, default synchronisation:</b>  <b>pSProcess</b> := ADR(pAC[axis_idx]^addr^.ATechnoUnitAxe_Std[tech_unit_idx].MSTH-ProzessAxe_Attribut.AB_Data[1]);</p> <p><b>Axis-specific, cross-block synchronisation:</b>  <b>pSProcess</b> := ADR(pAC[axis_idx]^addr^.ATechnoUnitAxe_Late[tech_unit_idx].MSTH-ProzessAxe_Attribut.AB_Data[1]);</p>
Access	PLC reads
Command position of spindle with M19	
Description	Command position for spindle position with M19
ST element	<b>.D_Pos</b>
Data type	DINT
Unit	10 <sup>-4</sup> °
Access	ActivePosition : DINT; ActivePosition := pSProcess^.D_Pos;pMHProcessData^.D_Pos
Programmed spindle speed	
Description	Programmed spindle speed
ST element	<b>.D_Rev</b>
Data type	DINT
Unit	10 <sup>-3</sup> °/s
Access	PrgRevolution: DINT; PrgRevolution: = pSProcess^.D_Rev;
Probable execution time	
Description	Time probably required to process an S function.
ST element	<b>.D_Zeit</b>
Data type	UDINT
Unit	1 µs
Access	ExpectedTime : UDINT; ExpectedTime:= pSProcess^.D_Zeit;
Number of the M function of the spindle indexing function	
Description	Number of the spindle indexing function (M03, M04, M05)
ST element	<b>.W_MoveCmd</b>
Data type	UINT



Value range	<b>Value</b>	<b>M function</b>
	3	M03
	4	M04
	5	M05
Access	MoveCmdNum : UINT; MoveCmdNum := pSProcess^. <b>W_MoveCmd</b> ;	
<b>Number of the M function of the spindle positioning function</b>		
Description	Number of the spindle positioning function (M19)	
ST element	<b>.W_PosCmd</b>	
Data type	UINT	
Value range	19 stands for M19	
Access	PosCmdNum : UINT; PosCmdNum := pSProcess^. <b>W_PosCmd</b> ;	
<b>Axis number</b>		
Description	Unique system-wide number of a logical axis/spindle	
ST element	<b>.W_log_AxeNr</b>	
Data type	UINT	
Value range	TwinCAT PLC typically [1, gNrAx]	
Access	LogAxisNum : UINT; LogAxisNum := pSProcess^. <b>W_Log_AxeNr</b> ;	
<b>Number of cross-block synchronised technology functions not acknowledged by the PLC</b>		
Description	Number of technology functions not yet acknowledged by the PLC but with cross-block synchronisation and waiting at the interface. The number contains technology functions of all type.	
ST element	<b>.W_NrLateSync</b>	
Data type	UINT	
Access	NrLateSync : UINT; NrLateSync := pSProcess^. <b>W_NrLateSync</b>	

### 6.6.3 Data of the T function

Data of the T function	
Description	All data for a tool change is combined in a T function.
Data type	TProzess
ST path	<p>pTProzess : POINTER TO TProzess;</p> <p><b>Channel-specific, default synchronisation:</b></p> <p><b>pTProzess</b> := ADR(pMC[channel_idx]^addr^.ATechnoUnitChannel_Std[tech_unit_idx].MSTHProcessChannel_Attribut.AB_Data[1]);</p> <p><b>Channel-specific, cross-block synchronisation:</b></p> <p><b>pTProzess</b> := ADR(pMC[channel_idx]^addr^.ATechnoUnitChannel_Late[tech_unit_idx].MSTHProsessChannel_Attribut.AB_Data[1]);</p>
Access	PLC reads
Tool identification	
Description	The structure contains the identification number of the tool. There also may be identification numbers for identical or similar tools. A description of the structure format is contained in User data of tool identification [► 146]
ST element	<b>.HLIToolID_Data</b>
Data type	HLIToolId [► 145]
Access	<p>HLIToolId : HLIToolId</p> <p>HLIToolId := pTProzess^.HLIToolID_Data;</p>
Information on the tool unit	
Description	Information supplied to the PLC referring to a tool unit.
ST element	<b>.D_AddInfo[ ]</b>
Data type	ARRAY [1..HLI_MAX_NBR_ADD_INFO] OF HLI_UN32
Access	<p>AddInfo: UDINT;</p> <p>AddInfo := pTProzess^.D_AddInfo[X] where X = [1, HLI_MAX_NBR_ADD_INFO]</p>
Number of cross-block synchronised technology functions not acknowledged by the PLC	
Description	Number of technology functions not yet acknowledged by the PLC but with cross-block synchronisation and waiting at the interface. The number contains technology functions of all type.
ST element	<b>.W_NrLateSync</b>
Data type	UINT
Access	<p>NrLateSync : UINT;</p> <p>NrLateSync := pTProzess^.W_NrLateSync</p>

### 6.6.3.1 User data of tool identification

With the definition of pTProzess [► 145] from the previous section and the definition

pHLIToolId: POINTER TO HLI\_TOOL\_ID applies:

**pHLIToolId** := ADR(pTProzess^.id);

and is used as shown in the table below.

Number of the tool to be replaced	
Description	Number of the tool to be replaced
ST element	<b>.D_Basic</b>
Data type	DINT
Access	ToChangeToolNum : DINT; ToChangeToolNum := pHLIToolId^.D_Basic;
Number of a sister tool	
Description	Number of an identical sister tool
ST element	<b>.D_Sister</b>
Data type	DINT
Access	SisterToolNum : DINT; SisterToolNum := pHLIToolId^.D_Sister;
Number of a variant tool	
Description	Number of a similar variant tool
ST element	<b>.D_Variant</b>
Data type	DINT
Access	VariantToolNum : DINT; VariantToolNum := pHLIToolId^.D_Variant;
Sister tool valid	
Description	Validity identifier for the sister tool.
ST element	<b>.X_SisterValid</b>
Data type	BOOL
Access	SisterToolValid : BOOL; SisterToolValid := pHLIToolId^.X_SisterValid;
Variant tool valid	
Description	Validity identifier for the variant tool.
ST element	<b>.X_VariantValid</b>
Data type	BOOL
Access	VariantToolValid : BOOL; SisterToolValid := pHLIToolId^.X_VariantValid;

## 7 External variables / V.E variables

Data can be exchanged between the NC program and the PLC via the HLI using external variables. Each channel has a separate data area for external variables known only in the channel. In addition, there is a cross-channel global data area which all channels can access.

At the PLC end, the data areas for the external variables are represented as an ARRAY OF UDINT. The index of the individual array elements starts with the value 1.

Each external variable always occupies one memory block of HLI\_VEByteCount (24) bytes, irrespective of its data type. If an array of external variables was defined, the individual variables are saved packed in the memory area (several variables per memory block). Here, several consecutive memory blocks may be used depending on the array size.

When the PLC accesses external variables, the index of the variables in the memory area of the external variables must first be determined:

### Example for calculating an index:

4th variable (VarNr = 4):

Offset = (VarNr - 1) \* HLI\_VEByteCount / 4 + 1

A variable with the index = 3 results in an offset in the memory of 13.

The memory area must then be accessed depending on the actual data type of the external variables. All information required for this is contained in the configuration list of the external variables.

Please refer to the documentation [EXTV] for further details on external variables.

If the NC kernel runs in the TwinCAT runtime environment, the number of external variables can be configured in the System Manager.

An example program to access external variables is available under the name of HLI-Ve1.pro.

External variable	
Description	Memory area for data exchange between NC program and PLC
Data type	VeData
ST Path	<p>PVeData : POINTER TO VEData;</p> <p><b>CNC global:</b> pVeData := ADR(pVeGlob^.addr);</p> <p><b>channel-specific:</b> pVeData := ADR(pVe[channelIdx]^ .addr);</p>
Access	<p>PLRealVal : POINTER TO LREAL; LRealVal : LREAL; VeOffset : DINT;</p> <p>VeOffset := (VarNr - 1) * HLI_VEByteCount / 4 + 1; pLRealVal := ADR(pVeData^.AHLI_UN32_Data[VeOffset]); LRealVal := pLRealVal^;</p>
Peculiarities	Access depends on the data type of the external variables.

## 8 Processing types

The CNC distinguishes between 5 operation modes. Switchover between these operation modes is performed using the operator interface and/or PLC interface. However, only **one operation mode may be active at any one time**.

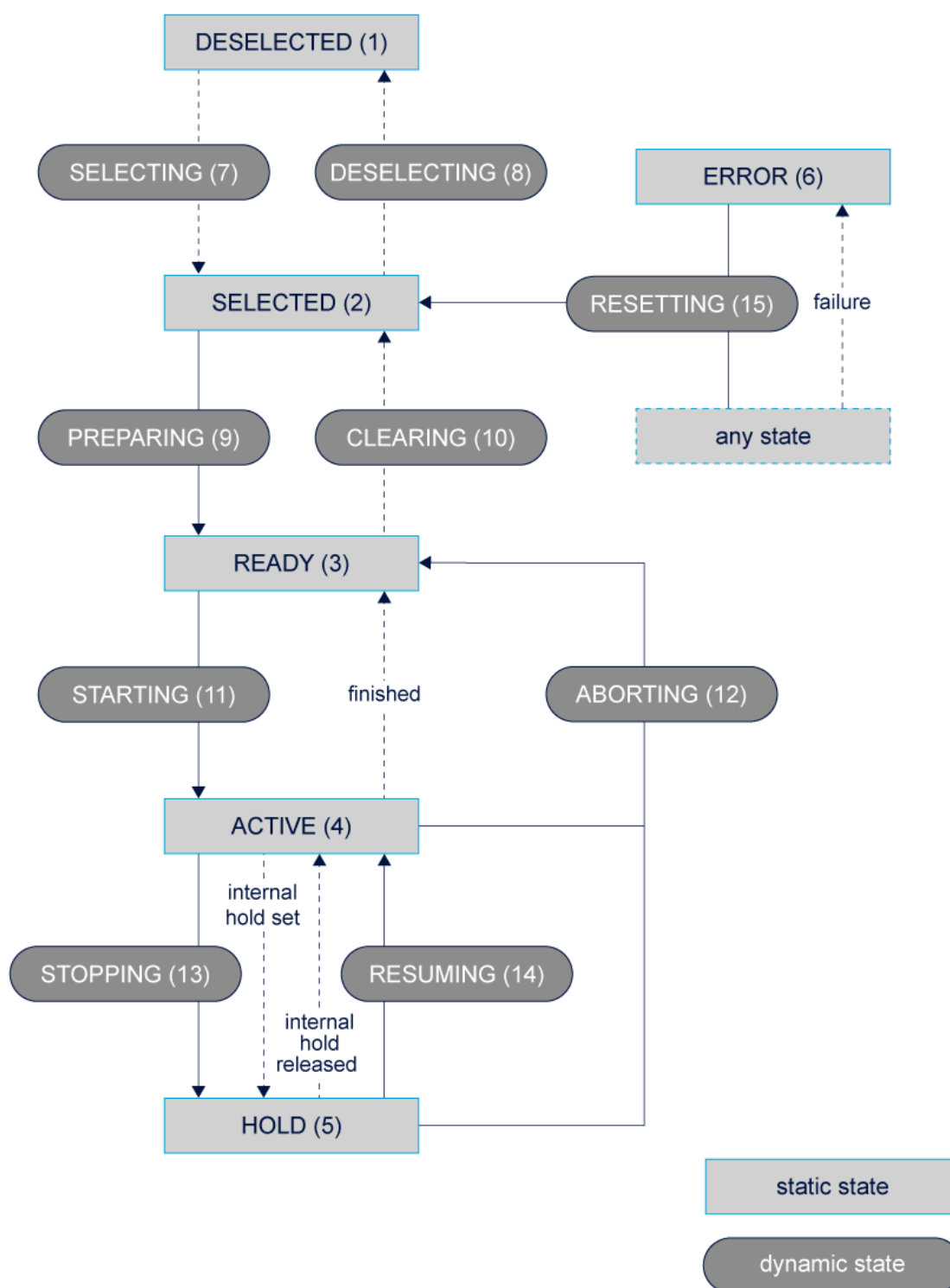
The following operation modes are defined:

Operation mode	ST constant	Value	Explanation
Standby	HLI_IMCM_STANDBY_MODE	1	No operation mode is selected. Default after controller start-up.
Automatic mode	HLI_IMCM_AUTOMATIC_MODE	2	The controller can process a complete NC program automatically. Program execution can be interrupted and resumed.
Manual block	HLI_IMCM_MDI_MODE	3	Motions are commanded by the operator computer via a single NC block. The NC block is transferred to the controller in a string and executed via a START command. A motion can be interrupted and resumed.
Manual operation mode	HLI_IMCM_MANUAL_MODE	4	Motions are commanded by peripherals connected directly to the controller (buttons, handwheels).
Homing	HLI_IMCM_REFERENCE_MODE	5	Axes can be referenced. An NC program with the name rpf.nc then starts.

An operation mode may have different states. The states of individual operation modes and their significance for the operation mode are discussed in the sections below.

### 8.1 State graph of operation modes

Note: There is no state graph for the “Standby” operation mode.



**Fig. 19: State graph of an operation mode**



## Notice

A CNC reset is automatically executed when the target operation mode and target state are commanded if the state change specifies an abort (see “abort”).

The error state is exited by an automatically executed reset if an operation mode was commanded.

In addition, a CNC reset can be forced by the following commands:

- Select operation mode = STANDBY and state = SELECTED
- Explicit specification of transition = RESET in the currently active operation mode
- Specify the target state = RESETTNG (see below)

### 8.1.1 States of operation mode: Automatic mode

State	ST constant	Value	Description
DESELECTED	GCW_250_HLIImcmProcessDeselected	1	Operation mode is deselected
SELECTED	GCW_250_HLIImcmProcessSelected	2	Automatic mode is selected
READY	GCW_250_HLIImcmProcessReady	3	NC program is selected
ACTIVE	GCW_250_HLIImcmProcessActive	4	NC program is running
HOLD	GCW_250_HLIImcmProcessHold	5	NC program is interrupted (see also Feedhold).
ERROR	GCW_250_HLIImcmProcessError	6	An error occurred while the NC program is executed.

States of Automatic mode

The NC program name must be transferred where there is a transition from selected to ready.

### 8.1.2 States of operation mode: Manual block

State	ST constant	Value	Description
DESELECTED	GCW_250_HLIImcmProcessDeselected	1	Manual block mode is deselected.
SELECTED	GCW_250_HLIImcmProcessSelected	2	Manual block mode is selected. An NC block (NC blocks) can be programmed.
READY	GCW_250_HLIImcmProcessReady	3	??? MDI block(s) is/are selected. ??? CNC accepted the programmed NC block(s).
ACTIVE	GCW_250_HLIImcmProcessActive	4	The NC block(s) is/are running.
HOLD	GCW_250_HLIImcmProcessHold	5	NC block(s) is/are stopped (see also feedhold). Feedhold).
ERROR	GCW_250_HLIImcmProcessError	6	Error state

## States of Manual block mode

The manual block (string) must be transferred when there is a transition from SELECTED to READY.

### 8.1.3 States of operation mode: Manual operation mode

State	ST constant	Value	Description
DESELECTED	GCW_250_HLIImcmProcessDeselected	1	Operation mode is deselected.
SELECTED	GCW_250_HLIImcmProcessSelected	2	Operation mode is selected. (Basic state).
READY	GCW_250_HLIImcmProcessReady	3	Manual mode is programmed.
ACTIVE	GCW_250_HLIImcmProcessActive	4	Manual mode is running.
HOLD	GCW_250_HLIImcmProcessHold	5	Manual mode is stopped (see also feedhold). Feedhold).
ERROR	GCW_250_HLIImcmProcessError	6	Error state

## States of Manual mode

### 8.1.4 States of operation mode: Homing

State	ST constant	Value	Description
DESELECTED	GCW_250_HLIImcmProcessDeselected	1	Operation mode is deselected.
SELECTED	GCW_250_HLIImcmProcessSelected	2	Operation mode is selected. (Basic state).
READY	GCW_250_HLIImcmProcessReady	3	Homing is programmed (e.g. with respect to sequence).
ACTIVE	GCW_250_HLIImcmProcessActive	4	Homing is running.
HOLD	GCW_250_HLIImcmProcessHold	5	Homing is stopped (see also feedhold). Feedhold).
ERROR	GCW_250_HLIImcmProcessError	6	Error state

## States of Homing mode

Axes to be referenced must be transferred during the transition from SELECTED to READY. If no string is transferred, the default homing program rpf.nc is started.

## 8.2 Control commands/status information for operation modes

The control unit described below contains data to command a change of operation mode and to poll the current state of operation mode management.



## 8.2.1 Control Unit

Operation mode	
Description	Control unit to switch over the operation mode and poll the current state of operation mode management, including flow control of user data.
Data type	MCCControlMCMMode_State_Unit, see description of control unit with usage check
ST path	pMC[channel_idx]^addr^.MCCControlChannel_Data.MCCControlMCMMode_State_Unit
Commanded, requested data	
ST element	<b>.HLIProcTransTo_Mode_StateCommand</b> <b>.HLIProcTransTo_Mode_StateRequest</b>
Data type	HLIProcTransTo_Mode_State [► 154]
Access	PLC writes Command and reads Request
Return data	
ST element	<b>.HLI_ElmCmProcState_ModeStateRequest</b>
Data type	HLI_ElmCmProcState_ModeState [► 158]
Access	PLC reads
Flow control of commanded value	
ST element	<b>.X_McmCommandSemaphor</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	<b>Consumption data item</b>
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data.  PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	<b>.X_McmRequestSemaphor</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	<b>Consumption data item</b>
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE.  PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST path	<b>.X_McmEnable</b>

## 8.2.2 User data

### 8.2.2.1 Requested and commanded user data

Initial operation mode		
Description	Operation mode from which the operation mode is to be changed.	
ST path	<b>Commanded value</b> pMC[channel_idx]^^.addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand <b>Requested value</b> pMC[channel_idx]^^.addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateRequest	
Commanded, requested value		
ST element	<b>.X_McmCommandFromMode</b>	
Data type	UDINT	
Value range	<b>Value</b>	<b>Constant</b>
	1	GCW_250_HLIImcmStandbyMode
	2	GCW_250_HLIImcmAutomaticMode
	3	GCW_250_HLIImcmMDIMode
	4	GCW_250_HLIImcmManualMode
	5	GCW_250_HLIImcmReferenceMode
Special features	This element does not need to be programmed when the operation mode is changed. However, if a value is specified, a check is conducted when the operation mode is changed to establish whether the CNC is actually in the specified operation mode. A warning is issued if this is not the case.	

Initial state of operation mode when the operation mode is changed		
Description	State within the operation mode from which the state switchover is to occur.	
ST path	<b>Commanded value</b> pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand <b>Requested value</b> pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateRequest. <b>X_McmCommandFromState</b>	
Commanded, requested value		
ST element	<b>.X_McmCommandFromState</b>	
Data type	UDINT	
Value range	<b>Value</b>	<b>Significance / constant</b>
	0	no significance
	1	GCW_250_HLIImcmProcessDeselected
	2	GCW_250_HLIImcmProcessSelected
	3	GCW_250_HLIImcmProcessReady
	4	GCW_250_HLIImcmProcessActive
	5	GCW_250_HLIImcmProcessHold
	6	GCW_250_HLIImcmProcessError
Special features	This element does not need to be programmed when the operation mode is changed. However, if a value is specified, a check is conducted when the operation mode is changed to establish whether the CNC is actually in the specified operation mode. A warning is issued if this is not the case.	

Target operation mode when the operation mode is switched over	
Description	Operation mode to which the system is to switch.
ST path	<b>Commanded value</b> pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand.  <b>Requested value</b> pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateRequest.
Commanded, requested value	
ST element	<b>.X_McmCommandToMode</b>
Data type	UDINT
Value range	See "Operation mode actual value on of operation mode change" → Value range

Target state when operation mode is changed	
Description	Target state within the target operation mode.
ST path	<b>Commanded value</b> pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand  <b>Requested value</b> pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateRequest
Commanded, requested value	
ST element	<b>.X_McmCommandToState</b>
Data type	UDINT
Value range	See "State actual value of operation mode at operation mode change" → value range



### Release Note

**If the target state = RESETTING = 15 is specified, a CNC reset is executed explicitly.**

This function is available as of the following Build:

V2.10.1033.01 and higher

V2.10.1507.02 and higher

V2.10.1800.04 and higher

Parameters for operation mode change	
Description	Parameters for operation mode change. It may be necessary to specify parameters when commanding an operation mode change to ensure the successful change to a specific state of an operation mode. These parameters are saved in this element.
ST Path	pParameter: POINTER TO STRING; Parameter : STRING;  <b>Commanded value</b> pParameter := ADR(pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand.AB_Data[1]); <b>Requested value</b> pParameter := pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateRequest.AB_Data[1]  Parameter := pParameter^;
Data type	ARRAY[1..GCW_250_HLIImCmProcModeStateParaSize] OF BYTE
Peculiarities	The length of the character string is limited to maximum 83 characters for AUTOMATIC operation mode. If the character string is longer, it is recommended to use the parameterisation of file paths. See P-CHAN-00401to P-CHAN-00404.

If an operation mode change is commanded, it may be necessary to specify a parameter so that commanding can be executed successfully. The table below lists the cases where this is necessary and what type of parameter needs to be transferred. If one of the listed state transitions is involved in the operation mode and state change, the corresponding parameter must be specified. Consider the "State graph of other operation modes" to determine whether one of the state transitions listed below is executed in case of a commanded operation mode change.

## Parameters for operation mode change

The content of the "parameter" structure element is only adopted by the NC kernel if a state transition is commanded from SELECTED to one of the states READY, ACTIVE or HOLD.

Operation mode setpoint	Parameter
Automatic mode	The NC program name as character string.
Manual block	NC block (blocks)
Manual mode	no parameter → all axes are activated (G200) explicit activation of specific axes with G200[Axis_1, ...]
Homing	no parameter → NC program <b>rpf.nc</b> is started explicit selection of axes by manual block (e.g.: <b>G74 X1 Z2</b> )

Channel number	
Description	Number of the channel whose operation mode is to be switched over.
ST path	<b>Commanded value</b> pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand.X_McmCommandChannelNo <b>Requested value</b> pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Request.X_McmCommandChannelNo
Data type	UDINT
Special features	<b>Not used (only for compatibility with the HÜMNOS standard).</b>

### 8.2.2.2 Status information

Operation mode actual value	
Description	Actual value of the operation mode.
ST path	pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLI_ElmCmProcState_ModeStateRequest.X_McmRequestReadMode
Data type	UDINT
Value range	See “Operation mode actual value on of operation mode change” → Value range

State actual value of the operation mode.	
Description	Actual value of the operation mode state.
ST path	pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLI_ElmCmProcState_ModeStateRequest.X_McmRequestReadState
Data type	UDINT
Value range	See “State actual value of operation mode at operation mode change” → value range

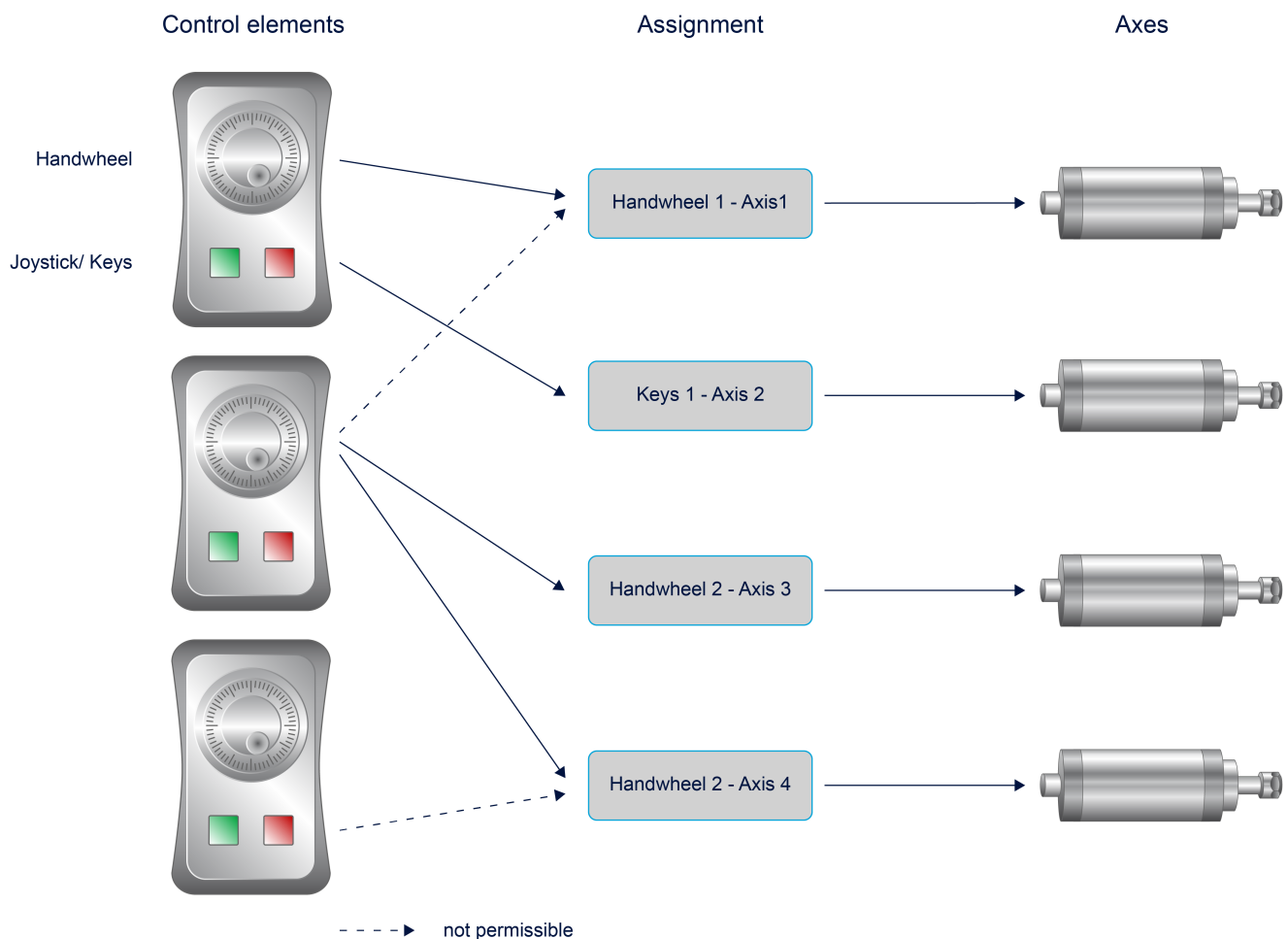
## 9 Manual operation mode

The manual mode function permits the external control of individual axes with physical elements of manual mode (handwheel, continuous jog buttons) via the HLI.

The following three options are available for axis motion:

- **Handwheel function:** : any path at any velocity by specifying handwheel increments.
- **Cont. jog mode:** any path at defined velocity when a button is operated.
- **Incr. jog mode:** defined path at defined velocity when a button is operated.

It is possible to assign the manual mode elements to one single or multiple axis and to change the parametrisation (e.g. the incremental jogging distance) during operation. The control element is assigned to the logical axes via the logical axis number. The diagram below shows as an example of an assignment of manual mode elements to CNC axes.



**Fig. 20: Control elements and assignment**

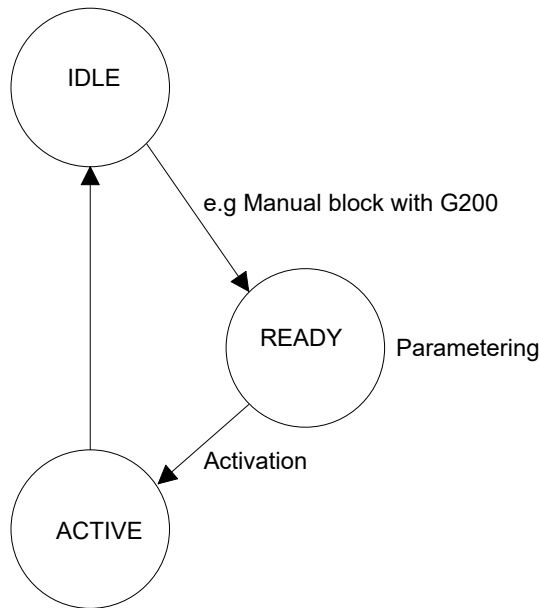
In manual mode, each axis can assume three different states:

- IDLE
- READY
- ENABLED

The following steps are required to use manual mode for an axis:

- Activating the READY state for axes

- Parametrising the manual mode type
- Activating manual mode (transition to ACTIVE state)
- Transferring control element actions (button presses, handwheel counter state) to the CNC.



**Fig. 21: Manual mode state transitions**

#### Transition to READY state

The transition to READY state can be executed by the following actions:

- Explicit switchover of operation mode via HLI or GUI
- Programming of G200/G201 in the NC program or manual block

The desired manual mode type can then be parameterised in the READY state. See also the section Parameterising manual mode

#### Transition to READY state

The axis transits to the ACTIVE state as soon as a control element was assigned. Details are contained in the section Activating control elements.

#### Transferring control element actions

In the ACTIVE state, control element actions can now be transferred to the CNC to move the axis.

#### Deselecting manual mode

The ACTIVE state of an axis is deselected when the axis is assigned to the control element 0 or If a reset was executed.

Information on the axis state regarding manual mode is contained in the structure HLI\_HB\_AXIS\_DISPLAY\_DATA (see Section Status information of manual mode).

## 9.1 Status information of manual mode



State of manual mode		
Description	The manual operation mode is in one of the states described below.	
Signal flow	CNC → PLC	
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. <b>W_State</b>	
Data type	UINT	
Value range	<b>Value</b>	<b>Meaning</b>
	0	Operation mode deactivated
	1	Operation mode enabled in NC program via G200, G201 but not assigned to a control element.
	2	Operation mode enabled in NC program via G200, G201, assigned to control element.
Access	PLC reads	

Operation mode of manual mode		
Description	Look ahead waits to receive request from an axis.	
Signal flow	CNC → PLC	
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. <b>W_OperationMode</b>	
Data type	UINT	
Value range	<b>Value</b>	<b>Operation mode</b>
	0	no operation mode selected
	1	Handwheel mode
	2	Continuous jog mode
	3	jog mode
Access	PLC reads	

Control element number	
Description	Logical number of the control element currently linked to the axis in question.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. <b>W_ControlElement</b>
Data type	UINT
Access	PLC reads

Handwheel resolution	
Description	Resolution of the handwheel which is linked as control element to the axis in question.
Signal flow	CNC → PLC
Unit	Increments per handwheel pulse
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. <b>D_HRAufloesung</b>
Data type	DINT
Access	PLC reads

Path velocity in continuous jog mode	
Description	Path velocity of the axis in question when moved in continuous jog mode.
Signal flow	CNC → PLC
Unit	mm/min
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. <b>D_TippGeschw</b>
Data type	DINT
Access	PLC reads

Path velocity in incremental jog mode	
Description	Path velocity of the axis in question when moved in continuous jog mode.
Signal flow	CNC → PLC
Unit	mm/min
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. <b>D_JogGeschw</b>
Data type	DINT
Access	PLC reads

Motion path in incremental jog mode	
Description	Motion path of the axis in question for each button press when moved in incremental jog mode.
Signal flow	CNC → PLC
Unit	Mm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data.D_JogWeg
Data type	DINT
Access	PLC reads

## 9.2 Control commands of manual mode

The individual manual operation modes are controlled by specific control units. These control units contain data on the flow control of user data and the user data itself. User data generally comprises elements of a structure.

### 9.2.1 Activating control elements for manual mode

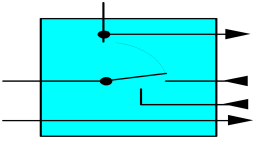
After an axis was prepared for manual mode by G200/G201, this axis can be assigned a logical control element (button/handwheel) by a parameter set. This takes place during activation of an axis.

The default value settings are defined in the axis-specific parameter lists:

```
# 1 handwheel, 2 continuous, 3 incremental
handbetrieb.default.operation_mode 2 # 2 continuous,
handbetrieb.default.control_element 1 # logical handwheel/key
```

The 0th parameter set (index = 0) of the manual mode parameters is used as default parameter.

Every time the manual mode (see Operation mode types) or G200/G201 is reselected, the last known setting of the axes (operation mode, link to control element and parameter set) is restored.

Activating control elements with manual mode	
Description	Control unit to manage data to activate a control element and assign it to an axis in manual mode, including flow control of user data.
 Data type	HBAutomationControlUnit, see description Control unit with usage check
Special features	<p>This control unit can only be used if the CNC is in manual mode or if the axes are enabled for manual mode by an explicit G200/G201.</p> <p>Otherwise, activation is refused with an error message (e.g. 150048 -&gt; "Impermissible selection of operation mode in continuous jog mode").</p>
ST path	pMC[channel_idx]^addr^.MCControlHB_Data.HBAutomationControlUnit_Activation
Access	PLC reads <b>HBAutomation_Request</b> and writes <b>HBAutomation_Command + X_Enable</b>
Commanded, requested data	
ST element	<b>.HBAutomation_Command</b> <b>.HBAutomation_Request</b>
Data type	HBAutomation [► 166], for description see Section User data on activation [► 165]
Flow control commanded data	
ST element	<b>.X_CommandSemaphor</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	<b>Consumption data item</b>
Access	<p>CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data.</p> <p>PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.</p>
Flow control of requested data	
ST element	<b>.X_RequestSemaphor</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	<b>Consumption data item</b>
Access	<p>CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE.</p> <p>PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.</p>
Redirection	
ST path	<b>.X_Enable</b>

### 9.2.1.1 User data on activation

Axis number	
Description	<p>Unique system-wide number of a logical axis.</p> <p>An control element is assigned to the specified logical axis with which the axis is to be moved in manual mode.</p>
ST path	<p><b>Commanded value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Command.<b>W_LogAchsNr</b></p> <p><b>Requested value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Request.<b>W_LogAchsNr</b></p>
Data type	UINT
Value range	TwinCAT PLC typically [1, gNrAx]

Control element number	
Description	Number of the logical control element to be assigned to the logical axis.
ST path	<p><b>Commanded value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Command.<b>W_ControlElement</b></p> <p><b>Requested value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Request.<b>W_ControlElement</b></p>
Data type	UINT
Value range	<p><b>When continuous and incremental jog mode is activated:</b></p> <p>one of the values defined as logical button pair numbers in the configuration list hand_mds.lis for the characteristics tasten_data[X].log_tasten_nr.</p> <p><b>When handwheel mode is activated:</b></p> <p>one of the values defined as logical handwheel numbers in the configuration list hand_mds.lis for the characteristics hr_data[0].log_hr_nr.</p>
Special features	If 0 is specified as the logical number, the current operation mode of an axis is deselected.

Manual operation mode		
Description	Manual operation mode to be assigned to the logical axis.	
ST path	<b>Commanded value</b> pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Command. <b>W_OperationMode</b> <b>Requested value</b> pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Request. <b>W_OperationMode</b>	
Data type	UINT	
Value range	<b>Value</b>	<b>Operation mode</b>
	0	no operation mode, current operation mode selected
	1	Handwheel mode
	2	Continuous jog mode
	3	jog mode

Manual operation mode parameter set		
Description	Specifies the index of the parameter set to be used for the manual mode.	
ST path	<b>Commanded value</b> pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Command. <b>W_ParameterIndex</b> <b>Requested value</b> pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Request. <b>W_ParameterIndex</b>	
Data type	UINT	
Value range	[0; 2]	
Special features	The first value set in the parameter table (index = 0) is overwritten by the PLC interface when individual parameters are specified. The remaining parameter sets are not changed. This means, they correspond to the values specified in the axis-specific parameter lists.	

#### Remark:

The currently active axis can be deactivated by deselecting the operation mode or by assigning the control element 0.

## 9.2.2 Parametrising of manual mode

Basically, every manual mode can be specifically parametrised. For example, the motion velocities or incremental step can be defined per axis. The default values here are pre-assigned in the axis-specific parameter lists.

handbetrieb.hr.auf1[1]	20
handbetrieb.hr.auf1[2]	30
#	
handbetrieb.tipp.geschw[0]	30000
handbetrieb.tipp.geschw[1]	30000
handbetrieb.tipp.geschw[2]	40000
handbetrieb.tipp.vb_eilgang	40000
#	
handbetrieb.jog.weg[0]	1000
handbetrieb.jog.weg[1]	2000
handbetrieb.jog.weg[2]	3000
handbetrieb.jog.geschw[0]	30000
handbetrieb.jog.geschw[1]	30000
handbetrieb.jog.geschw[2]	60000

In addition, an individual parameter value can be pre-assigned via the PLC interface. This parameter value is the first element (index 0) stored in the table of default parameter. It can be selected when an axis is activated.

Parameters can be changed at any time but they only become effective when an axis is activated (see above). When an axis is activated, the number (index) of the desired parameter set is specified in addition to the operation mode and the control element.

### 9.2.2.1 Continuous jog mode (by button press)

#### 9.2.2.1.1 Control Unit

Parameterising continuous jog mode in manual mode	
Description	Control unit to manage data for parameterising continuous jog mode in manual mode, including flow control of user data.
Data type	HBTipParameterControlUnit, see description Control unit with usage check
Special features	The manual mode parameter can be written at any time and it is stored internally in a table under the index 0. This value is only becomes effective in the corresponding operation mode when the axis is activated.
Access	PLC reads <b>HBTipParameter_Request</b> and writes <b>HBTipParameter_Command + X_Enable</b>
ST path	pMC[channel_idx]^^.addr^.MCControlHB_Data.HBTipParameterControlUnit_TipParameter
Commanded, requested data	
ST element	<b>.HBTipParameter_Command</b> <b>.HBTipParameter_Request</b>
Data type	HBTipParameter [► 168], for description see Section User data [► 168]
Flow control commanded data	
ST element	<b>.X_CommandSemaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data.  PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	<b>.X_RequestSemaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE.  PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST path	<b>.X_Enable</b>

### 9.2.2.1.2 User data



Axis number	
Description	Unique system-wide number of a logical axis. The specified logical axis is assigned the velocity at which it will be moved in manual mode in continuous jog mode.
ST path	<b>Commanded value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBTipParameterControlUnit_TipParameter.HBTipParameter_Command.W_LogAchsNr</code> <b>Requested value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBTipParameterControlUnit_TipParameter.HBTipParameter_Request.W_LogAchsNr</code>
Data type	UINT
Value range	TwinCAT PLC typically [1, gNrAx]

Continuous joy velocity	
Description	Velocity assigned to the logical axis in continuous jog mode.
ST path	<b>Commanded value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBTipParameterControlUnit_TipParameter.HBTipParameter_Command.D_Speed</code> <b>Requested value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBTipParameterControlUnit_TipParameter.HBTipParameter_Request.D_Speed</code>
Unit	1 µm/s
Data type	UDINT

## 9.2.2.2 Incremental jog mode (by button press)

### 9.2.2.2.1 Control Unit

Programming incremental jog mode in manual mode	
Description	Control unit to manage data to parameterise incremental jog mode in manual mode, including flow control of user data.
Data type	HBJogParameterControlUnit, see description Control unit with usage check
Special features	The manual mode parameter can be written at any time and it is stored internally in a table under the index 0. This value is only becomes effective in the corresponding operation mode when the axis is activated.
ST path	PMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter
Access	PLC reads <b>HBJogParameter_Request</b> and writes <b>HBJogParameter_Command + X_Enable</b>
Commanded, requested data	
ST element	<b>.HBJogParameter_Command</b> <b>.HBJogParameter_Request</b>
Data type	HBJogParameter [► 170], for description see Section User data [► 170]
Flow control commanded data	
ST element	HBJogParameterControlUnit_JogParameter. <b>X_CommandSemaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data.  PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	<b>.X_RequestSemaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE.  PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST element	<b>.X_Enable</b>

#### 9.2.2.2.2 User data

Axis number	
Description	<p>Unique system-wide number of a logical axis.</p> <p>The specified logical axis is assigned the velocity and incremental step width for each button press to move the axis in manual mode in incremental jog mode.</p>
ST path	<p><b>Commanded value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Command.<b>W_LogAchsNr</b></p> <p><b>Requested value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Request.<b>W_LogAchsNr</b></p>
Data type	UINT
Value range	TwinCAT PLC typically [1, gNrAx]

Incremental job velocity	
Description	Velocity to be assigned to the logical axis in incremental jog mode.
ST path	<p><b>Commanded value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Command.<b>D_Speed</b></p> <p><b>Requested value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Request.<b>D_Speed</b></p>
Unit	1 µm/s
Data type	UDINT

Incremental jog path	
Description	Path traversed by the logical axis in incremental jog mode each time the incremental jog button is pressed.
ST path	<p><b>Commanded value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Command.<b>D_Distance</b></p> <p><b>Requested value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Request.<b>D_Distance</b></p>
Unit	0,1 µm
Data type	UDINT

## 9.2.2.3 Handwheel mode

### 9.2.2.3.1 Control Unit

Parameterising handwheel mode in manual mode	
Description	Control unit to manage data for parameterising handwheel mode in manual mode, including flow control of user data.
Data type	HBHRParameterControlUnit, see description Control unit with usage check
Special features	The manual mode parameter can be written at any time and it is stored internally in a table under the index 0. This value is only becomes effective in the corresponding operation mode when the axis is activated.
ST path	pMC[channel_idx]^addr^.MCControlHB_Data.HBHRParameterControlUnit_HRParameter.
Access	PLC reads <b>HBHRParameter_Request</b> and writes <b>HBHRParameter_Command + X_Enable</b>
Commanded, requested data	
ST element	<b>.HBHRParameter_Command</b> <b>.HBHRParameter_Request</b>
Data type	HBHRParameter [► 172], for description see Section User data [► 172]
Flow control commanded data	
ST element	<b>.X_CommandSemaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data.  PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	<b>.X_RequestSemaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE.  PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST element	<b>.X_Enable</b>

### 9.2.2.3.2 User data

Axis number	
Description	<p>Unique system-wide number of a logical axis.</p> <p>The specified logical axis is assigned the handwheel resolution which is the basis for moving the axis in manual mode handwheel mode.</p>
ST path	<p><b>Commanded value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBHRParameterControlUnit_HRParameter.HBHRParameter_Command.<b>W_LogAchsNr</b></p> <p><b>Requested value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBHRParameterControlUnit_HRParameter.HBHRParameter_Request.<b>W_LogAchsNr</b></p>
Data type	UINT
Value range	TwinCAT PLC typically [1, gNrAx]

Handwheel resolution	
Description	<p>Resolution of axis motion path for one handwheel revolution.</p> <p>The internally used total resolution of the axis in 0.1 µm per applied handwheel increment results from the current handwheel resolution in 0.1 µm/increment divided by the physical handwheel resolution in increment/revolution of the handwheel specified.</p> <p>Manual parameter list:</p> <pre>hr_data[0].hr_auf1_z    1000    # Incr./rev. - denominator hr_data[0].hr_auf1_n    14      # Incr./rev. - denominator</pre> <p>Programming command (specified here in mm/revolution):</p> <p>#HANDWHEEL [AX=X RES1=0.1 RES2=90.2 RES3=0.5] or</p> <p>#SET HR [0.1, 90.2, 0.5] X (old tool)</p>
ST path	<p><b>Commanded value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBHRParameterControlUnit_HRParameter.HBHRParameter_Command.<b>D_Resolution</b></p> <p><b>Requested value</b></p> <p>pMC[channel_idx]^addr^.MCControlHB_Data.HBHRParameterControlUnit_HRParameter.HBHRParameter_Request.<b>D_Resolution</b></p>
Unit	0,1 µm / Handradumdrehung
Data type	DINT

## 9.2.3 Control elements for manual mode

### 9.2.3.1 Enforcing a button press

The HLI has four similar control units to command button presses in parallel.

This section names the control units with HBKeyControlX where the placeholder X can assume the value "" (nothing), "2", "3" through to "9". The ST Path table section lists the names of all control units present.

**Transfer of GUI requests:**

To transfer a request received from the GUI to the PLC, the activated control units are checked in the order HBKeyControlUnit\_Key, HBKeyControlUnit\_Key2, HBKeyControlUnit\_Key3 through to HBKeyControlUnit\_Key9 whether the request semaphore is free and the first free request element is used to transfer the button press request to the PLC. The PLC must therefore process X\_RequestSemaphore for all control units for which they set X\_Enable to TRUE.

#### **9.2.3.1.1 Control Unit**

Enforcing a button press in manual mode	
Description	Control unit to manage data to enforce a button press in manual mode, including flow control of user data.
Data type	HBKeyControlUnit_Key, see description Control unit with usage check
Special features	The signalling of a button press is possible at any time but it remains without any effect if there is no axis assigned in continuous/incremental jog mode.
ST path	<p>PMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key</p> <p>PMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key2</p> <p>PMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key3</p> <p>PMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key4</p> <p>PMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key5</p> <p>PMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key6</p> <p>PMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key7</p> <p>PMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key8</p> <p>PMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key9</p>
Access	PLC reads <b>HBKey_Request</b> and writes <b>HBKey_Command + X_Enable</b>
Commanded, requested data	
ST element	<p><b>.HBKey_Command</b></p> <p><b>.HBKey_Request</b></p>
Data type	HBKey [▶ 176], for description see Section User data [▶ 176]
Flow control commanded data	
ST element	<b>.X_CommandSemaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	<p>CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data.</p> <p>PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.</p>
Flow control of requested data	
ST element	<b>.X_RequestSemaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	<p>CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE.</p> <p>PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.</p>
Redirection	
ST element	<b>.X_Enable</b>

### 9.2.3.1.2 User data

Button number	
Description	Logical button number from which the command comes.
ST path	<b>Commanded value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Command.W_LogKeyNr</code> <b>Requested value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Request.W_LogKeyNr</code>
Data type	UINT
Value range	One of the values which are defined as logical jog button pair numbers in the configuration list <code>hand_mds.lis</code> for the characteristics <code>tasten_data[X].log_tasten_nr</code> .

Start/end of button press		
Description	Start/end of button press event and direction of the motion direction keys in manual mode	
ST path	<b>Commanded value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Command.W_Direction</code> <b>Requested value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Request.W_Direction</code>	
Data type	INT	
Value range	<b>Value</b>	<b>Meaning</b>
	-1	Start of button press, negative motion direction
	0	End of button press
	1	Start of button press, positive motion direction
Special features	<b>Consumption data item</b> Since the enforcement of a button press is a message-orientated transfer, both the “start of button press” event and the “end of button press” event must be generated by the PLC.	



Lifetime of the button signal	
Description	If this element has a value unequal to 0, the CNC independently generates the “End of button press” event after receiving a “Start of button press” event after the time period expires and which was defined by the number of specified interpolator cycles.
ST path	<b>Commanded value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Command.D_LifeTime</code> <b>Requested value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Request.D_LifeTime</code>
Unit	Number of interpolator cycles
Data type	UDINT
Special features	See Note [▶ 176]

Retriggering “start of button press” event	
Description	Retriggering the “start of button press” event. If the element “Life time of a button signal” [▶ 176] has a value unequal to 0, the “start of button press” event is retriggered if the “Lifetime of the button signal” has not yet expired.
ST path	<b>Commanded value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Command.X_FRefresh</code> <b>Requested value</b> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Request.X_FRefresh</code>
Data type	BOOL
Value range	[TRUE = Retriggering the “start of button press” event, FALSE]
Special features	See Note [▶ 176]

**Note:** The “Lifetime of the button signal” and “Retriggering the ‘start of button press’ event” elements represent a sort of watchdog function. Their use is indicated in particular if the time response of the PLC is not ensured (deterministic, e.g. soft PLC as Windows task).

### 9.2.3.2 Rapid traverse velocity during path motion

During continuous jog mode it is possible to switch between the normal velocity and rapid traverse velocity. Rapid traverse velocity is defined in the axis-specific parameter list.

handbetrieb.tipp.vb\_eilgang (P-AXIS-00210)      4000

Here the rapid traverse is a button-specific feature and only becomes effective when the corresponding button is pushed and linked to an axis.

### 9.2.3.2.1 Control Unit

Rapid traverse velocity during path motion in manual mode	
Description	Control unit to activate/deactivate rapid traverse mode by a normal button press in manual mode.
Data type	HBRapidKeyControlUnit, see description Control unit with usage check
Special features	Button-specific signalling of rapid traverse mode is possible at any time and is saved internally for each button. But this only has an effect when the corresponding button is pressed.
ST path	pMC[channel_idx]^^.addr^.MCControlHB_Data.HBRapidKeyControlUnit_RapidKey
Access	PLC reads <b>HBRapidKey_Request</b> and writes <b>HBRapidKey_Command + X_Enable</b>
Commanded, requested data	
ST element	<b>.HBRapidKey_Command</b> <b>.HBRapidKey_Request</b>
Data type	HBRapidKey [► 178], for description see Section User data [► 178]
Flow control commanded data	
ST element	<b>.X_CommandSemaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data. PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	<b>.X_RequestSemaphor</b>
Data type	BOOL
Special features	<b>Consumption data item</b>
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE. PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST element	<b>.X_Enable</b>

### 9.2.3.2.2 User data

Button number	
Description	Logical button number for which the rapid traverse mode should be selected/deselected.
ST path	<b>Commanded value</b> pMC[channel_idx]^addr^.MCControlHB_Data.HBRapidKeyControlUnit_RapidKey.HBRapidKey_Command.W_LogKeyNr <b>Requested value</b> pMC[channel_idx]^addr^.MCControlHB_Data.HBRapidKeyControlUnit_RapidKey.HBRapidKey_Request.W_LogKeyNr
Data type	UINT
Value range	One of the values defined as logical jog button pair numbers in the configuration list hand_mds.lis for the characteristics tasten_data[X].log_tasten_nr

Rapid traverse velocity during path motion	
Description	Rapid traverse mode of the button on/off.
ST path	<b>Commanded value</b> pMC[channel_idx]^addr^.MCControlHB_Data.HBRapidKeyControlUnit_RapidKey.HBRapidKey_Command.W_KeyPressed <b>Requested value</b> pMC[channel_idx]^addr^.MCControlHB_Data.HBRapidKeyControlUnit_RapidKey.HBRapidKey_Request.W_KeyPressed
Data type	UINT
Value range	TRUE = Button for rapid traverse mode is active. The parameterised rapid traverse path motion is used for continuous jog mode. FALSE = Button not active in rapid traverse mode. The parameterised normal path velocity is used in continuous jog mode.]

### 9.2.3.3 Handwheel increments

Handwheel increments, counter state	
Description	Array of control units to manage the counts of handwheel increments for all handwheels, including flow control of user data.
Data type	MCCControlSGN32Unit, see description of Control Unit
Special features	Handwheel counts can be changed at any time but they only have an effect if there is an assigned axis in handwheel mode.  Relative changes to a handwheel count are only evaluated after manual mode is activated.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCCControlHB_Data. <b>AMCCControlSGN32Unit_HandWheelIncs[idx]</b>
Commanded, requested and return value	
ST element	<b>.D_Command</b> <b>.D_Request</b> <b>.D_State</b>
Data type	DINT
Redirection	
ST path	<b>.X_Enable</b>

## 10 Safety system

### 10.1 Channel-specific interface

#### 10.1.1 Watchdog mechanism

##### Watchdog, CNC monitors PLC

Description	The CNC uses the data of this unit to monitor whether the PLC is still ready for operation.
-------------	---



##### Notice

When the CNC monitors the PLC, the PLC task in which the PLC life sign `X_alive_rw [► 181]` is set must be executed with a higher priority than the two CNC tasks SDA and COM.

In order to prevent problems with CNC block supply in large PLC projects, it is also recommended to relocate the watchdog functionality to an extra PLC task and to only assign this task with a higher priority.

Data type	MCWatchDogUnit
ST Path	pMC[channel_idx]^^.addr^.HLIHeadChannel_Data. <b>MCWatchDogUnit_mc</b>
PLC operates functionality	
Description	By setting this element to the value TRUE, the PLC signals to the CNC that it supplies the data for the watchdog functionality.
ST Element	<b>.X_enable_w</b>
Data type	BOOL
Value range	[TRUE = PLC supplies data for a watchdog monitor, FALSE]
Access	PLC writes
PLC signal	
Description	CNC uses this element to determine whether the PLC is still ready for operation. This check is executed when the elements “PLC operates functionality” [► 181] and “PLC present” have the value TRUE.
ST element	<b>.X_alive_rw</b>
Data type	BOOL
Value range	[TRUE = PLC signals that it is ready for operation, FALSE]
Access	CNC detects whether this element changes from FALSE to TRUE within the time specified by the element “Watchdog cycle time” [► 181]. The CNC sets the value to FALSE after detecting this value.  CNC writes the value TRUE in this element in every CNC cycle in order to confirm its readiness for operation.
Signal “PLC not ready for operation”	
Description	If the CNC detects that the PLC is no longer ready for operation, the CNC sets this element to TRUE.
ST element	<b>.X_alive_state_r</b>
Data type	BOOL
Value range	[TRUE = PLC ready for operation, FALSE = PLC is not ready for operation]
Access	CNC writes
Watchdog cycle time	
Description	Cycle time of the watchdog.  The values for the watchdog cycle time must be greater than the PLC cycle time in order to allow practical monitoring of the PLC by the CNC.
ST element	<b>.D_cycle_time_w</b>
Unit	1 µs
Data type	UDINT
Access	PLC writes

## Watchdog, PLC monitors CNC

Description	The PLC uses the data of this unit to monitor whether the CNC is still ready for operation.
-------------	---



### Notice

When the PLC monitors the CNC, the PLC task in which a check is made for the CNC life sign `X_alive_rw` [▶ 181] must be executed at a higher priority than the two CNC tasks `SDA` and `COM` (see the example TwinCAT [▶ 181]). In order to prevent problems with CNC block supply in large PLC projects, it is also recommended to relocate the watchdog functionality to an extra PLC task and to only assign this task with a higher priority.

Data type	LCWatchDogUnit
ST Path	<code>pMC[channel_idx]^addr^.HLIHeadChannel_Data.LCWatchDogUnit_lc</code>
CNC signal	
Description	The CNC writes TRUE to this element in every interpolation cycle in order to confirm that the CNC is ready to operation.
ST Path	<code>.X_alive_rw</code>
Data type	BOOL
Access	The CNC writes the value TRUE in every CNC cycle to confirm its readiness for operation.  PLC detects whether this element changes from FALSE to TRUE within the time specified by the element "Watchdog cycle time" [▶ 181]. The PLC sets the value to FALSE after successful detection.
Signal "CNC not ready for operation"	
Description	If the PLC detects that the CNC is no longer ready for operation, it sets this element to TRUE.
ST Path	<code>.X_alive_state_w</code>
Data type	BOOL
Value range	[TRUE = CNC ready for operation, FALSE = CNC is not ready for operation]
Access	PLC writes
Watchdog cycle time	
Description	The NC kernel writes the interpolator cycle time in this element. The values for the watchdog cycle time must be greater than the PLC cycle time to permit practical monitoring of the CNC by the PLC.
ST Path	<code>.D_cycle_time_r</code>
Unit	1 µs
Data type	UDINT
Access	PLC reads

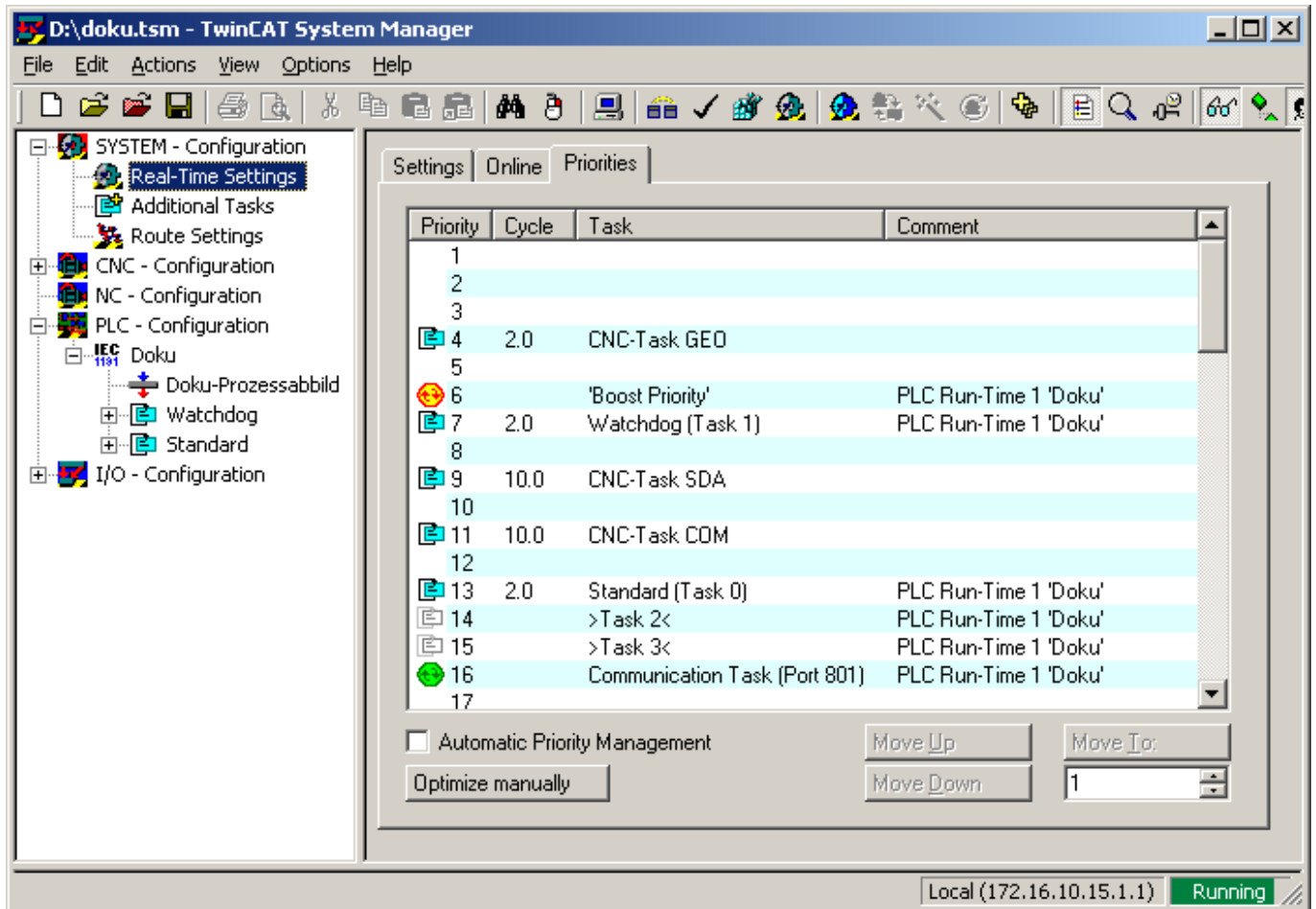


Fig. 22: Priorities for watchdog mechanism (example TwinCAT 2)



# 11 Management

## 11.1 Channel-specific interface

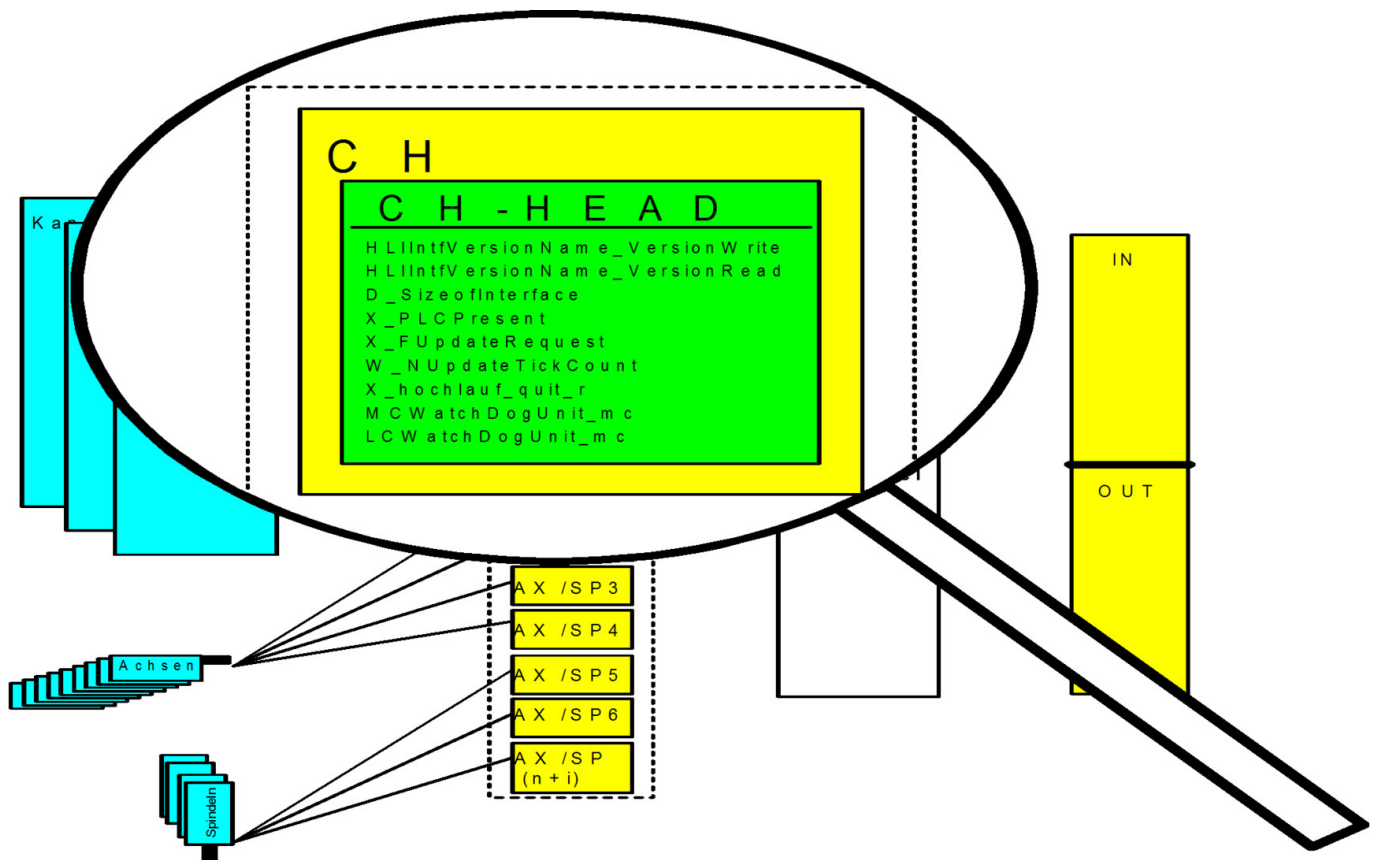


Fig. 23: Management data of the channel-specific interface

Version identifier PLC → CNC	
Description	If the version of the HLI at the NC kernel end is evaluated, the PLC stores the agreed version identifier in this element. Evaluation at the NC kernel end must be agreed for a specific application.
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.HLIIntfVersionName_Version-Write.AB_Zeichen
Data type	ARRAY [1..GCW_250_HLIIntfVersionNameLength] OF BYTE
Access	PLC writes
Special features	Is currently not evaluated in the NC kernel.

Version identifier CNC → PLC	
Description	If the version of the HLI is evaluated at the PLC end, the NC kernel stores the agreed version identifier in this element. Evaluation at the PLC end must be agreed for a specific application.
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.HLIIntfVersionName_Version-Read.AB_Zeichen
Data type	ARRAY [1..GCW_250_HLIIntfVersionNameLength] OF BYTE
Access	PLC reads

Size of the HLI	
Description	The NC kernel writes the size of the entire channel-specific interface area in this element.
Unit	Byte
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.D_SizeofInterface
Data type	UDINT
Access	PLC reads

PLC present	
Description	The PLC signals that it is present by setting the value to TRUE.
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.X_PLCPresent
Data type	BOOL
Value range	[TRUE = PLC present, FALSE]
Access	PLC writes



### Notice

Only if this value is set to TRUE can the PLC influence the NC kernel by channel-specific control units and be supplied with technology commands by the NC kernel.

NC kernel start-up completed	
Description	By setting this value to TRUE, the NC kernel signals to the PLC that the NC kernel start-up is completed and that the HLI is now supplied cyclically. This means that the display data is valid and the control commands are transferred to the NC kernel.
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.X_hochlauf_quit_r
Data type	BOOL
Value range	[TRUE = Start-up of NC kernel completed, FALSE]
Access	PLC reads

Updating the HLI	
Description	By setting this value to TRUE, the PLC can initiate an update of the entire axis-specific interface area. This value is reset to FALSE by the NC kernel after the update is completed.
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.X_FUpdateRequest
Data type	BOOL
Value range	[TRUE = Updating of the HLI activated, FALSE]
Access	PLC writes/reads
Special features	<b>Currently not supported, i.e. the HLI is updated in every NC cycle.</b>

Update cycle of the HLI	
Description	Using this element, the PLC informs the NC kernel of the number of interrupt cycles the NC kernel may distribute the updating of the axis-specific interface area. This leads to a reduced load on the interrupt task in particular with multi-channel and multi-axis configurations.
Unit	Number of cycles of the NC kernel
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.W_NUpdateTickCount
Data type	UINT
Access	PLC writes
Special features	<b>Currently not supported, i.e. the HLI is updated in every NC cycle.</b>

## 11.2 Axis-specific interface

PLC present	
Description	The PLC signals that it is present by setting the value to TRUE.
Signal flow	PLC → CNC
ST path	pAC[axis_idx]^addr^.HLIHeadAxe_Data.X_PLCPresent
Data type	BOOL
Value range	[TRUE = PLC present, FALSE]
Access	PLC writes



### Notice

Only if this value is set to TRUE can the PLC influence the NC kernel by axis-specific control units and be supplied with technology commands by the NC kernel.

Logical axis identifier	
Description	Logical number of an axis which is unique within the entire system and identifies the axis. This datum is available after controller start-up irrespective of the configured axis mode or the assignment of the axis to a channel.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.HLIHeadAxe_Data.W_LogAxNr
Data type	UINT
Access	PLC reads

Axis error	
Description	The CNC sets this value to TRUE if the axis is in error state.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.HLIHeadAxe_Data.X_Error
Data type	BOOL
Value range	[TRUE = Axis is in error state, FALSE = No error]
Access	PLC reads

Axis error, not resettable	
Description	The CNC sets this value to TRUE if an unresettable error was issued for an axis.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.HLIHeadAxe_Data.X_ResetErrorLocked
Data type	BOOL
Value range	[TRUE = Not resettable axis error occurred, FALSE = No error or resettable error occurred]
Access	PLC reads

Updating the HLI	
Description	By setting this value to TRUE, the PLC can initiate an update of the entire axis-specific interface area. This value is reset to FALSE by the NC kernel after the update is completed.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.HLIHeadAxe_Data.X_FUupdateRequest
Data type	BOOL
Value range	[TRUE = Updating of the HLI activated, FALSE]
Access	PLC writes/reads
Special features	<b>Still currently supported, i.e. the HLI is updated in every NC cycle.</b>

Update cycle of the HLI	
Description	Using this element, the PLC informs the NC kernel of the number of interrupt cycles the NC kernel may distribute the updating of the axis-specific interface area. This leads to a reduced load on the interrupt task in particular with multi-channel and multi-axis configurations.
Signal flow	PLC → CNC
Unit	Number of cycles of the NC kernel
ST path	pAC[axis_idx]^addr^.HLIHeadAxe_Data.W_NUupdateTickCount
Data type	UINT
Access	PLC writes
Special features	<b>Currently not supported, i.e. the HLI is updated in every NC cycle.</b>

## 12 Error messages

Errors occurring in the NC kernel can be displayed on the HLI interface in order to provide the PLC with the option of responding to the error and logging it.

The significance of the individual structure elements is documented in [DIAG].

Each error message is identified by a unique error number and the error messages are documented in [DIAG].

### 12.1 Management data of an error message

### 12.2 User data of an error message

Error number	
Description	Unique error number.
ST path	pMC[channel_idx]^addr^.MCErrSatz_Data.SatzR.Kopf. <b>ErrorId</b>
Data type	UDINT
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Module name of the module signalling an error	
Description	Module name of the module signalling an error
ST path	pMC[channel_idx]^addr^.MCErrSatz_Data.SatzR.Kopf. <b>ModulName</b>
Data type	STRING(GCW_250_HLIFileNameLaengeMinus1)
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Line number	
Description	Line number in the module on which the error occurred.
ST path	pMC[channel_idx]^addr^.MCErrSatz_Data.SatzR.Kopf. <b>Line</b>
Data type	INT
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Error number of a utility function	
Description	Error number when a utility function is used.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf.UtilErrorId
Data type	UDINT
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Module name of the module with utility functions signalling an error	
Description	Module name of the module with utility functions signalling an error
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf.UtilModulName
Data type	STRING(GCW_250_HLIFileNameLaengeMinus1)
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Line number of a utility function	
Description	Line number of the line in which the error occurred in a module with utility function.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf.UtilLine
Data type	INT
Special features	Used for internal diagnostic purposes.

Multiple error number	
Description	Error messages may be issued at several different points in the NC kernel. A unique multiple error number is issued to distinguish multiple usage.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf.MultipleId
Data type	UINT
Special features	Used for internal diagnostic purposes.

Type of commandable function	
Description	Type of commandable function in which an error occurred.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. <b>BFTyp</b>
Data type	UINT
Special features	Used for internal diagnostic purposes.

Channel number	
Description	Channel number of the channel in which the signalled error occurred.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. <b>CncChannel</b>
Data type	UINT

Communication ID	
Description	Communication ID of the BF signalling an error in the CNC
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. <b>Kommuld</b>
Data type	UINT
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Time specification on output of an error message:	
Description	Date of the instant of the error message
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. <b>FbZeitangabe.DateCounter</b>
Data type	UDINT
Special features	Currently not implemented, value is always 0.

Time at error message: interrupt cycles since system start	
Description	Number of interrupt cycles since system start at the instant of an error message
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. <b>FbZeitangabe.ZyklCounter</b>
Data type	UDINT
Access	PLC reads



CNC version name	
Description	Version name of the CNC specified in the error message.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. <b>VersionName</b>
Data type	STRING(GCW_250_HLIIntfVersNameLaengeMin1)

Recovery class	
Description	Recovery class of an error
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. <b>BehebungsKlasse</b>
Data type	UINT
Value range	[0, 8]

Reaction class of an error	
Description	Reaction class of an error
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. <b>ReaktionsKlasse</b>
Data type	UINT
Value range	[0, 8]

Body type of an error		
Description	Body type of an error. Depending on the error type, the error set body contains further information on the error which occurred. The individual structure elements are described in [DIAG].	
ST path	pMC[channel_idx]^^.addr^.MCErrorsatz_Data.SatzR.Kopf. <b>RumpfTyp</b>	
Value range	<b>Value</b>	<b>Constant</b>
	1	HLI_RumpfTypNcProg
	2	HLI_RumpfTypMds
	3	HLI_RumpfTypKommu
	4	HLI_RumpfTypRamDisk
	5	HLI_RumpfTypFile
	6	HLI_RumpfTypIntprFile
	7	HLI_RumpfTypListeBinaer
	8	HLI_RumpfTypGcm
	9	HLI_RumpfTypLeer
	10	HLI_RumpfTypHLI
	11	HLI_RumpfTypNCProgLr

## 12.2.1 Error message content, body nc program

NC program	
Description	Error information in relation to the NC program
Data type	RumpfNcProg
ST path	pNcProgErr : POINTER TO RumpfNcProg; <b>pNcProgErr</b> := ADR(pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Rumpf.Maske);
Description	Logical path number (see start-up list).
ST path	<b>.logPfadNr</b>
Data type	UINT
Value range	s. Start-up list
Description	Program name
ST path	<b>.ProgName</b>
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)
Description	Data name
ST path	<b>.FileName</b>
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)
Description	File offset in bytes
ST path	<b>.FileOffset</b>
Data type	UDINT
Description	Token offset in current NC line
ST path	<b>.TokenOffset</b>
Data type	UINT
Description	Block number of current NC line
ST path	<b>.SatzNumber</b>
Data type	UDINT
Description	Additional program information
ST path	<b>.SaddProgInfo</b>
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)

## 12.2.2 Error message content, body machine data

Machine data	
Description	Error information in relation to an update of machine data
Data type	RumpfMds
ST path	pMachineDataErr : POINTER TO RumpfMds; <b>pMachineDataErr</b> := ADR(pMC[channel_idx]^ .addr^ .MCError-Satz_Data.SatzR.Rumpf.Maske);
Description	Type of list
ST path	<b>.ListenTyp</b>
Data type	UINT
Value range	1 - manual data, 2 – channel list, 3 – axis list, 4 – zero offsets, 5 – tool data, 6 – clamp position, 7 – start-up list, 9 – axis compensations, 12 – external variables
Description	Name of wrong structure
ST path	<b>.StruktName</b>
Data type	STRING(GCW_250_HLIStruktNameLaengeMinus1)

### 12.2.3 Error message content, body communication

Communication	
Description	Error information in relation to a protocol data unit (message)
Data type	RumpfKommu
ST path	pCommuErr : POINTER TO RumpfKommu; <b>pCommuErr</b> := ADR(pMC[channel_idx]^ .addr^ .MCErrSatz_Data.SatzR.Rumpf.Maske);
Description	Type of list
ST path	<b>.Medium</b>
Data type	UINT
Value range	1 – function block, 2 – PDU
Description	Code of message
ST path	<b>.typ</b>
Data type	UDINT
Description	Receiver or sender of message
ST path	<b>.Partner</b>
Data type	UDINT

## 12.2.4 Error message content, body RAM disk

RAM disk	
Description	Error information in relation to RAM disk access
Data type	RumpfRAMDisk
ST path	pRamDiskErr : POINTER TO RumpfRamDisk; <b>pRamDiskErr</b> := ADR(pMC[channel_idx]^^.addr^.MCErr- Satz_Data.SatzR.Rumpf.Maske);
Description	Listentyp
ST path	<b>.Medium</b>
Data type	UINT
Value range	1 – function block, 2 – PDU
Description	Code of message
ST path	<b>.typ</b>
Data type	UDINT
Value range	
Description	Receiver or sender of message
ST path	<b>.Partner</b>
Data type	UDINT
Value range	
Description	File name
ST path	<b>.FileName</b>
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)
Description	Dateoffset in Bytes
ST path	<b>.FileOffset</b>
Data type	UDINT

## 12.2.5 Error message content, body file

File	
Description	Error information in relation to file access
Data type	RumpfFile
ST path	pFileErr : POINTER TO RumpfFile; <b>pFileErr</b> := ADR(pMC[channel_idx]^ .addr^ .MCErrSatz_Data.SatzR.Rumpf.Maske);
Description	File name
ST path	<b>.FileName</b>
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)
Description	File offset in bytes
ST path	<b>.FileOffset</b>
Data type	UDINT

## 12.2.6 Error message content, body interpreted file list

List interpretation		
Description	Error information in relation to interpretation of ASCII parameter list	
Data type	RumpflntprFile	
ST path	pIntprFileErr: POINTER TO RumpflntprFile; <b>pIntprFileErr</b> := ADR(pMC[channel_idx]^ .addr^ .MCErr- Satz_Data.SatzR.Rumpf.Maske);	
Description	File name	
Data type	<b>.FileName</b>	
ST path	STRING(GCW_250_HLIProgNameLaengeMinus1)	
Description	File offset in bytes	
Data type	<b>.FileOffset</b>	
ST path	UDINT	
Description	Line number in current file	
Data type	<b>.FileLine</b>	
ST path	UDINT	
Description	Listentyp	
Data type	<b>.ListenTyp</b>	
ST path	UINT	
Value range	<b>Value</b>	<b>Meaning</b>
	1	Manual mode list
	2	Channel parameter list
	3	Axis parameter list
	4	Zero offset list
	5	Tool parameter list
	6	Position offset list
	7	Start-up list
	9	Axis compensation list
Description	Name of wrong structure	
Data type	<b>.StruktName</b>	
ST path	STRING(GCW_250_HLIStruktNameLaengeMinus1)	



## 12.2.7 Error message content, body binary list

Binary list	
Description	Error information in relation to binary list update
Data type	pBinaryListErr : POINTER TO RumpflntprFile; <b>pBinaryListErr</b> := ADR(pMC[channel_idx]^ .addr^ .MCErr- Satz_Data.SatzR.Rumpf.Maske);
ST path	pMC[channel_idx]^ .addr^ .MCErrSatz_Data.SatzR.Rumpf.Maske. <b>NCProg</b>
Description	Name of wrong structure
ST path	<b>.StruktName</b>
Data type	STRING(GCW_250_HLIStruktNameLaengeMinus1)

## 12.2.8 Error message content, body global channel manager

Global channel manager	
Description	Error information in relation to Global Channel Manager
Data type	RumpfListeGcm
ST path	pGcmErr : POINTER TO RumpfGcm; <b>pGcmErr</b> := ADR(pMC[channel_idx]^ .addr^ .MCErrSatz_Data.SatzR.Rumpf.Maske);
Description	Name of wrong structure
ST path	<b>.Token</b>
Data type	STRING(GCW_250_HLIVarStringLaengeMinus1)
Description	File name
ST path	<b>.FileName</b>
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)
Description	Interpreter number
ST path	<b>.InterpNo</b>
Data type	UDINT
Description	Line number
ST path	<b>.LineNo</b>
Data type	UDINT
Description	Column number
ST path	<b>.ColNo</b>
Data type	UDINT
Description	Offset
ST path	<b>.OffsetNo</b>
Data type	UDINT
Description	Number of command
ST path	<b>.CommandNo</b>

Data type	UDINT
Description	Part ID
ST path	<b>.PartId</b>
Data type	UDINT
Description	Step ID
ST path	<b>.StepId</b>
Data type	UDINT
Description	Level of if constructs
ST path	<b>.IfLevel</b>
Data type	UDINT

## 12.2.9 Additional error information value 1 - 5

Additional error information	
Description	Individual error information
ST path	WertB
Data type	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Rumpf. <b>Wert1 ... Wert5</b>
Description	Data type
ST path	<b>.typ</b>
Data type	INT
Value range	See Value range of variable 'Type' of additional error information [► 205]
Description	Dimension of datum
ST path	<b>.Dimension</b>
Data type	INT
Value range	See Value range of 'Dimension' variable of the additional error information [► 206].
Description	Significance of datum
ST path	<b>.bedeutung</b>
Data type	INT
Value range	See Value range of 'Significance' variable of additional error information [► 207].
Description	Datum itself
ST path	<b>.Contents</b>
Data type	(as union, must be type-caste corresponding to the specified data type)

### 12.2.9.1 Value range of variable 'Type' of additional error information

Constant	Value	Description
HLI_TYP_BOOLEAN	1	Boolean value
HLI_TYP_UNSO8	2	Unsigned 1 byte value
HLI_TYP_SGN08	3	Signed 1 byte value
HLI_TYP_UNSO16	4	Unsigned 2 byte value
HLI_TYP_SGN16	5	Signed 2 byte value
HLI_TYP_UNSO32	6	Unsigned 4 byte value
HLI_TYP_SGN32	7	Signed 4 byte value
HLI_TYP_UNSO64	8	Unsigned 8 byte value
HLI_TYP_SGN64	9	Signed 8 byte value
HLI_TYP_REAL64	10	8 byte decimal value*
HLI_TYP_POINTER	12	Address
HLI_TYP_CHAR	18	Character, 1 byte
HLI_TYP_STRING	19	Character string
HLI_TYP_ADRESSE	20	Address
HLI_TYP_A3_REAL64	23	Array containing three 8-byte decimal values
HLI_TYP_HIGH_RES_SGN_POS	24	Signed value, size depends on system and may be 4 or 8 bytes.
HLI_TYP_BITARRAY_32	25	Array of 4-byte values
HLI_TYP_BITARRAY_16	26	Array of 2-byte values

[Back to description of 'Type' variable.](#)

### 12.2.9.2 Value range of 'Dimension' variable of the additional error information

Constant	Value	Unit	Description
HLI_NO_DIMENSION	-1	-	This is not a dimension specification
HLI_DIM_DIMENSIONSLOS	0	-	Without dimension
HLI_DIM_POSITION	1	$10^{-4}$ mm or $10^{-4}$ °	Position
HLI_DIM_GESCHWINDIGKEIT	2	$10^{-3}$ mm/s or $10^{-3}$ °	Velocity
HLI_DIM_BESCHLEUNIGUNG	3	$1 \text{ mm/s}^2$ or $1^\circ/\text{s}^2$	acceleration
HLI_DIM_RUCK	4	$1 \text{ mm/s}^3$ or $1^\circ/\text{s}^3$	jerk
HLI_DIM_ZEIT	5	1 µs	time
HLI_DIM_PROZENT	6	0,1 %	Prozent
HLI_DIM_ADRESSE	7	-	Address
HLI_DIM_INKREMENTE	8	Increments	Path increment
HLI_DIM_UMDR_VORSCHUB	9	$10^{-4}$ mm/rev	Feedrate per revolution
HLI_DIM_V_SCHNITT	10	$10^{-3}$ mm/s	Cutting velocity
HLI_DIM_WEG_AUFLOESUNG	11	Increments/ $10^{-3}$ mm	Path resolution
HLI_DIM_INKR_UMDREHUNG	12	Increments/rev.	Increments per revolution
HLI_DIM_BYTE	13	-	Number of bytes
HLI_DIM_PROPORTIONAL_GAIN	14	0.01/s	Proportional gain
HLI_DIM_FREQUENCY	15	Hz	Frequency
HLI_DIM_LOAD	16	1 kg or $1 \text{ kg}\cdot\text{m}^2$	Motor load
HLI_DIM_POSITION_HIGH_RES	17	$10^{-8}$ mm or $10^{-8}$ °	Position, high resolution
HLI_DIM_INKREMENTE_HIGH_RES	18	$10^{-4}$ increments	Increment, high resolution

Back to description of variable dimension.

### 12.2.9.3 Value range of 'Significance' variable of additional error information

Constant	Value	Description
HLI_BEDEUT_IGNORE	0	Ignore value may not be changed: Initialise with 0
HLI_BEDEUT_GRENZ_WERT	1	Limit value
HLI_BEDEUT_AKT_WERT	2	Current value
HLI_BEDEUT_FEHL_WERT	3	Error value
HLI_BEDEUT_ERWARTET_WERT	4	Expected value
HLI_BEDEUT_KORR_WERT	5	Corrected value
HLI_BEDEUT_LOG_ACHS_NR	6	logical axis identifier
HLI_BEDEUT_ANTR_TYP	7	Drive type
HLI_BEDEUT_LOG_BED_ELEM_NR	8	logical identifier of control element
HLI_BEDEUT_ZUSTAND	9	State
HLI_BEDEUT_TRANSITION	10	Transition
HLI_BEDEUT_SENDER	11	Sender
HLI_BEDEUT_KLASSE	12	Class
HLI_BEDEUT_INSTANZ	13	Instance
HLI_BEDEUT_IDENT_NR	14	Identification number
HLI_BEDEUT_STATUS	15	Status
HLI_BEDEUT_RING_NR	16	Ring number
HLI_BEDEUT_SATZ_NR	17	Block number
HLI_BEDEUT_MIN_LIMIT	18	Lower limit value
HLI_BEDEUT_MAX_LIMIT	19	Upper limit value
HLI_BEDEUT_START_WERT	20	Initial value
HLI_BEDEUT_ZIEL_WERT	21	Final value
HLI_BEDEUT_FILENAME	22	File name
HLI_BEDEUT_LINE_NUMBER	24	Line number in a file
HLI_BEDEUT_COLUMN_NUMBER	25	Column number in a file
HLI_BEDEUT_ARGUMENT	26	Identifier of an argument
HLI_BEDEUT_PARAMETER	27	Identifier of a parameter
HLI_BEDEUT_AXIS	28	Identifier of an axis
HLI_BEDEUT_COMPENSATION	29	Compensation index
HLI_BEDEUT_IDENTIFIER	30	Identifier
HLI_BEDEUT_CHAIN	31	Kinematic chain

[Back to description of variable significance.](#)

## 13 Messages

### 13.1 Control Unit

Message to/from PLC	
Description	<p>This control unit receives messages sent by the CNC to the PLC and messages sent by the PLC to other users of the communication process provided by the CNC.</p> <p>The CNC can send a message to the PLC via the commands <b>#MSG PLC</b> ["..."] or <b>#MSG SYN PLC</b> ["..."] (see [PROG]). The information on which of the commands was used to send a message is also sent in the user data of this control unit.</p>
Data type	McControlMsgUnit, see description Control unit with usage check
ST path	pMC[channel_idx]^addr^ <b>MsgMcControl_Data</b>
Commanded, requested data	
ST element	<b>.CommandW</b> <b>.RequestR</b>
Data type	HliMsgSendung
Access	PLC writes CommandW and reads RequestR
Flow control of commanded value	
ST element	<b>.CommandSemaphorRw</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	<b>Consumption data item</b>
Access	<p>PLC can write data for commanding if CommandSemaphorRw has the value FALSE. When all data is written, the PLC sets CommandSemaphorRw to the value TRUE.</p> <p>CNC takes the commanded data if CommandSemaphorRw is TRUE and then sets the flag to FALSE.</p>
Flow control of requested data	
ST element	<b>.RequestSemaphorRw</b>
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	<b>Consumption data item</b>
Access	<p>The data requested by the GUI can be written to the control unit if RequestSemaphorRw is FALSE. This element is then set to TRUE.</p> <p>PLC reads the data requested by the GUI if RequestSemaphorRw is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.</p>
Redirection	
ST path	<b>.EnableW</b>



## 13.2 User data

### 13.2.1 Requested and commanded user data

Content of message to/from PLC	
Description	Data of message which was received or sent.
ST path	<b>Commanded value</b> pMC[channel_idx]^ .addr^ .MsgMcControl_Data.CommandW <b>Requested value</b> pMC[channel_idx]^ .addr^ .MsgMcControl_Data.RequestR
ST element	<b>.Contents</b>
Data type	ARRAY [1 .. GCW_250_HliMessageLaenge] OF BYTE

Type of message to/from PLC		
Description	Marks whether a message was sent by CNC to the PLC by a #MSG or #MSGINFO command (see @@[PROG]).	
ST path	<b>Commanded value</b> pMC[channel_idx]^ .addr^ .MsgMcControl_Data.CommandW <b>Requested value</b> pMC[channel_idx]^ .addr^ .MsgMcControl_Data.RequestR	
ST element	<b>.typ</b>	
Data type	UDINT	
Value range	<b>Value</b>	<b>Meaning</b>
	0	Initial value
	1	Type of message which was sent by a <b>#MSG</b> command (if it was sent by the CNC)
	2	Type of message which was sent by a <b>#MSGINFO</b> command (if it was sent by the CNC)

## 14 Implementation as PLC library

To access the HLI, a PLC library is provided for use by the PLC application developer. This library is implemented specifically for various PLC systems. The table below contains the names of the libraries for the supported PLC systems.

PLC system	KW	3S	TwinCAT
HLI PLC library	-	-	tccnchli1.lib

This library implements

data structures which are components of the HLI.

global variables that are used to access the elements of the HLI

function blocks which are used to initialise global pointers

### 14.1 Access to HLI

#### 14.1.1 PLC system TwinCAT

In this system, the global pointers to specific areas of the HLI are created in the HLI-PLC library. These pointers are used to access control units and status data or other data for this type of HLI area. The table below shows the uses of these pointers and in each of the individual PLC systems:

	Explanation	PLC systems
<b>Global pointers</b>		<b>TwinCAT</b>
pPlatform	Platform-specific area	X
pMC	Array of pointers to channel-specific areas	X
pAC	Array of pointers to axis-specific areas	X
pCTM	Array of pointers to job planning areas	X
pVEGlob	Pointers to global V.E variables	X
pVE	Array of pointers to channel-specific V.E variables	X

Global pointers to HLI areas

Make sure only to initialise the pointers which make sense depending on the configuration of the NC kernel (number of channels number of axes, etc.). All other pointers are ZERO pointers.

### 14.2 Function blocks within the PLC library

#### 14.2.1 Overview of the PLCopen FBs

The table below provides an overview of the availability of the function blocks implemented in the PLC systems supported by ISG.

Function block	PLC systems		
	KW	3S	TwinCAT
ISG_HliInterface	-	-	X

## 14.2.2 ISG HLI Interface

An instance of the ISG\_HliInterface must be **definitely** invoked by every PLC application for whose system this FB was implemented and which wants to access the HLI elements.

The NC kernel requests information about the HLI configuration (number of channels, number of axes, etc.) on a rising edge at the bStart input and compares the HLI configuration with the PLC configuration. If differences occur, the bError output is set to TRUE and an error message identifier is output at the iErrorID output. On the other hand, if the attributes of the HLI are identical on both sides, the bInitialized output is set to TRUE and the PLC application can access the relevant areas of the HLI via the global defined pointers (see PLC main program frame).

### Block diagram

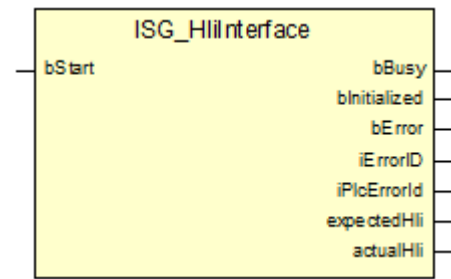


Fig. 24: FB parameters

VAR_INPUT		
Variable name	Data type	Description
bStart	BOOL	Rising edge triggers a check on whether the HLI definitions of the PLC and the NC kernel are identical.

VAR_OUTPUT		
Variable name	Data type	Description
bBusy	BOOL	Request and checking process are still active.
bInitialized	BOOL	The check was successfully completed and now <b>access</b> to the HLI is permitted for the first time.
bError	BOOL	Is TRUE if an error occurs in the FB.
iErrorId	UDINT	Error identifier. For possible values, see errors. Reference source not found.
iPlcErrorId	UDINT	Specific error message of the PLC system used. For further explanations, see errors. Reference source not found. Error value 9 (ISG_PLC_PFORM_ERROR)
expectedHli	HLI_DIAGNOSTIC	Information regarding the HLI configuration on the PLC side. The data structure contains the number of main areas of the HLI and the version identifier of the HLI definitions.
actualHli	HLI_DIAGNOSTIC	Contains the HLI configuration on the NC kernel side.

#### Behaviour of the FB:

- Table of error identifiers

• Error value	• Global constant • defined in PLC library	• Description
• 0	• ISG_NO_ERROR	• No error occurred
• 1	• ISG_WRONG_VERSION	• Version identifier of HLI from the PLC and NC kernel are different.
• 2	• ISG_WRONG_PARAMETER	• Different configuration (number of channels or number of axes or etc.) of the HLI in PLC and NC kernel.
• 9	• ISG_PLC_PFORM_ERROR	<ul style="list-style-type: none"> <li>• The PLC system requested the NC kernel to send a description of the interface (HLI) on the NC side. This request was acknowledged negatively. An error value specific to the PLC system is output at the <b>iPlcErrorId</b> output, if available (example: with TwinCAT, this case involves an ADS error whose error number is output at the <b>iPlcErrorId</b> output).</li> <li>•</li> <li>• <b>Reason for this error message:</b></li> <li>• On a TwinCAT system, an empty CNC configuration for the CNC was activated with the system manager.</li> <li>•</li> <li>• <b>Solution:</b></li> <li>• Activate a CNC configuration which contains the axes and/or channels for the CNC.</li> </ul>

Error value from ISG\_HliInterface

## 15 Programming examples

### 15.1 PLC main program frame

```
PROGRAM MAIN
VAR
    HLI                               : ISG_HliInterface;

    HliInitError                      : BOOL := FALSE; (* Error on initialisation of
HLI *)
    UserInitialisationDone : BOOL := FALSE; (* User initialisation done *)
END_VAR

(* Request description of the HLI from the CNC *)
Hli(bStart := TRUE);

(* Check if initialisation of HLI ended successfully and if
errors occurred during initialisation phase. *)

IF Hli.bInitialized = TRUE AND Hli.bError = FALSE
THEN
    (* Do the initialization we do once the PLC starts up. *)
    IF UserInitialisationDone = FALSE THEN
        (* Get the result of the user defined initialisation *)
        UserInitialisationDone := UserInitialisations [ 213 ] (dummy:=TRUE);
    END_IF;

    (* ----- *)
    (* Insert your PLC application code after this comment *)
    (* ----- *)

    IF Hli.bError = TRUE THEN
        (* Error on initialisation of the HLI *)
        (* iErrorId contains error number *)

        HliInitError := TRUE;
    END_IF;
END_IF;
```

#### 15.1.1 Initialisation function UserInitialisations()

```
FUNCTION UserInitialisations : BOOL

VAR_INPUT
    dummy : BOOL := FALSE; (* not_used *)
END_VAR
VAR
    AxIdx : UDINT;
    ChIdx : UDINT;
END_VAR
```

```
(* Register PLC at all axes interfaces *)
FOR AxIdx := 1 TO nAxis DO
  (* Set plc_present_w at each axis *)
  pAC[AxIdx]^addr^.HLIHeadAxe_Data.X_PLCPresent := TRUE;

  (* Register at all axis specific control units you want to handle by
  PLC *)
  (* Register at all control units to enable a drive *)
  pAC[AxIdx]^addr^.McControlLr_Data.MCControlBoolUnit_TorquePermis-
  sion.X_Enable
    := TRUE;
  pAC[Laufl]^addr^.McControlLr_Data.MCControlBoolUnit_ReleaseFeed-
  hold.X_Enable
    := TRUE;
  pAC[Laufl]^addr^.McControlLr_Data.MCControlBoolUnit_DriveOn.X_En-
  able := TRUE;

END_FOR;

(* Register PLC at all channel interfaces *)
FOR ChIdx := 1 TO nChannel DO
  (* Set plc_present_w at each channel *)
  pMC[ChIdx]^addr^.HLIHeadChannel_Data.X_PLCPresent := TRUE;

  (* Register at all channel specific control units you want to handle
  by PLC *)

END_FOR;

UserInitialisations := TRUE;
```

# Keyword index

## A

---

Antriebsdaten lesen.....	37
Antriebsdaten zyklisch lesen/schreiben .....	70
Antriebsdrift .....	39
Antriebstyp .....	34
AxeHBDisplayData_Data .....	161
Abstandsregelung	
Achse	
Achsgruppe	
Achsname	
Achsnummer	
Achstyp	
ACS	
Aktivierung von Bedienelementen	
Aktualisierung	
Antrieb	
Anzahl	
assignment	
Auflösung	
Auftragsfehler	
Ausführungsdauer	
axis	
Axis supply	

Beauftragung .....	64	Position:Soll .....	24
Kommando .....	66	Position:Ziel .....	24
Status .....	67	Referenzpunktfahrt erfolgt .....	27
Abstandsregelung:Beauftragung .....	64	Handbetrieb .....	164
Abstandsregelung:Kommando .....	66	HLI:achsspezifisch .....	189
Abstandsregelung>Status .....	67	HLI:kanalspezifisch .....	187
Deaktivierung .....	59	betriebsbereit .....	35
fährt vorwärts .....	29	Daten .....	37
Fehler .....	188	EIN .....	46
Fehler:nicht resetbar .....	189	Fehler .....	36
Gantry:Differenz:ausfahren .....	54	Istwert:gültig .....	30
Identifikation .....	20	Leistungszuschaltung:bereit zur .....	34
im Regelfenster .....	28	Zustand .....	36
in Position .....	28	Technologiefunktion:nicht quittiert:satzübergreifend .....	143, 145, 146
Interpolator:Zustand .....	31	Technologiefunktion:satzübergreifend:Achse... ..	135
ist bewegt .....	28	Technologiefunktion:satzübergreifend:Kanal ... ..	136
Konfiguration:Modus .....	31	Technologiefunktion:satzweise:Achse .....	134
Kopplung:Beauftragung .....	60	Technologiefunktion:satzweise:Kanal .....	135
Kopplung:Kommando .....	62	axis:logical .....	31
Kopplung:Nummer:Master .....	33	Handrad .....	162
Kopplung>Status .....	61	Spindel .....	79
Kopplung:Typ .....	32	H-Funktion .....	142
Kopplung:Vorschrift .....	63	M-Funktion .....	142
Nachführbetrieb .....	53	S-Funktion .....	144
Name:PCS .....	20	assignment:logical .....	31
Nummer:Gantry-Master .....	31	physical axis to logical axis .....	30
Nummer:logische .....	20, 188		
parken .....	59		
Position:ACS .....	24		
Position:Messwert .....	58		
Position:PCS .....	22		
Referenzposition .....	56		
Referenzposition:Löschen .....	57		
Referenzposition:Übernahme .....	55		
Simulation:Bearbeitungs .....	50		
Sollwerte:externe .....	72		
Spindel:ACS .....	21		
Technologiefunktion:Daten .....	137		
Typ:ACS .....	33		
Typ:PCS .....	21		
Vorschubfreigabe .....	43		
Vorschuboverride .....	45		
Vorschuboverride gültig .....	46		
Vorschubstopp EIN/AUS .....	44		
Warten auf angeforderte .....	94		
wird verfahren .....	29		
Zustand:ACS .....	26		
Zustand:PCS .....	26		
in Position .....	93		
Warten auf in Position .....	93		
PCS .....	20		
Handbetrieb:Aktivierung von Bedienelementen ..	165		
Handbetrieb:Handradbetrieb .....	173		
Handbetrieb:Jogbetrieb .....	171		
Handbetrieb:Tippbetrieb .....	169		
S-Funktion .....	145		
ACS .....	33		
PCS .....	21		
Achse:Spindel .....	21		
Achse:Typ .....	33		
Achse:Zustand .....	26		
Position:Ist .....	24		

## B

Bahngeschwindigkeit -Vorschubaktivierung externe .....	118
Bedienelementnummer .....	162
Betriebsart .....	71
Bahn .....	
Bahngeschwindigkeit -vorschub .....	
Bearbeitung .....	
Bearbeitungssimulation .....	
Beauftragbare Funktion .....	
Beauftragung .....	
Bedienelement .....	
Behebungsklasse .....	
Betriebsartenumschaltung .....	
Betriebszustand .....	
Bewegung .....	
Box .....	



Stopp:Grund .....	90
Tangente .....	101
Tangente:Normalenvektor .....	102
Tangente:Vektor .....	102
aktuell .....	89
externe Vorgabe .....	117
Grenzwert unterschritten .....	95
programmiert .....	88
programmiert:Echtzeiteinfluß .....	89
Simulation:Achse .....	50
EIN/AUS .....	114
Typ:bei Fehlermeldung .....	192
Spindel:extern .....	82
Spindel:Typ .....	84
Nummer .....	162
Handbetrieb:Aktivierung von Bedienelementen .....	165
bei Fehlermeldung .....	193
Ausgangs .....	154
Ausgangszustand:Betriebsartwechsel .....	155
Control Unit .....	153
Handbetrieb .....	161
Istwert .....	158
Parameter:Betriebsartwechsel .....	157
Zielzustand:Betriebsartwechsel .....	156
Zustands-Istwert .....	158
Zielzustand .....	155
Spindel .....	77
Bahn:Satznummer .....	96
fortsetzen:Kollision .....	96
Fortsetzung .....	113
Freigabe .....	93
TwinCAT:Ads NetId .....	42
TwinCAT:Kanal .....	42

## C

Channel	
Command value	
Control Unit	
Operation mode .....	107
Activation:external .....	72
Betriebsart .....	153

## D

DRIVE_STATE_MODE_0 .....	37
DRIVE_STATE_MODE_1 .....	37
DRIVE_STATE_MODE_2 .....	38
DRIVE_STATE_MODE_3 .....	38
DRIVE_STATE_MODE_4 .....	38
DRIVE_STATE_MODE_5 .....	38
DRIVE_STATE_MODE_6 .....	39
Daten	
Deaktivierung	
Dekodierung	
Dienstfunktion	
Drehrichtung	
Drehzahl	
Drehzahlüberwachung	
Durchsetzung eines Tastendrucks	

Antrieb .....	37
Technologiefunktion:Achse .....	137
Technologiefunktion:Kanal .....	139
Achse .....	59
NC-Programm:Fortsetzung .....	105
NC-Programm:Stopp .....	106
Fehlernummer .....	191
Spindel .....	85
programmiert .....	76
aktiv .....	78
ungültig .....	79
Handbetrieb .....	175

## E

Einzelatzbetrieb .....	112
Ecke	
Eilgang	
Eilganggeschwindigkeit	
Eilganggeschwindigkeit während Verfahrbewegung	
Einlesefreigabe	
Einleesperre	
Einzelatzmodus	
extern	
external	
externe	
Grenzwert:Abstand:nach .....	99
Grenzwert:Abstand:vor .....	99
Wartezeit .....	99
Override:Kanal .....	115
Bewegung .....	96
Handbetrieb .....	178
Handbetrieb:Eilganggeschwindigkeit während Verfahrbewegung .....	179
Unterdrückung .....	50
Interpolator .....	118
NC-Satz .....	104
Vorgabe Bahngeschwindigkeit .....	117
Command value:Activation .....	72
Bahngeschwindigkeit:Aktivierung .....	118

## F

Fahrweg .....	83
Fehlernummer .....	190
Fahren	
Fehler	
Fehlermeldung	
Fehlermeldungsrumpl	
Fläche	
Fortsetzposition	
Fortsetzung	
Freigabe	
Funktionsbaustein	

rückwärts .....	126
aktuell:NC-Programm .....	88
aktuell:NC-Satz .....	87
Rest:NC-Satz .....	27
Rest:verwerfen:Ende-Marke .....	125
Rest:verwerfen:Kommando .....	124
Rest:verwerfen:Zustand .....	98
Achse .....	188
Achse:nicht resetbar .....	189
Behebung erwartet .....	92
Spindel .....	81
warten auf externe Vorgabe .....	92
Beauftragbare Funktion:Typ .....	192
Behebungsklasse .....	193
Fehlernummer .....	190
Kanalnummer .....	192
Kommunikations-Id .....	192
Modulname .....	190
Modulname:Dienstfunktion .....	191
Nummer:Dienstfunktion .....	191
Nummer:Mehrfachfehler .....	191
Reaktionsklasse .....	193
Rumpftyp .....	194
Versionsbezeichnung CNC .....	193
Zeilennummer .....	190
Zeilennummer:Dienstfunktion .....	191
Zeitangabe:Interruptzyklen .....	192
Zeitangabe:Interruptzyklen seit Systemstart ....	192
binary list .....	201
Datei .....	199
global channel manager .....	202
IListeninterpretation .....	200
Kommunikation .....	197
Maschinendaten .....	196
NC program .....	195
RAM disk .....	198
Wert 1 – 5 .....	204
Dienstfunktion .....	191
Mehrfachfehler .....	191
Normalenvektor .....	102
Satzvorlauf:Abstand .....	95
der Bewegung .....	113
NC-Programm:Dekodierung .....	105
Bearbeitung nächster NC-Satz .....	105
Bewegung .....	93
Regler:Kommando .....	47
Regler:Zustand .....	29
ISG_HliInterface .....	211

## G

G-Funktion .....	83
G01 .....	
G100 .....	
G108 .....	
G200 .....	
G201 .....	
Gantry .....	
Gerät .....	
Gesamtoffset .....	
Geschwindigkeit .....	
Grenzwert .....	

Größe .....	
Zeitangabe .....	98
aktiv .....	98
aktiv .....	98
aktiv .....	97
aktiv .....	97
Differenz:ausfahren .....	54
Masterachse:Nummer .....	31
Identifikation:TwinCAT .....	42
PCS .....	23
Achse:Reduktion .....	48
Achse:reduziert, Zone 1 .....	49
Achse:reduziert, Zone 2 .....	49
Bahn:Grenzwert unterschritten .....	95
Eilgang:Achssbewegung .....	96
Kanal:Reduktion .....	121
Kanal:reduziert, Zone 1 .....	121
Kanal:reduziert, Zone 2 .....	122
minimale:Werkzeug:Ignorierung .....	51
NC-Satz, aktuell:Ende .....	100
Ecke:Abstand:nach .....	99
Ecke:Abstand:vor .....	99
HLI .....	186

## H

Handradauflösung .....	162
HBActivation_Command .....	165
HBActivationControlUnit_Activation .....	164
Halt .....	
Handbetrieb .....	
Handbetriebsart .....	
Handbetriebsparametersatz .....	
Handrad .....	
Handradbetrieb .....	
Handradinkremente Zählerstand .....	
Hardware .....	
H-Funktion .....	
HLI .....	
Hochlauf .....	

Bahn:Bedingung .....	90
Nothalt:Kanal .....	112
Nothalt:Spindel .....	81
wahlweiser .....	113
Achsnnummer:Aktivierung von Bedienelementen .....	165
Achsnnummer:Handradbetrieb .....	173
Achsnnummer:Jogbetrieb .....	171
Achsnnummer:Tippbetrieb .....	169
aktiv:mit paralleler Interpolation .....	97
aktiv:ohne parallele Interpolation .....	97
Aktivierung von Bedienelementen .....	164
Bedienelementnummer:Aktivierung von Bedienelementen .....	165
Betriebsart .....	161
Durchsetzung eines Tastendrucks .....	175
Eilganggeschwindigkeit während Verfahrbewegung .....	178
Eilganggeschwindigkeit:Eilganggeschwindigkeit während Verfahrbewegung .....	179
Handbetriebsart:Aktivierung von Bedienelementen .....	166
Handradauflösung .....	173
Handradbetrieb .....	172
Handradinkremente Zählerstand .....	180
Jogbetrieb .....	170
Joggeschwindigkeit .....	171
Jogweg .....	171
Lebenszeit des Tastensignals:Durchsetzung eines Tastendrucks .....	177
Offset:zurück fahren .....	52
Offset:zurück fahren:Anhalten .....	53
Retriggern Tastendruck-Beginn-Ereignisses:Durchsetzung eines Tastendrucks .....	177
Tastendruck Beginn/Ende:Durchsetzung eines Tastendrucks .....	176
Tastennummer:Durchsetzung eines Tastendrucks .....	176
Tastennummer:Eilganggeschwindigkeit während Tastendruck .....	179
Tippbetrieb .....	168
Tippgeschwindigkeit .....	169
Zustand .....	161
Aktivierung von Bedienelementen .....	166
Handbetrieb:Aktivierung von Bedienelementen .....	166
Auflösung .....	162
Handbetrieb .....	173
Achsnnummer .....	173
Auflösung .....	173
Parametrierung .....	172
Handbetrieb .....	180
extern:Messen .....	73
extern:Messen:Auftrag .....	74
extern:Messen:Flanke .....	74
extern:Messen:Nummer:Auftrag .....	74
extern:Messen:Nummer:Eingang .....	74
Ausführungsdauer, voraussichtlich .....	142
Daten .....	142
Nummer .....	142
Programmzeilennummer .....	143
Satznummer .....	142
Zusatzinformation .....	143

Aktualisierung:achsspezifisch .....	189
Aktualisierung:kanalspezifisch .....	187
Größe .....	186
Updatezyklus:achsspezifisch .....	189
Updatezyklus:kanalspezifisch .....	187
Zeiger .....	210
NC-Kern, abgeschlossen .....	187

## I

ISG_HllInterface .....	211
Istdrehzahl .....	76
Index	
Inhalt	
Interpolator	
Istposition	
Platzversatzgruppe .....	108
Nachricht:an/von PLC .....	209
Achse:Zustand .....	31
aktiv .....	92
Einlesesperre .....	118
ACS .....	24
PCS .....	22
Spindel .....	77

## J

Jogbetrieb	
Joggeschwindigkeit	
Jogweg	
Achsnnummer .....	171
Geschwindigkeit .....	171
Parametrierung .....	170
Verfahrgerwindigkeit .....	162
Verfahrweg .....	163
Weg .....	171
Handbetrieb .....	171
Handbetrieb .....	171

## K

Kanal	
Kanalnummer	
Kollision	
Kommunikations-Id	
Kompensation	
Konfiguration	
Kopplung	

Nummer.....	21
Technologiefunktion:Daten.....	139
Vorschubstopp EIN/AUS .....	109
bei Fehlermeldung.....	192
Betriebsart:Umschaltung .....	158
Bewegung:fortsetzen.....	96
bei Fehlermeldung.....	192
Antriebsdrift .....	39
Spindelsteigungsfehler .....	39
Temperatureinfluss.....	40
Umkehrspanne .....	40
Achse:Modus.....	31
Achse:Beauftragung .....	60
Achse:Kommando .....	62
Achse:Nummer:Master.....	33
Achse>Status .....	61
Achse:Typ .....	32
Achse:Vorschrift .....	63

## L

Lebenszeit des Tastensignals	
Handbetrieb:Durchsetzung eines Tastendrucks	
.....	177

## M

Messsignal .....	57
Messwert gültig .....	41
M-Funktion .....	83
MCS	
Mehrfachfehlernummer	
Messen	
Messtaster	
Messvorgang	
Messwert	
Modulname	
Position:Soll:Werkzeugmittelpunkt .....	25
bei Fehlermeldung.....	191
Hardware:extern.....	73
Hardware:extern:Auftrag .....	74
Hardware:extern:Flanke .....	74
Hardware:extern:Nummer:Auftrag .....	74
Hardware:extern:Nummer:Eingang.....	74
Zustand .....	41
aktiv .....	98
Position:Achse.....	58
Ausführungsdauer, voraussichtlich .....	142
Daten.....	142
Nummer.....	142
Nummer:Spindelpositionierfunktion.....	145
Nummer:Spindelschaltfunktion.....	144
Programmzeilennummer .....	143
Satznummer .....	142
Zusatzinformation .....	143
bei Fehlermeldung.....	190
Dienstfunktion:bei Fehlermeldung.....	191

## N

Nachführbetrieb .....	53
-----------------------	----

Nachricht	
NC block	
NC-Kern	
NC-Programm	
NC-Satz	
Nothalt	
Nummer	
an/von PLC.....	208
an/von PLC:Inhalt .....	209
an/von PLC:Typ .....	209
Skip mode.....	104
Hochlauf abgeschlossen .....	187
Dekodierung:Fortsetzung .....	105
Dekodierung:Stopp.....	106
Fahrweg:aktuell .....	88
Zeile:Zähler.....	88
aktiver/folgender:Winkel, zwischen .....	100
aktuell:Geschwindigkeit:Satzende .....	100
eingefügt.....	97
Einzelsatzmodus .....	104
Fahrweg:aktuell .....	87
Fahrweg:Rest .....	27
Freigabe Bearbeitung, nächster .....	105
Kanal .....	112
Spindel.....	81
Achse:Gantry-Master.....	31
Achse:logische .....	20, 188
Kanal .....	21
Werkzeug:einzuwechselndes .....	147
Werkzeug:Schwesterwerkzeug .....	147
Werkzeug:Variantenwerkzeug.....	147

## O

Offset	
Operation mode	
Orientierung	
OTC	
output of command values to real axes	
Override	
Handbetrieb:zurück fahren .....	52
Handbetrieb:zurück fahren:Stopp.....	53
OTC .....	52
Channel .....	107
Werkzeug .....	101
Offset .....	52
radius .....	126
interrupt .....	116
Eilgangvorschub:Kanal .....	115
Vorschub:Achse .....	45
Vorschub:Achse:gültig.....	46
Vorschub:Kanal .....	114
Zeit.....	123
Zeit:gültig .....	122

## P

pAC	
Parameter	
Parken	
PCS	
Platzversatzgruppe	

## PLC

Position

Positionssollwert

Programmende

Programmzeilennummer

W_AxLinkMasterAxNr .....	33
X_LinkToLogicalAxis .....	30
X_LinkToPhysicalAxis .....	31
Betriebsartwechsel .....	157
Achse .....	59
Achse:logische Nummer .....	20
Achse:Name .....	20
Achse:Typ .....	21
Achse:Zustand .....	26
Fahrweg:NC-Programm:Rest .....	88
Fahrweg:NC-Satz:Rest .....	27
Offset:Gesamt .....	23
Offset:Handbetrieb .....	23
Position:Ist .....	22
Position:Soll .....	22
Position:Ziel .....	22
Referenzpunktfahrt erfolgt .....	27
Index .....	108
Nachricht an/von .....	208
Nachricht an/von:Inhalt .....	209
Nachricht an/von:Typ .....	209
Reset:Achse .....	128
Reset:Kanal .....	129
vorhanden:achs-spezifisch .....	188
vorhanden:kanal-spezifisch .....	186
Achse, in .....	28
Ist:ACS .....	24
Ist:PCS .....	22
Messwert:Achse .....	58
Offset:PCS:Gesamt .....	23
Offset:PCS:Handbetrieb .....	23
offset:physical, logical axis .....	25
Satzvorlauf:Fortsetzung .....	95
Soll:ACS .....	24
Soll:MCS:Werkzeugmittelpunkt .....	25
Soll:PCS .....	22
Ziel:ACS .....	24
Ziel:PCS .....	22
Spindel:M19 .....	144
erreicht .....	89, 90
H-Funktion .....	143
M-Funktion .....	143

## Q

Quittierung

Technologiefunktion .....	92
---------------------------	----

## R

Referenznocken .....	48
Referenzposition .....	56
Reaktionsklasse	
Reduktion	
Referenzpunktfahrt	
Regelfenster	
Reglerfreigabe	

Reset

Restfahrweg

Restweg

Retriggern Tastendruck-Beginn-Ereignis

Rückwärts

Rumpftyp

bei Fehlermeldung .....	193
Geschwindigkeit:Achse .....	48
Geschwindigkeit:Achse, Zone 1 .....	49
Geschwindigkeit:Achse, Zone 2 .....	49
Geschwindigkeit:Kanal .....	121
Geschwindigkeit:Kanal, Zone 1 .....	121
Geschwindigkeit:Kanal, Zone 2 .....	122
Löschen .....	57
Übernahme .....	55
erfolgt:ACS .....	27
erfolgt:PCS .....	27
Achse, im .....	28
Kommando .....	47
Zustand .....	29
PLC:Achse .....	128
PLC:Kanal .....	129
Spindel .....	80
NC-Satz:PCS .....	27
Spindel .....	79
verwerfen:Zustand .....	98
verwerfen:Ende-Marke .....	125
verwerfen:Kommando .....	124
Handbetrieb:Durchsetzung eines Tastendrucks .....	177
fahren .....	126
bei Fehlermeldung .....	194

## S

Satznummer .....	30
Schleppfehler .....	27
simuliertes Vorwärtsfahren .....	127
Solldrehzahl .....	76
Speicher zum Rückwärtsfahren zurücksetzen .....	127
Spindelfehler .....	81
Spindelreset .....	80
Spindelsteigungsfehler .....	39
Satzvorlauf	
Satzvorlauf an/aus	
Satzvorschub	
Schwesterwerkzeug	
S-Funktion	
Simulation	
Skip mode	
Sollposition	
Sollwerte	
Spindel	
Spindelbeauftragung	
Spindeldrehrichtung	
Stopp	
Synchronisation	

Bahn:Bewegung, aktuell.....	96
H-Funktion .....	142
M-Funktion .....	142
Bahnvorbereitung aktiv.....	94
Decoder aktiv .....	94
Interpolator aktiv .....	94
Interpolator:Abstand:Fortsetzposition.....	95
Interpolator:Zustand .....	95
an PLC .....	131
programmiert .....	82
gültig.....	147
Nummer.....	147
Achsnummer .....	145
Ausführungsdauer, voraussichtlich .....	144
Daten .....	144
Drehzahl.....	144
Nummer:Spindelpositionierfunktion.....	145
Nummer:Spindelschaltfunktion.....	144
Sollwert:Position .....	144
Bearbeitung EIN/AUS.....	114
NC block.....	104
erreicht .....	78
ACS .....	24
MCS:Werkzeugmittelpunkt.....	25
PCS .....	22
externe:Achse .....	72
Achse:ACS .....	21
Auftragsfehler .....	79
Beauftragung:extern .....	82
Betriebszustand.....	77
Drehrichtung .....	85
Drehzahl:programmiert.....	144
Positionssollwert:M19.....	144
Restfahrweg .....	79
steht.....	78
Stopp:Programm:Ende.....	80
extern .....	82
Typ .....	84
Typ .....	85
Bahn:Grund .....	90
Handbetriebsoffset:zurück fahren .....	53
NC-Programm:Dekodierung.....	106
Spindel:Programm:Ende .....	80
satzübergreifend:Technologiefunktion:Kanal ...	136
satzweise:Technologiefunktion:Kanal .....	135
Technologiefunktion:satzübergreifend:Achse ..	134
Technologiefunktion:satzweise:Achse .....	134

## T

Temperatureinfluss.....	40
Tangente	
Tastendruck Beginn/Ende	
Tastenummer	
Technologiefunktion	
T-Funktion	
Tippbetrieb	
Tippgeschwindigkeit	
TwinCAT	
Typ	

Bahn .....	101
Handbetrieb:Durchsetzung eines Tastendrucks .....	176
Handbetrieb:Durchsetzung eines Tastendrucks .....	176
Handbetrieb:Eilganggeschwindigkeit während Tastendruck.....	179
Achse:Daten .....	137
Anzahl.....	134, 135, 136
Anzahl:nicht quittiert:satzübergreifend .....	143, 145, 146
H-Funktion .....	142
Kanal:Daten.....	139
M-Funktion.....	142
Quittierung .....	92
S-Funktion .....	144
Synchronisation:satzübergreifend:Achse .....	134
Synchronisation:satzübergreifend:Kanal .....	136
Synchronisation:satzweise:Achse .....	134
Synchronisation:satzweise:Kanal .....	135
T-Funktion .....	146
Werkzeug:Identifikation .....	146
Werkzeugeinheit:Information .....	146
Daten .....	146
Achsnummer .....	169
Geschwindigkeit .....	169
Parametrierung.....	168
Verfahrgeschwindigkeit .....	162
Handbetrieb .....	169
Box .....	42
Gerät:Identifikation .....	42
Nachricht:an/von PLC.....	209

## U

Umkehrspanne .....	40
Updatezyklus	
HLI:achsspezifisch.....	189
HLI:kanalspezifisch.....	187

## V

V.E.-Variable .....	148
Vorsteuerung .....	40
Variable	
Variantenwerkzeug	
Vektor	
Verfahrgeschwindigkeit	
Verfahrweg	
Verschleiß	
Versionsbezeichnung CNC	
Versionskennung	
Verweilzeit	
Vorschub	
Vorschubfreigabe	
Vorschubstopp EIN/AUS	

extern .....	148
gültig .....	147
Nummer .....	147
Bahn:Tangente .....	102
Bahn:Tangente:Normal .....	102
Fläche:Normal .....	102
Werkzeug:Richtung .....	101
Jogbetrieb .....	162
Tippbetrieb .....	162
Jogbetrieb .....	163
Werkzeug:Radius .....	126
bei Fehlermeldung .....	193
CNC ® PLC .....	186
PLC ® CNC .....	185
aktiv .....	93
Bahn:Grenzwert unterschritten .....	95
Override:Achse .....	45
Override:Achse:gültig .....	46
Override:Kanal .....	114
Achse .....	43
Achse .....	44
Kanal .....	109

Zeile:NC-Programm .....	88
HLI .....	210
Nummer:Fehlermeldung .....	190
Nummer:Fehlermeldung:Dienstofffunktion .....	191
Zähler:NC-Programm .....	88
Ecke .....	99
bei Fehlermeldung:Interruptzyklen .....	192
bei Fehlermeldung:Interruptzyklen seit Systemstart .....	192
Bewegung:G01/G02 .....	98
gültig .....	122
ACS .....	24
PCS .....	22
Spindel .....	77
Betriebsart:Betriebsartwechsel .....	156
H-Funktion .....	143
M-Funktion .....	143
Achse:Interpolator .....	31
Antrieb .....	36
Handbetrieb .....	161
:Betriebsart .....	158

## W

Wahlweiser Halt .....	113
Wartezeit	
Watchdog	
Werkzeug	
Winkel	
Ecke .....	99
CNC überwacht PLC .....	181
PLC überwacht CNC .....	183
Anzahl:nicht quittiert:satzübergreifend .....	146
Einheit:Information .....	146
Geschwindigkeit:minimale:Ignorierung .....	51
gültig:Schwesterwerkzeug .....	147
gültig:Variantenwerkzeug .....	147
Identifikation .....	146
Korrektur:Online:Radius .....	126
Nummer:einzuwechselndes .....	147
Nummer:Schwesterwerkzeug .....	147
Nummer:Variantenwerkzeug .....	147
Orientierung .....	101
Richtung:Vektor: .....	101
Standgröße:Erfassung, ausschalten .....	124
aktiver/folgender NC-Satz .....	100

## Z

Zeitoverride .....	123
Zähler	
Zeiger	
Zeile	
Zeit	
Zeitangabe	
Zielposition	
Zielzustand	
Zusatzinformation	
Zustand	
Zustands-Istwert	



## 16 Appendix

### 16.1 Suggestions, corrections and the latest documentation

Did you find any errors? Do you have any suggestions or constructive criticism? Then please contact us at [documentation@isg-stuttgart.de](mailto:documentation@isg-stuttgart.de). The latest documentation is posted in our Online Help (DE/EN):



QR code link: <https://www.isg-stuttgart.de/documentation-kernel/>

The link above forwards you to:

<https://www.isg-stuttgart.de/fileadmin/kernel/kernel-html/index.html>



#### Notice

##### Change options for favourite links in your browser;

Technical changes to the website layout concerning folder paths or a change in the HTML framework and therefore the link structure cannot be excluded.

We recommend you to save the above "QR code link" as your primary favourite link.

##### PDFs for download:

DE:

<https://www.isg-stuttgart.de/produkte/softwareprodukte/isg-kernel/dokumente-und-downloads>

EN:

<https://www.isg-stuttgart.de/en/products/softwareproducts/isg-kernel/documents-and-downloads>

**E-Mail:** [documentation@isg-stuttgart.de](mailto:documentation@isg-stuttgart.de)





© Copyright  
ISG Industrielle Steuerungstechnik GmbH  
STEP, Gropiusplatz 10  
D-70563 Stuttgart  
All rights reserved  
[www.isg-stuttgart.de](http://www.isg-stuttgart.de)  
[support@isg-stuttgart.de](mailto:support@isg-stuttgart.de)

