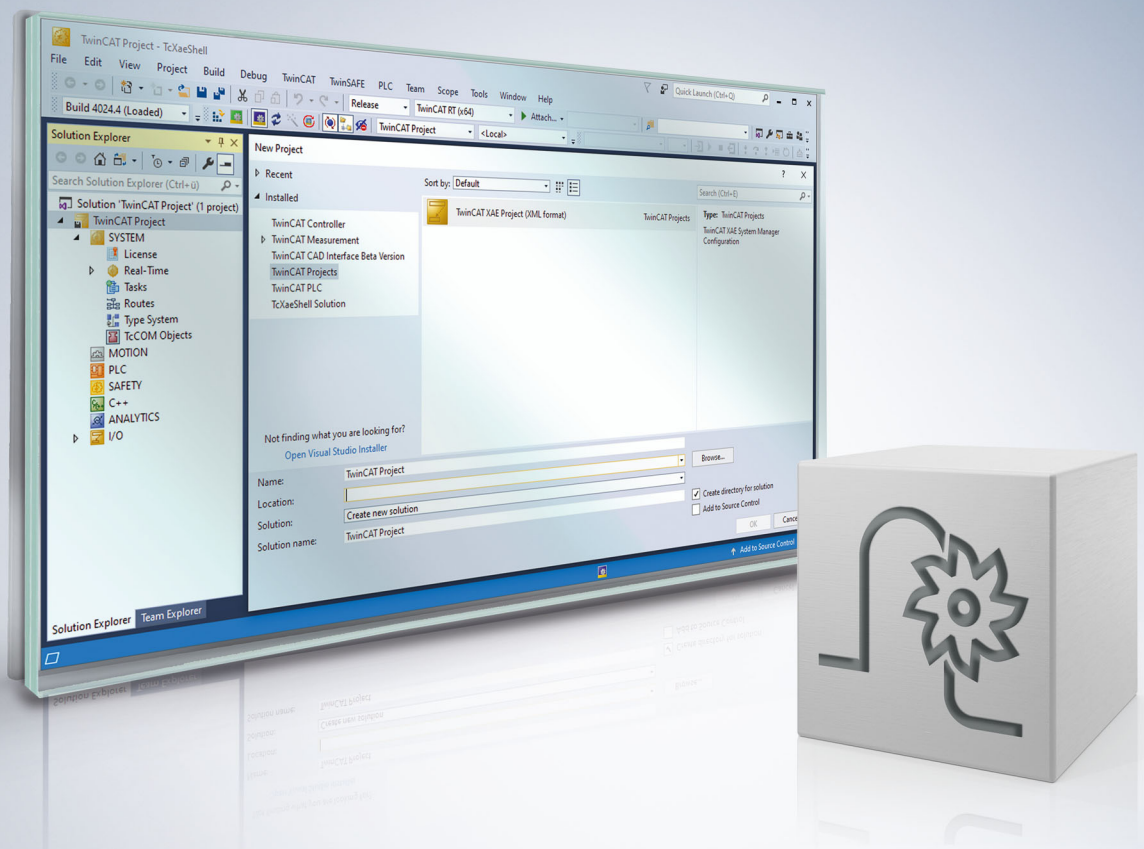
BECKHOFF New Automation Technology

Functional description | EN

TF5291 | TwinCAT 3 CNC

AM Plus



6/10/2025 | Version: 1.0

Notes on the documentation

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning the components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.
We reserve the right to revise and change the documentation at any time and without prior announcement.
No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH.
Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT technology is patent protected, in particular by the following applications and patents:
EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702
with corresponding applications or registrations in various other countries.

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

Copyright

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

1. 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.

NOTICE

Restriction or error

This icon describes restrictions or warns of errors.

Tips and other notes

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

General example

Example that clarifies the text.

NC programming example

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

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

List of figures

1 Estimation of future data

1.1 Overview

Task

When it is used in particular in additive manufacturing, a predictive control of the applying unit may help to compensate for dead times within the system and so improve the machining result.

This function is available as of CNC Build V3.1.3074.0.

Properties

Starting from the momentary point in time, the state at a specified future time is determined by a pre-calculation and then provided for use.

This function is an additional option requiring a license.

Parameterisation

The function must be activated by P-STUP-00070.
P-CHAN-00324 defines the point in time for the future state.

Programming

The points in time can also be defined by the NC command
#CHANNEL SET[ESA_TIME<i>=...] [▶ 10].

Mandatory note on references to other documents

For the sake of clarity, links to other documents and parameters are abbreviated, e.g. [PROG] for the Programming Manual or P-AXIS-00001 for an axis parameter.

For technical reasons, these links only function in the Online Help (HTML5, CHM) but not in pdf files since pdfs do not support cross-linking.

1.2 Description

The precalculation of future data elements function provides users with a prediction of future date elements at a configurable future point in time starting from the present point in time.

This function is activated by P-STUP-00070:

configuration.channel[0].interpolator.function FCT_DEFAULT | **FCT_CALC_STATE_AT_T**

Depending on the mode setting (P-CHAN-00325), the function permits the precalculation.

Mode 1: Precalculation of path velocities at up to 10 future points in time

Mode 2: In addition to the path velocities of Mode 1, the axis positions, velocities and accelerations of all axes located in the channel are precalculated in the first time entry.

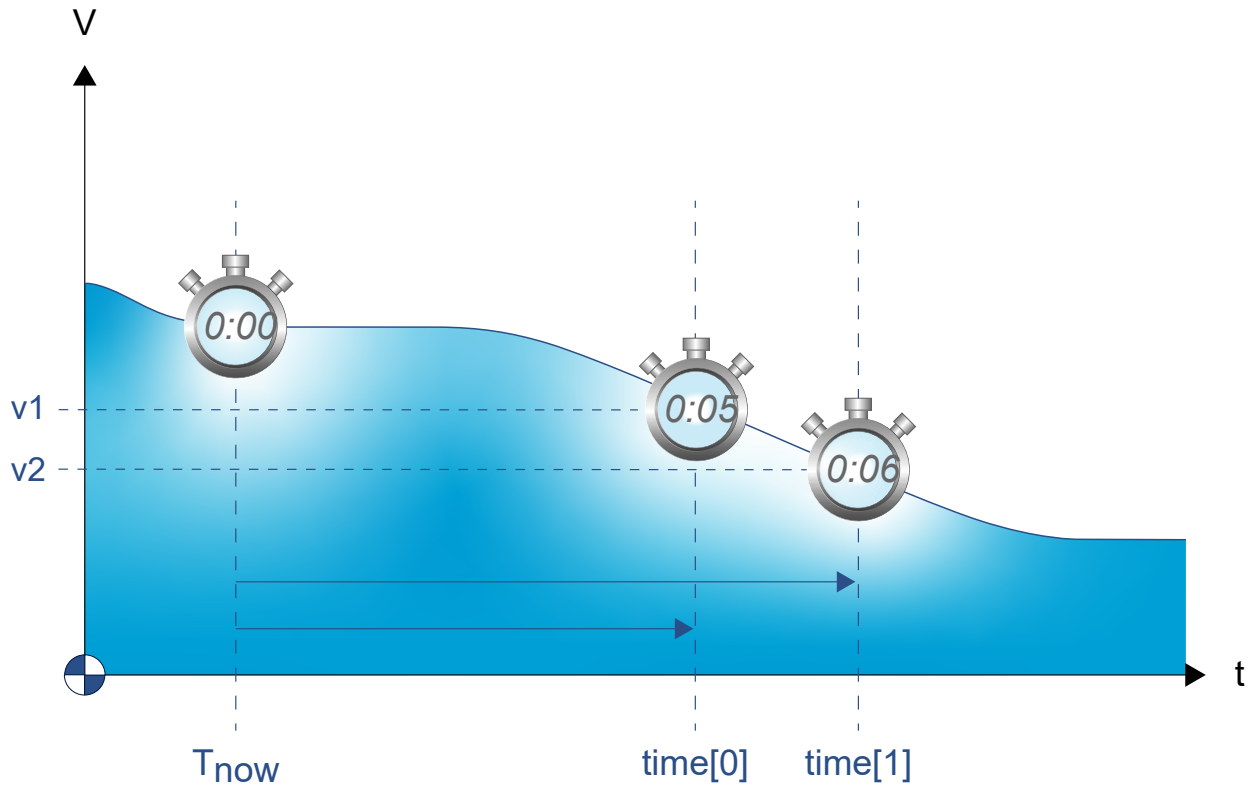


Fig. 1: Chronological sequence

Precalculation of path velocity

The results for future path velocities can be read by CNC objects [▶ 13] (ESA pathfeed).

Precalculation is limited to the look-ahead function. If no prediction is possible for path velocity planning (time offset too great), the value -1.0 is output.

If the result -1 is output for future path velocity, no value could be calculated.
Adapt the look-ahead buffer by P-STUP-00071

As of CNC Build V3.3104.08 future path velocities [▶ 18] can also be read over the HLI interface [▶ 16].
The data element esa_data_valid [▶ 18] indicates whether the future data element is valid.

Precalculation of an axis state

The results of these calculations are also contained in corresponding CHC objects to precalculate the axis position [▶ 14], velocity [▶ 14] or acceleration [▶ 14] of axes. Only the value of the first time entry is used to precalculate at axis level.

The first time entry is defined by P-CHAN-00324 [▶ 11] (esa.time[0]) or by #CHANNEL SET [ESA_TIME0=...] [▶ 10].

As of CNC Build V3.3104.08 future axis states can also be read over the HLI interface [▶ 16]. The data element esa_data_valid [▶ 16] indicates whether the future data element is valid.

Recommendation for look-ahead buffer

Setting for the available look-ahead buffer (P-STUP-00071):

configuration.channel[0].interpolator.number_blocks_lah 500

1.3 Programming

The offset time can also be defined in the NC program by the following command as an alternative to configuration in the channel parameter list with P-CHAN-00324 [▶ 11]:

```
Syntax:
```

#CHANNEL SET [ESA_TIME<i>**=..]**

ESA_TIME<i>=<*expr*> Offset time i in [s] where i = 0 ... 9. 10 ESA times (Estimated State of Arrival) can be defined. Only time values greater than 0 are considered.

Setting 3 ESA times

```
#CHANNEL SET [ESA_TIME0=0.3 ESA_TIME1=0.5 ESA_TIME2= 0.8]
```

1.4 Parameter

1.4.1 Overview

1.4.1.1 Start-up parameters

ID	Parameter	Description
P-STUP-00070	configuration.channel[i].interpolator.function	Define interpolator functionality
P-STUP-00071	configuration.channel[i].interpolator.number_blocks_lah	User-specific size of look-ahead buffer

1.4.1.2 Channel parameter

ID	Parameter	Description
P-CHAN-00324	esa.time[i]	Precalculation - time offset
P-CHAN-00325	esa.mode	Precalculation - mode

1.4.2 Description

1.4.2.1 Start-up parameters

P-STUP-00070	Definition of interpolator functionalities
Description	This parameter defines individual functionalities and the size of the look-ahead buffer in the interpolator, i.e. it defines the number of blocks to calculate deceleration distance and dynamic planning.
Parameter	configuration.channel[i].interpolator.function
Data type	STRING
Data range	See Description [▶ 33].
Dimension	----
Default value	FCT_IPO_DEFAULT
Remarks	

P-STUP-00071	User-specific size of look-ahead buffer
Description	This parameter permits the user-defined definition of the number of NC blocks in the look-ahead buffer. The parameter is only evaluated if P-STUP-00070 is set with FCT_LOOK_AHEAD_CUSTOM.
Parameter	configuration.channel[i].interpolator.number_blocks_lah *
Data type	UNS32
Data range	10 ... 10000
Dimension	----
Default value	120
Remarks	As of Build V2.11.20 and higher, the default size of the look-ahead buffer is 70 blocks. As of Build V2.11.28 and higher, the default size is 120 blocks. As the size increases, the additional calculations make greater demands on the controller hardware. As of Build V3.1.3067.07 the upper limit of the data range is 500 blocks. If #SLOPE[TYPE=STEP] is used, the upper limit is 10000 blocks as of Build V3.1.3060.0. * P-STUP-00071 in V2.11.20 and higher : configuration.channel[i].interpolator.parameter

1.4.2.2 Channel parameter

P-CHAN-00324	Default offset time to calculate future states
Description	At a set time greater than 0, the • path velocity • attempts to calculate axis position, velocity and acceleration at the parameterised point in the future.
Parameter	esa.time[i] where i = 0 ... 9
Data type	REAL64
Data range	0 ≤ time[i] ≤ MAX_REAL64
Dimension	s
Default value	0.0
Remarks	The maximum possible number of entries is limited to 10. Axis position, velocity and acceleration only estimated with the entry esa.time[0].

P-CHAN-00325	Precalculation mode

Description	This parameter sets the precalculation mode. • Mode 1: Precalculation of path velocity at up to 10 future points in time • Mode 2: in addition to Mode 1, the precalculation of axis positions, velocities and accelerations of all the axes in the channel is conducted for the first time entry
Parameter	esa.mode
Data type	UNS32
Data range	1 / 2
Dimension	----
Default value	1
Remarks	

1.4.3 CNC objects

1.4.3.1 Channel-specific CNC objects

Name	ESA: Active time [0]		
Description	First configured time at which feedrate is detected.		
Task	GEO (Port 551)		
Index group	0x12130<C_{ID}>	**Index offset**	0x112
Data type	REAL64	**Length**	8
Attributes	read	**Unit**	[s]
Remarks			

Name	ESA: Pathfeed [0]		
Description	Calculated pathfeed at first configured point in time. Determining the point of time: P-CHAN-00324 [▶ 11] (esa.time[0]) or #CHANNEL SET[ESA_TIME0 = <value>] [▶ 10]		
Task	GEO (Port 551)		
Index group	0x12130<C_{ID}>	**Index offset**	0x113
Data type	REAL64	**Length**	8
Attributes	read	**Unit**	[µm/s]
Remarks			

Additional points in time can be read similarly to the first point in time (Active time [0]/ ESA: Pathfeed [0]).

Point in time i	Offset ESA: Active time [i]	Offset ESA: Pathfeed [i]
0	0x112	0x113
1	0x114	0x115
2	0x116	0x117
3	0x12b	0x12c
4	0x12d	0x12e
5	0x12f	0x130
6	0x131	0x132
7	0x133	0x134
8	0x135	0x136
9	0x137	0x138

Points in time can also be specified in analogy, either using P-CHAN-00324 [▶ 11] (esa.time[i]) or #CHANNEL SET[ESA_TIME<i> = <value>]]

1.4.3.2 Axis-specific CNC objects

Name	ESA: position in future		
Description	Predicted axis position at defined point in time in the axis coordinate system (ACS) The point in time is defined by P-CHAN-00324 [▶ 11] (esa.time[0]) or by #CHANNEL SET [ESA_TIME0=<value>) [▶ 10]		
Task	GEO (Port 551)		
Index group	0x12130<C_{ID}>	**Index offset**	0x<A_{ID}>0087
Data type	REAL64	**Length**	8
Attributes	read	**Unit**	[0.1 µm or 0.0001°]
Remarks			

Name	ESA: velocity in future		
Description	Predicted axis velocity at defined point in time in axis coordinate system (ACS). The point in time is defined by P-CHAN-00324 [▶ 11] (esa.time[0]) or by #CHANNEL SET [ESA_TIME0=<value>) [▶ 10]		
Task	GEO (Port 551)		
Index group	0x12130<C_{ID}>	**Index offset**	0x<A_{ID}>0088
Data type	REAL64	**Length**	8
Attributes	read	**Unit**	[1µm/s or 0.001°/s]
Remarks			

Name	ESA: acceleration in future		
Description	Predicted axis acceleration at defined point in time in the axis coordinate system (ACS) The point in time is defined by P-CHAN-00324 [▶ 11] (esa.time[0]) or by #CHANNEL SET [ESA_TIME0=<value>) [▶ 10]		
Task	GEO (Port 551)		
Index group	0x12130<C_{ID}>	**Index offset**	0x<A_{ID}>0089
Data type	REAL64	**Length**	8
Attributes	read	**Unit**	[mm/s² or °/s²]
Remarks			

Name	ESA: position PCS in future		
Description	Predicted axis position at defined point in time without transformation or coordinate system in programming coordinate system (PCS) The point in time is defined by P-CHAN-00324 [▶ 11] (esa.time[0]) or by #CHANNEL SET [ESA_TIME0=<value>) [▶ 10]		
Task	GEO (Port 551)		
Index group	0x12130<C_{ID}>	**Index offset**	0x<A_{ID}>009A
Data type	REAL64	**Length/byte**	8
Attributes	read	**Unit**	[0.1 µm]
Remarks	Available as of CNC Build V3.1.3109		

Name	ESA: velocity PCS in future

Description	Predicted axis velocity at defined point in time without transformation or coordinate system in programming coordinate system (PCS). The point in time is defined by P-CHAN-00324 [▶ 11] (esa.time[0]) or by #CHANNEL SET [ESA_TIME0=<value>) [▶ 10]		
Task	GEO (Port 551)		
Index group	0x12130<C_{ID}>	**Index offset**	0x<A_{ID}>009B
Data type	REAL64	**Length/byte**	8
Attributes	read	**Unit**	[0.1 µm/s]
Remarks	Available as of CNC Build V3.1.3109		

Name	ESA: acceleration PCS in future		
Description	Predicted axis acceleration at defined point in time without transformation or coordinate system in programming coordinate system (PCS). The point in time is defined by P-CHAN-00324 [▶ 11] (esa.time[0]) or by #CHANNEL SET [ESA_TIME0=<value>) [▶ 10]		
Task	GEO (Port 551)		
Index group	0x12130<C_{ID}>	**Index offset**	0x<A_{ID}>009C
Data type	REAL64	**Length/byte**	8
Attributes	read	**Unit**	[mm/s²]
Remarks	Available as of CNC Build V3.1.3109		

1.4.4 HLI parameters

Connection to the HLI interface is available as of CNC Build V3.3104.08.

Precalculated axis-specific states

Precalculated data is valid, axis	
Description	This data element indicates whether precalculated axis data is valid. If the data element is TRUE, the values are valid for the precalculation of position [▶ 16], velocity [▶ 16] and acceleration [▶ 16] at a future point in time. FALSE indicates that no data could be calculated for the future point in time. The point in time is defined by the Index 0 in P-CHAN-00324 [▶ 11] or by #CHANNEL SET[ESA_TIME0=...] [▶ 10].
Signal flow	CNC →PLC
ST path	gpAx[*axis_idx*]^.ipo_state.**esa_data_valid**
Data type	BOOL
Value range	TRUE/FALSE
Access	PLC is reading
Special feature	**Available as of CNC Build V3.1.3104.08**

Precalculated position, axis	
Description	Axis position at a future point in time in the axis coordinate system (ACS). The point in time is defined by the Index 0 in P-CHAN-00324 [▶ 11] or by #CHANNEL SET[ESA_TIME0=...] [▶ 10].
Signal flow	CNC →PLC
ST path	gpAx[*axis_idx*]^.ipo_state.**esa_pos**
Data type	LREAL
Unit	[0.1 µm or 0.0001°]
Access	PLC is reading
Special feature	**Available as of CNC Build V3.1.3104.08**

Precalculated velocity, axis	
Description	Axis velocity at a future point in time in the axis coordinate system (ACS). The point in time is defined by the Index 0 in P-CHAN-00324 [▶ 11] or by #CHANNEL SET[ESA_TIME0=...] [▶ 10].
Signal flow	CNC →PLC
ST path	gpAx[*axis_idx*]^.ipo_state.**esa_vel**
Data type	LREAL
Unit	[1µm/s or 0.001°/s]
Access	PLC is reading
Special feature	**Available as of CNC Build V3.1.3104.08**

Precalculated acceleration, axis	
Description	Axis acceleration at a future point in time in the axis coordinate system (ACS). The point in time is defined by the Index 0 in P-CHAN-00324 [▶ 11] or by #CHANNEL SET[ESA_TIME0=...] [▶ 10].
Signal flow	CNC →PLC
ST path	gpAx[*axis_idx*]^. ipo_state.**esa_acc**

Data type	LREAL
Unit	[mm/s² or °/s²]
Access	PLC is reading
Special feature	**Available as of CNC Build V3.1.3104.08**

Precalculated position (PCS), axis	
Description	Axis position at a future point in time in the axis coordinate system (PCS). The point in time is defined by the Index 0 in P-CHAN-00324 [▶ 11] or by #CHANNEL SET[ESA_TIME0=...] [▶ 10].
Signal flow	CNC →PLC
ST path	gpAx[*axis_idx*]^.ipo_state.**esa_pos_pcs**
Data type	LREAL
Unit	[0.1 µm or 0.0001°]
Access	PLC is reading
Special feature	**Available as of CNC Build V3.1.3109**

Precalculated velocity (PCS), axis	
Description	Axis velocity at a future point in time in the axis coordinate system (PCS). The point in time is defined by the Index 0 in P-CHAN-00324 [▶ 11] or by #CHANNEL SET[ESA_TIME0=...] [▶ 10].
Signal flow	CNC →PLC
ST path	gpAx[*axis_idx*]^.ipo_state.**esa_vel_pcs**
Data type	LREAL
Unit	
Access	PLC is reading[1µm/s or 0.001°/s]
Special feature	**Available as of CNC Build V3.1.3109**

Precalculated acceleration (PCS), axis	
Description	Axis acceleration at a future point in time in the axis coordinate system (PCS). The point in time is defined by the Index 0 in P-CHAN-00324 [▶ 11] or by #CHANNEL SET[ESA_TIME0=...] [▶ 10].
Signal flow	CNC →PLC
ST path	gpAx[*axis_idx*]^. ipo_state.**esa_acc_pcs**
Data type	LREAL
Unit	[mm/s² or °/s²]
Access	PLC is reading
Special feature	**Available as of CNC Build V3.1.3109**

Precalculated axis-specific states

Up to 3 precalculated velocities can be read by the HLI interface.

Precalculated data is valid, path	
Description	If a data element in the field is TRUE, path velocity was precalculated for a future point in time and is therefore valid. This is indicated by the same index as precalculated velocity [▶ 18]. FALSE indicates that no value could be calculated for the future point in time. Several points in time were defined by P-CHAN-00324 [▶ 11]. Times can also be defined by #CHANNEL SET[ESA_TIME<i>=...] [▶ 10].
Signal flow	CNC →PLC
ST Path	gpCh[channel_idx]^.bahn_state.**esa_data_valid[ ]**
Data type	ARRAY[0..2] OF BOOL
Value range	TRUE/FALSE
Access	PLC is reading
Special feature	**Available as of Build V3.1.3104.08**

Precalculated velocity, path	
Description	Path velocity at a future point in time. Several points in time were defined by P-CHAN-00324 [▶ 11]. The index of a configured point in time corresponds with the index of the precalculated path velocity. Times can also be defined by #CHANNEL SET[ESA_TIME<i>=...] [▶ 10].
Signal flow	CNC →PLC
ST path	gpCh[channel_idx]^.bahn_state.**esa_vb[ ]**
Data type	ARRAY[0..2] OF LREAL
Unit	[µm/s]
Access	PLC is reading
Special feature	**Available as of Build V3.1.3104.08**

2 Contour look-ahead

2.1 Overview

Task

The “contour look-ahead” function of the CNC provides the PLC with programmed contour elements on the HLI in advance.

Effectiveness/possible applications

Process parameters, such as velocity, can be optimised based on the information about the future contour.

This function is available as of CNC Build V3.1.3107.10

This function is an additional option requiring a license.

Parameterisation

The “contour look-ahead” function must be configured with the following parameters:

- P-CHAN-00650 (alternatively P-STUP-00070) and
- P-CHAN-00658 (alternatively P-STUP-00076).

Programming

In order to command the CNC to provide future contour elements on the HLI,

- Program sequences can be marked using the commands #CONTOUR LOOKAHEAD LOG ON/OFF [▶ 29] or
- the PLC can send a request via the control unit [▶ 26].

Mandatory note on references to other documents

For the sake of clarity, links to other documents and parameters are abbreviated, e.g. [PROG] for the Programming Manual or P-AXIS-00001 for an axis parameter.

For technical reasons, these links only function in the Online Help (HTML5, CHM) but not in pdf files since pdfs do not support cross-linking.

2.2 Description

When optimising processes, it can be and advantage to know future contour elements in advance. The CNC can supply them using the contour look-ahead function.

The PLC can use these elements to optimise the process parameters. Data provision is either programmed by NC commands or by request from the PLC.

The data provided for future contour elements are:

- Type of contour element: Linear block, circular block, polynomial block
- Block number
- Start position of the contour element

- Length of the contour element
- Swept angle, programmed and compensated radius
- Program line number
- Length of the scales geometry for die sinking EDM

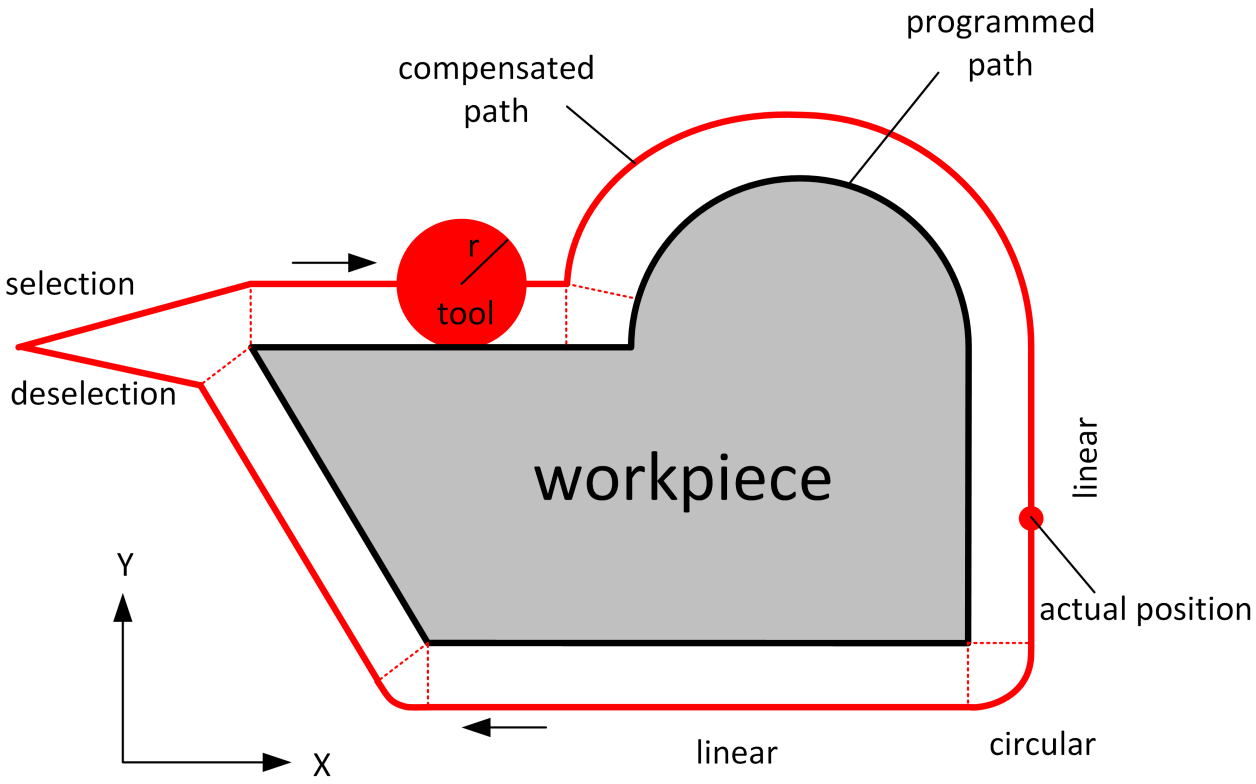


Fig. 2: Example of a programmed contour with tool radius compensation

Sequence of a PLC request

The PLC sends a request to the CNC. The request contains either an index that belongs to a motion block ore a distance based on the program start.

There are 2 modes for these two different requests:

- Mode 1: Request via the index [▶ 22].
- Mode 2: Request via distance from program start [▶ 24].

The CNC then sends the reply information to this request for each motion block. The process looks like this:

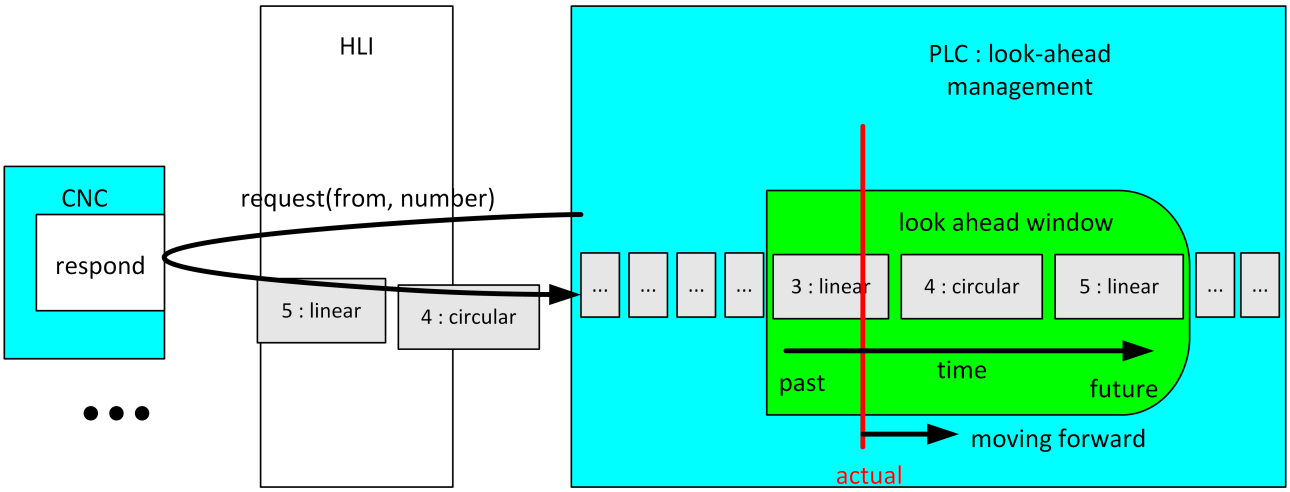


Fig. 3: Schematic diagram of a PLC request

2.2.1 Mode 1 – Request via index

This mode is available as of CNC Build V3.1.3105.01.

To use Mode 1, a suitable value must be set for the parameter P-STUP-00033 [▶ 32].

In this mode, the PLC reads the look-ahead buffer of NC blocks of the CNC previously marked with the command #CONTOUR LOOKAHEAD LOG ON/OFF [▶ 29].

The information is provided before these blocks are actually executed. The PLC can set synchronisation with any stop conditions (e.g. M functions) to ensure that the PLC has all the necessary data.

The command #CONTOUR LOOKAHEAD LOG [**PARAM=<val>**] is used to provide the PLC with an additional parameter value within a flagged area.

```
N100 #DISTANCE PROG START CLEAR
N110 X200
..
N200 #CONTOUR LOOKAHEAD LOG ON
N210       Y100
...
N500 #CONTOUR LOOKAHEAD LOG OFF
```

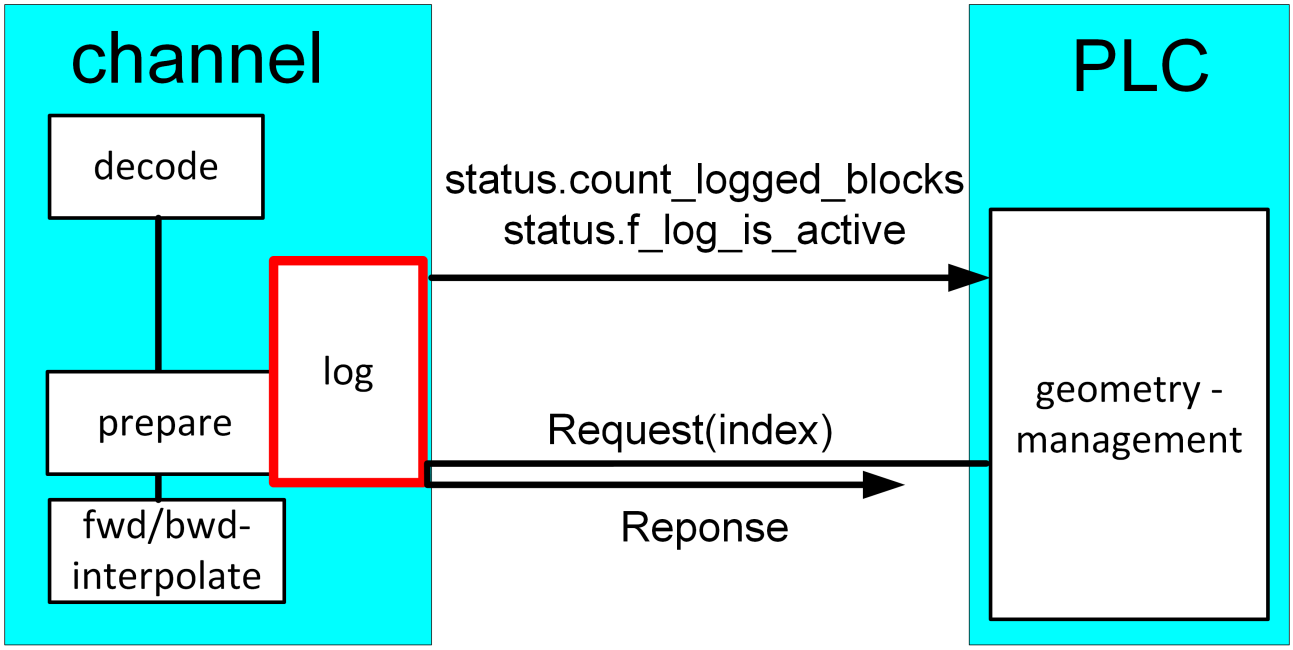


Fig. 4: PLC request via index

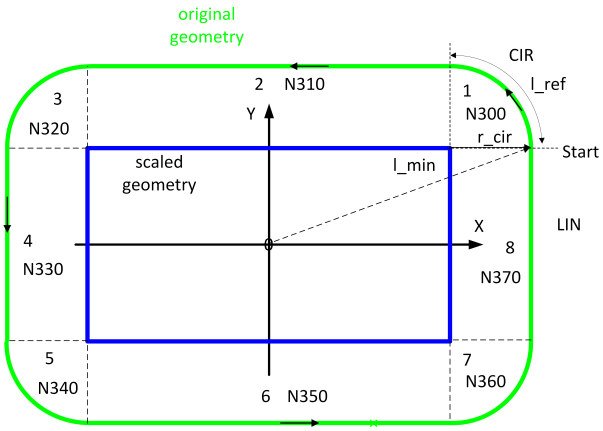


Fig. 5: Example of a logged programmed contour

In index mode, the individual elements can be requested multiple times, including backward motion on the path. The elements starting from the specified index are always output to the end.

2.2.2 Mode 2 - Request via distance from program start

This mode can only be used if P-CHAN-00658 [▶ 33] is parameterised.

Every contour element is identifiable by the path distance covered.

By default the distance starts at program start and is re-initialised with zero at every program start.

The distance can also be reset in the NC program with the NC command #DISTANCE PROGRAM START CLEAR.

While an NC program is in process, the distance covered is aggregated and displayed as the current distance.

The start position of a contour element ($CLAH_{start\_position}$) is defined as:

$$CLAH_{start\_position,n} = \sum_{k=0}^{n-1} block_{length_k}$$

If the PLC requests contour elements via distance, the correct distance must be specified for the blocks.

The figure below is a schematic diagram of a programmed contour using tool radius compensation (TRC) and the relationship between distance and $CLAH_{start\_position}$.

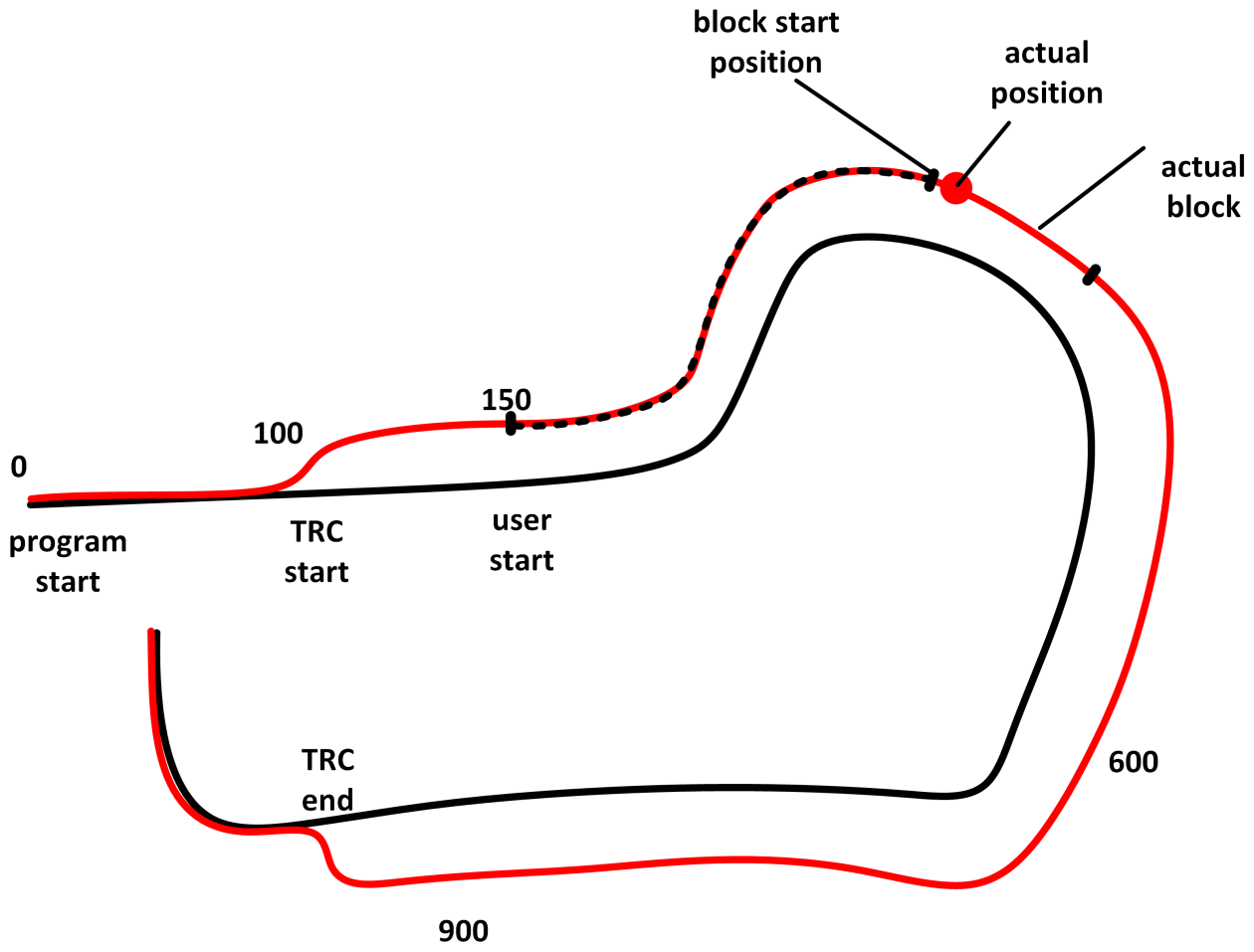


Fig. 6: Schematic diagram - distance and CLAH start_position

Properties of the look-ahead function:

- The maximum look-ahead distance available in the CNC is supplied by the HLI in each cycle and can be requested by a PLC. This information can be used to detect and request new available contour elements (start distance of the last element + length of the last element < maximum look-ahead distance).
- The maximum look-ahead distance depends on the internal buffer size of the interpolator and the block supply of the path preparation.
- The maximum look-ahead distance is independent of feedhold, override, synchronised M functions or G04.

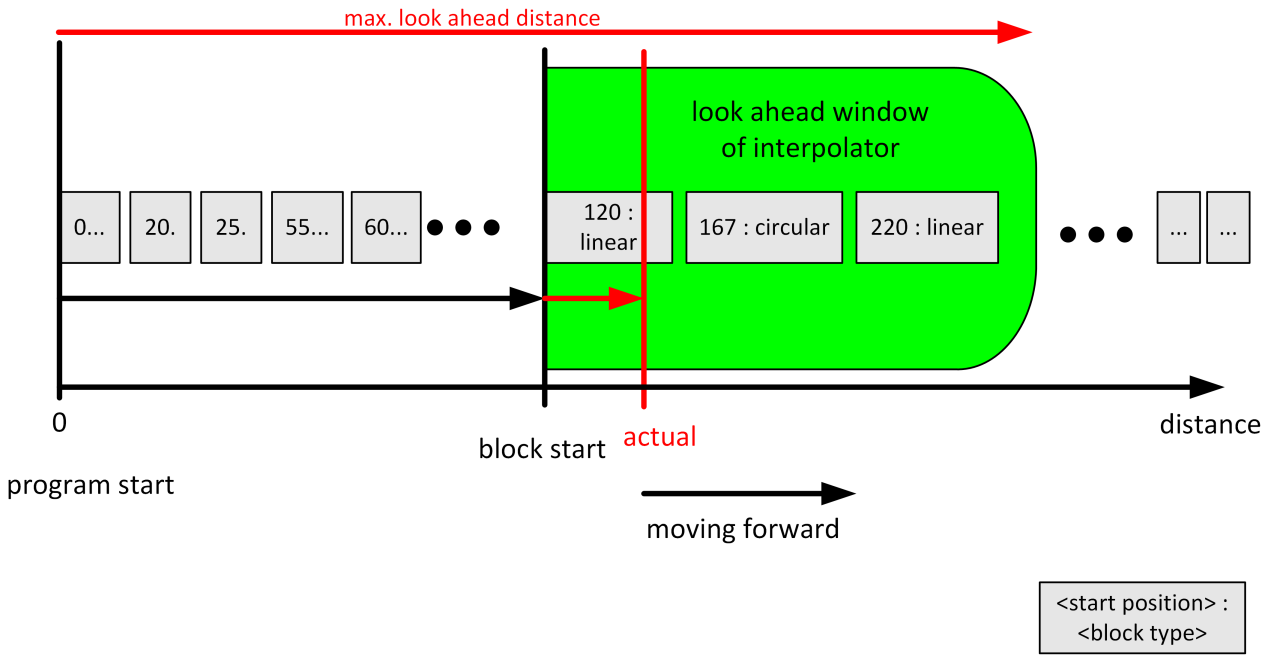


Fig. 7: Representation of the reply in PLC to distance request

2.2.3 PLC interface

The following data is available on the HLI to use the function:

1. The current distance from program start on the path corrected for tool radius compensation is displayed on the channel-specific HLI area (see bahn_state.dist_prog_start_high [▶ 34]). This distance can be used to request future contours in Mode 2.
2. The contour look-ahead control unit is provided to request and read out contour elements. The PLC requests information with increasing semaphore. The CNC supplies the information and deletes the semaphore afterwards.

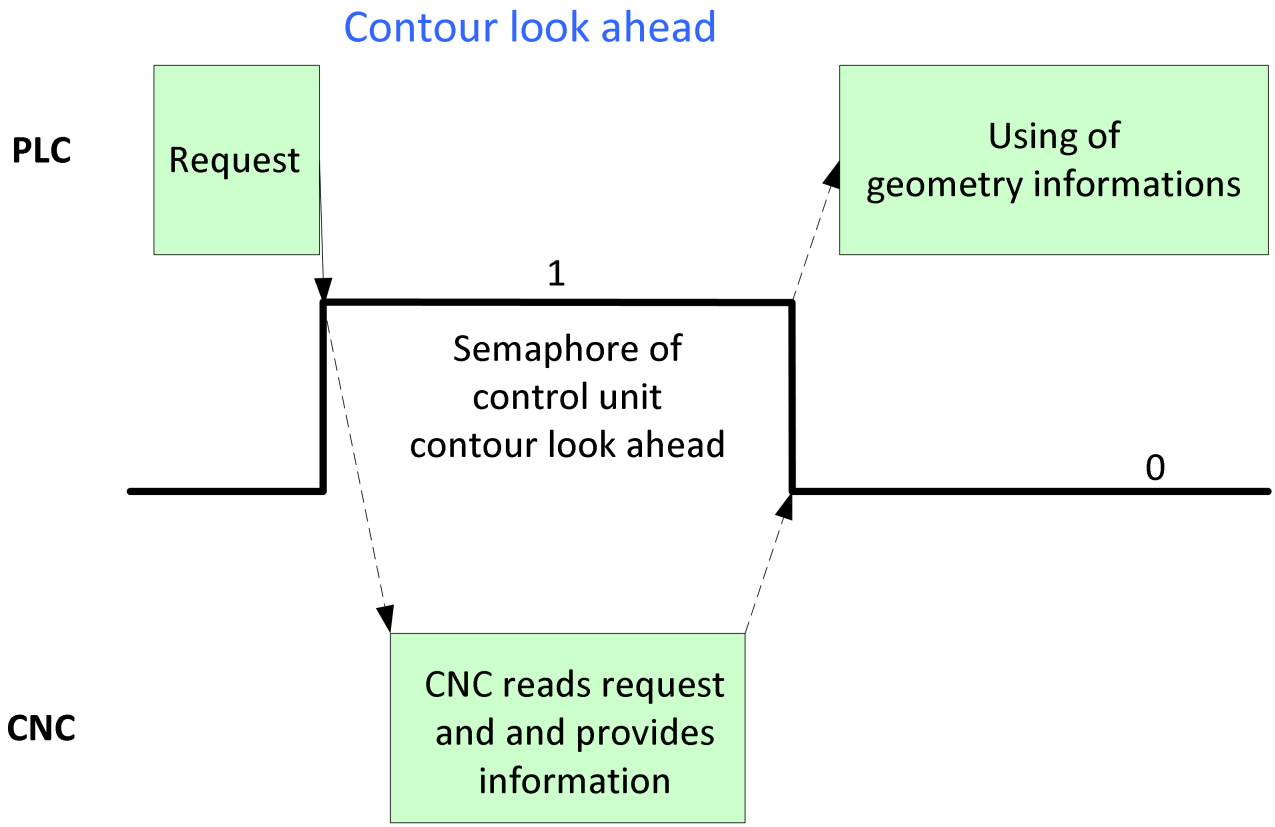


Fig. 8: Handshake - contour look-ahead control unit

Contour look-ahead	
Description	The PLC can use this control unit to request and read out information on contour elements from the CNC. The PLC can use this information to plan the look-ahead process.
Data type	HLI_CONTOUR_LOOKAHEAD
ST path	gpCh[*channel_idx*]^.**contour_lookahead**
Command data	
ST Element	**.request**
Data type	HLI_CONTOUR_LOOKAHEAD_REQUEST [▶ 27]
Access	PLC writes request
Return data	
ST element	**.response**
Data type	HLI_CONTOUR_LOOKAHEAD_RESPONSE [▶ 27]
Access	PLC reads
Flow control of commanded value	
ST element	**.semaphor_rw**

Data type	BOOL
Value range	[TRUE, FALSE]
Special features	**Consumption data item**
Access	TRUE : PLC triggers on new request FALSE : CNC has read new request
Status value	
ST Element	**.state**
Data type	HLI_CONTOUR_LOOKAHEAD_STATE [▶ 27]

2.2.3.1 User data

Contour look-ahead request data	
Description	Data to request the contour elements from the CNC
ST path	gpCh[*channel_idx*]^.*contour_lookahead.request*
ST name	HLI_CONTOUR_LOOKAHEAD_REQUEST
ST element	**.start_position**
Data type	LREAL
Description/ special features	The meaning of the element is dependent on the mode of the contour look-ahead function used. Mode 1: Index of the contour element in the marked area Mode 2: Start distance from which the distance of the “length” parameter applies
ST element	**.length**
Data type	LREAL
Description/ special features	A distinction is made whether mode 1 or 2 should be used depending on the entry of this element. Value = 0 : Request contour element by index, mode 1 Value != 0: Request contour element by distance, mode 2 This mode indicates the distance up to which contour element is to be requested

Contour look-ahead status data	
Description	Status data of the contour look-ahead control unit
ST Path	gpCh[*channel_idx*]^.contour_lookahead**.state**
ST name	HLI_INSERT_CMD_STATE
ST element	**.max_dist_prog_start**
Data type	LREAL
Description	Maximum look-ahead distance from program start in [0.1 µm].
ST element	**.count_logged_blocks**
Data type	UDINT
Description	Number of available contour elements that can be read out.
ST element	**.f_log_is_active**
Data type	BOOL
Description	This flag signals whether logging is still active for the marked area and all contour elements were read in. TRUE : not all contour elements were read in

Contour look-ahead return data	
Description	Contour information supplied by the CNC
ST Path	gpCh[*channel_idx*]^.contour_lookahead**. response**

ST name	HLI_CONTOUR_LOOKAHEAD_RESPONSE
ST Element	**.block[i]**
Data type	HLI_CONTOUR_LOOKAHEAD_BLOCK [▶ 28]
Description	

Structure HLI_CONTOUR_LOOKAHEAD_BLOCK	
Description	Information on a contour element that was supplied to the PLC by the CNC
ST Element	**.block_type**
Description	Block type 0 – no element exists 1 - linear contour element 2 - circular contour element 3 - #CONTOUR LOOKAHEAD LOG ON 4 - #CONTOUR LOOKAHEAD LOG [PARAM] 5 - #CONTOUR LOOKAHEAD LOG OFF 6 – Polynomial contour element
ST Element	**.block_number**
Description	Programmed NC block number
ST Element	**.block_count_r**
Description	Unique NC block ID number Corresponds to the displayed number of block_count_r in the Status information of a channel.
ST Element	**.start_position**
Description	Start distance of the contour element from program start
ST Element	**.length**
Description	Length of the contour element of the programmed value of the PARAM.
ST Element	**.programmed_radius**
Description	Programmed radius of the circular element
ST Element	**.compensated_radius**
Description	Compensated radius of the circular element
ST Element	**.circle_angle**
Description	Swept angle of the circular element. • Clockwise [-2*pi,0) • Counter-clockwise (0,2*pi].
ST Element	**.length_min**
Description	With die sinking, the length of the contour element is displayed on the scaled contour (radius = R_MAX_SCALE).
ST Element	**.tangent_variation**
Description	End angle to the previous motion block. (0,pi)

2.2.4 Examples (graphic)

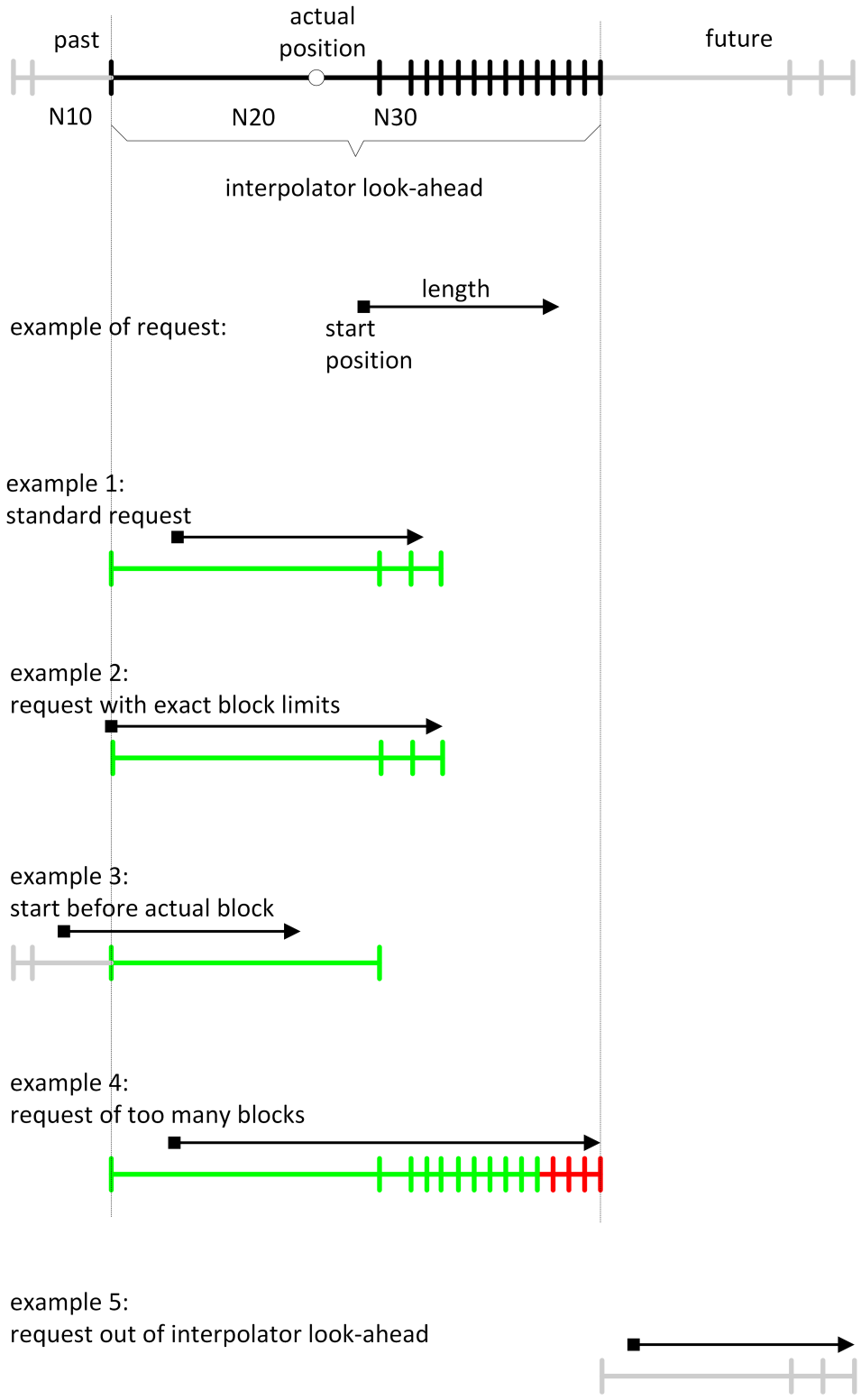


Fig. 9: Examples (graphic) of PLC requests

2.3 Programming

- Mode 1: Modifications required to the NC program.
- Mode 2: Only controlled by the PLC.

```
Syntax:
```

#CONTOUR LOOKAHEAD LOG [ON | OFF | CLEAR] [[PARAM=..]]

ON	Activate logging of contour elements by the NC command.
OFF	Deactivate logging of contour elements by the NC command.
CLEAR	The logging buffer is cleared at every program start. Use this NC command if the buffer needs to be cleared while the program is running to log a new area. Every time the buffer is cleared, decoding and execution are synchronised by an implicit #FLUSH&WAIT. Clearing can be executed while logging is active or inactive.
PARAM=..	This parameter allows the user to add an user-defined additional parameter (8 byte floating value) in the log entry.

If #CONTOUR LOOKAHEAD LOG ON is detected several times while a program is running, only the first marked area is provided. The command #CONTOUR LOOKAHEAD LOG CLEAR can be used to delete the last marked area. The next area is then supplied again with #CONTOUR LOOKAHEAD LOG ON.

2.3.1 Contour look-ahead and real-time loops

There are two programming options to obtain contour elements in a real-time loop (#RT WHILE/ ENDWHILE) in Mode 1:

1. Only one geometry is provided when programming #CONTOUR LOOKAHEAD LOG ON/OFF within a real-time loop. It has the properties of the entry loop, which means the parameter "tangent_variation [▶ 28]" has the angle to the motion block before the loop.
2. If #CONTOUR LOOKAHEAD LOG ON/OFF is programmed outside the real-time loop, the elements are supplied three times to obtain all the properties of the possible transitions. The transitions are:
   - start of loop
   - loop-loop
   - end of loop

NOTICE

Do not program the NC command #CONTOUR LOOKAHEAD CLEAR within a real-time loop.

If the command is programmed within the real-time loop, error ID 22073 is output.

Behaviour of contour look-ahead within a real-time loop

In this programming example, the contour look-ahead function is programmed within a real-time loop. The contour of the motion blocks 80 - 120 is logged once.

```
N060 #RT WHILE
N070 #CONTOUR LOOKAHEAD LOG ON [PARAM=1]
N080 G1 X4 Y4
…
N120 G1 X0 Y0
N130 #CONTOUR LOOKAHEAD LOG ON [PARAM=8]
N140 #RT ENDWHILE
```

Behaviour of contour look-ahead outside a real-time loop

In this programming example, the contour look-ahead is programmed outside a real-time loop. The contour of the motion blocks 80 to 120 is logged three times.

```
N060 #CONTOUR LOOKAHEAD LOG ON [PARAM=1]
N070 #RT WHILE
N080 G1 X4 Y4
…
N120 G1 X0 Y0
N130 #RT ENDWHILE
```

```
N140 #CONTOUR LOOKAHEAD LOG ON [PARAM=8]
```

2.4 Parameter

2.4.1 Overview

ID	start-up parameters	Description
P-STUP-00033	fb_storage_size[i]	Memory size for backward motion

ID	Channel parameters	Description
P-CHAN-00650	function	Activating functions (alternative to P-STUP-00070)
P-CHAN-00658	contour_lookahead_log_max	Maximum number of logged contour elements in the look-ahead area. (alternative P-STUP-00076)

2.4.2 Description

Start-up parameters

P-STUP-00033	Memory size for backward motion
Description	This parameter defines the memory size in bytes used for backward motion on the path. During start-up, the NC checks whether the required minimum size is available. If this is not the case, a warning is output and the memory size is set to the required minimum value. If the size is set to 0, the “forward/ backward motion on the path” function is not available. The maximum size is only limited by the resources available on the PC.
Parameter	fb_storage_size[i] where i = 0 to 11 (maximum number of channels: 12, application-specific)
Data type	UNS32
Data range	0 ... MAX(UNS32)
Dimension	----
Default value	0
Remarks	

Channel parameters

P-CHAN-00650	Definition of interpolator functionalities
Description	This parameter defines individual functionalities and the size of the look-ahead buffer in the interpolator, i.e. it defines the number of blocks required to calculate deceleration distance and dynamic planning.
Parameter	configuration.interpolator.function
Data type	STRING
Data range	See Description [▶ 33]
Dimension	----
Default value	FCT_IPO_DEFAULT
Remarks	

Interpolation function table

Identifier	Description
FCT_IPO_DEFAULT	FCT_LOOK_AHEAD_STANDARD
FCT_LOOK_AHEAD_LOW	30 blocks
FCT_LOOK_AHEAD_STANDARD	120 blocks
FCT_LOOK_AHEAD_HIGH	190 blocks
FCT_LOOK_AHEAD_CUSTOM	Number of look-ahead blocks in the interval [10; P-CHAN-00653].
FCT_SYNC	Synchronising an axis in coordinated motion Example: FCT_IPO_DEFAULT | FCT_SYNC
FCT_LOOK_AHEAD_OPT	The path velocity curve can be further improved for HSC machining by additional calculations. This generally reduces machining time. The additional calculations place greater demands on the controller hardware.
FCT_LIFT_UP_TIME	Automatic lifting/lowering of an axis (time-based coupling). Example: FCT_IPO_DEFAULT | FCT_LIFT_UP_TIME
FCT_SHIFT_NCBL	Path-controlled offset of M functions (dwell time). Example: FCT_IPO_DEFAULT | FCT_SHIFT_NCBL
FCT_CALC_STATE_AT_T	Calculation of path velocity at a time in the future. Function only available in combination with HSC slope and only as of V3.1.3057.0 Example: FCT_IPO_DEFAULT | FCT_CALC_STATE_AT_T
FCT_CALC_TIME	Calculation of interpolation time to next feed block (G01,G02,G03). Example: FCT_IPO_DEFAULT | FCT_CALC_TIME
FCT_CONTOUR_LAH	Contour look-ahead: advance output of motion blocks to the PLC as of V3.1.3104.07
FCT_DYN_POS_LIMIT	Dynamic limitation of axis positions
FCT_EXTENSION_EQUIDIST	Die-sinking EDM Orbiting
FCT_CALC_POS_V_0	Calculating and supplying the braking distance on the path until velocity and acceleration are 0. Supplying CNC objects on the path and assigned PCS and ACS axis positions at the end of this deceleration process. as of V3.01.3081.7 or V3.1.3114.0
FCT_DLM	Activate the conveyor tracking function. as of V4.20.0

The look-ahead buffer size values specified above apply as of CNC Builds V2.11.2800 and higher; the following settings apply to CNC Build V2.11.20xx:

FCT_LOOK_AHEAD_LOW	30 blocks
FCT_LOOK_AHEAD_STANDARD	70 blocks
FCT_LOOK_AHEAD_HIGH	120 blocks

P-CHAN-00658	Maximum number of logged contour elements in the contour look-ahead.

Description	This parameter can be used to set the maximum number of stored motion blocks that can be supplied to the PLC in advance.
	The CNC command #CONTOUR LOOKAHEAD LOG [] can be used to activate the save function.
	FCT_CONTOUR_LAH must be enabled in P-CHAN-00650 [▶ 32] for this functionality.
	`configuration.interpolator.fct_enable[0]`
	`FCT_IPO_DEFAULT|FCT_CONTOUR_LAH`

Parameter	configuration.interpolator.contour_lookahead_log_max
Data type	UNS32
Data range	0 <= contour_lookahead_log_max < MAX_UNS32

Dimension	----
Default value	128
Remarks	Parameter available as of V3.1.3107.10

2.4.3 PLC parameters

Currently covered path in the NC program (PCS)	
Description	Reads the current distance covered 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
ST path	gpCh[*channel_idx*]^.bahn_state.**dist_prog_start**
Data type	UDINT (* LREAL)
Unit	0.1 µm
Access	PLC is reading
Special features	* As of CNC Build V3.1.3104.01 the data element is provided in LREAL forma.

3 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Download finder

Our download finder contains all the files that we offer you for downloading. You will find application reports, technical documentation, technical drawings, configuration files and much more.

The downloads are available in various formats.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on our internet page: www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49 5246 963-157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49 5246 963-460
e-mail:	service@beckhoff.com

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49 5246 963-0
e-mail:	info@beckhoff.com
web:	www.beckhoff.com

Index

More Information:
www.beckhoff.com/TF5291

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com