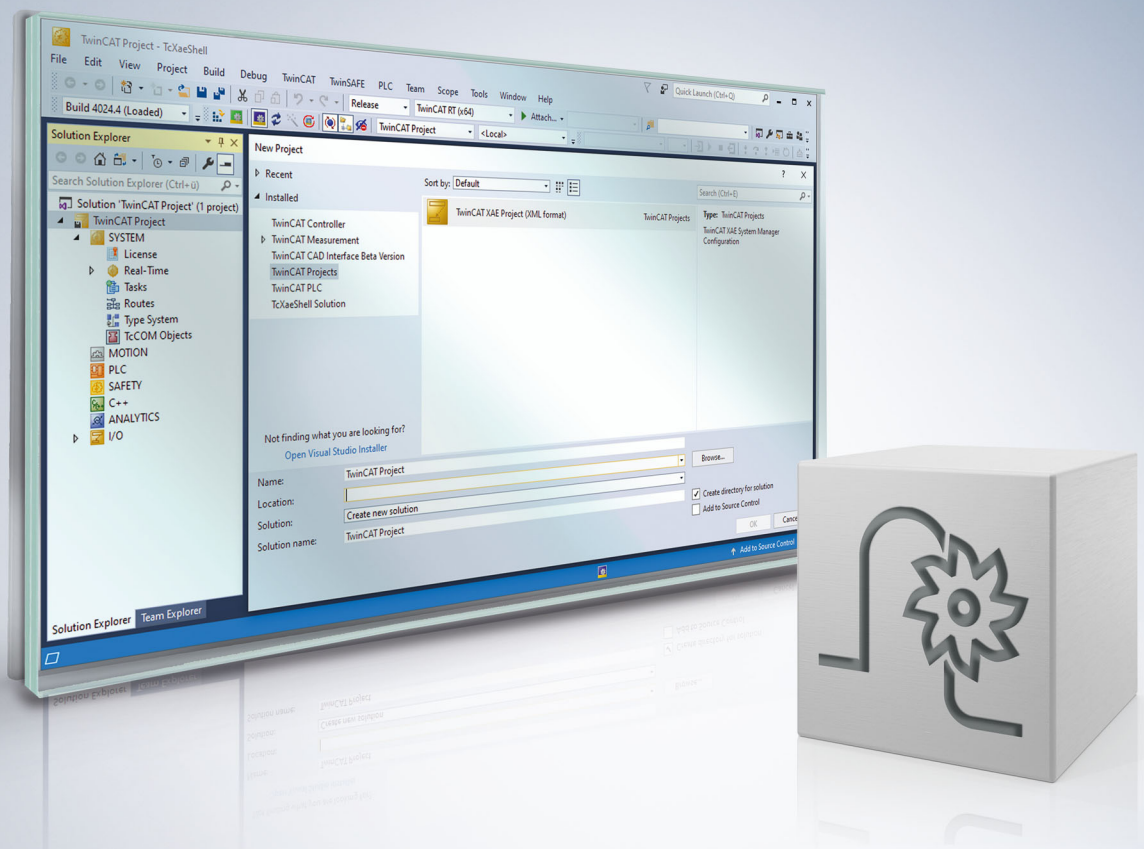
BECKHOFF New Automation Technology

Functional description | EN

TF5200 | TwinCAT 3 CNC

Contour visualization



12/18/2023 | Version: 1.16

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 Overview

Task

The controller can supply the axis positions for the graphic display of machine movements and visualise them by means of a user program or in the graphic user interface.

This can be executed as follows:

- additively to normal controller mode
- or simulatively without real axis movements.

Characteristics

Before start of the NC program, the execution mode must be switched to simulation to activate the simulation.

This is possible via:

- the user interface or
- the PLC interface

Parametrisation

To configure the above modes, a number of different parameters [▶ 40] must be assigned.

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 Description

The controller can supply the axis positions for the graphic display of machine movements and visualise them by means of a user program or in the graphic user interface.

In normal mode, axis positions are supplied in the CNC as display data in the interpolation cycle. To simplify visualisation, the volume of data supplied can be reduced by output of the relevant positions for visualisation, e.g. the exact end point of a contour element. Corners also remain identifiable as corners in the reduced visualisation data. The correct visualisation of corners is also possible if only very few points are declared if display is intended to be very rapid.

Different operation modes of contour visualisation

Dry run

In **Dry Run** mode, the NC program is decoded normally and the positions are interpreted. Axis motions are not forwarded to the position controller, meaning that there is no axis motion.

Rapid contour visualisation

The controller operates in simulation mode without real axis motion; the CNC program is processed rapidly. This function samples programmed contours and corners are all retained.

This considerably decreases the number of interpolation points for visualisation.

No real axis motion occurs.

Online contour visualisation

The controller operates in normal mode and CNC program execution is not affected. Position values are supplied to the contour visualisation interface in a coarser grid for visualisation.

Scene

The sequential kinematic chain is defined in the CNC program. A graphical object can be positioned in any coordinate system of the kinematic chain (LINKPOINT). Coordinate system movement is logged via an interface. The movements of graphical objects can be logged in kernelCAM or another system.

The **Scene** function is not available in TwinCAT.

The table below contains a comparison of modes:

Execution mode	Data reduction before interpolation	Data reduction after interpolation	Coordinate system of output data	Special features	Viewer
1. Dry run	- none -	- none -	PCS	Normal program execution without real axis motion	
2. Rapid contour visualisation	Geometric grid, abs./rel. secant error	No data reduction after interpolation; no interpolation points are generated if they do not lie on the visualisation grid.	WCS or ACS	possible without real axis motion. Rapid program execution	kernelCAM in preparation
3. Online contour visualisation	- none -	Geometric grid, abs./rel. secant error	WCS or ACS		kernelCAM in preparation
4. Scene	- none -	Time sampling in frames per second	MCS=W0 any point on the kinematic chain, also TCP	Available for any serial kinematics Kinematic chain must be initialised in the NC program	VirtuosV as vCAM

Coordinate systems

A number of different coordinate systems are available for individual interfaces. The following definition is used here:

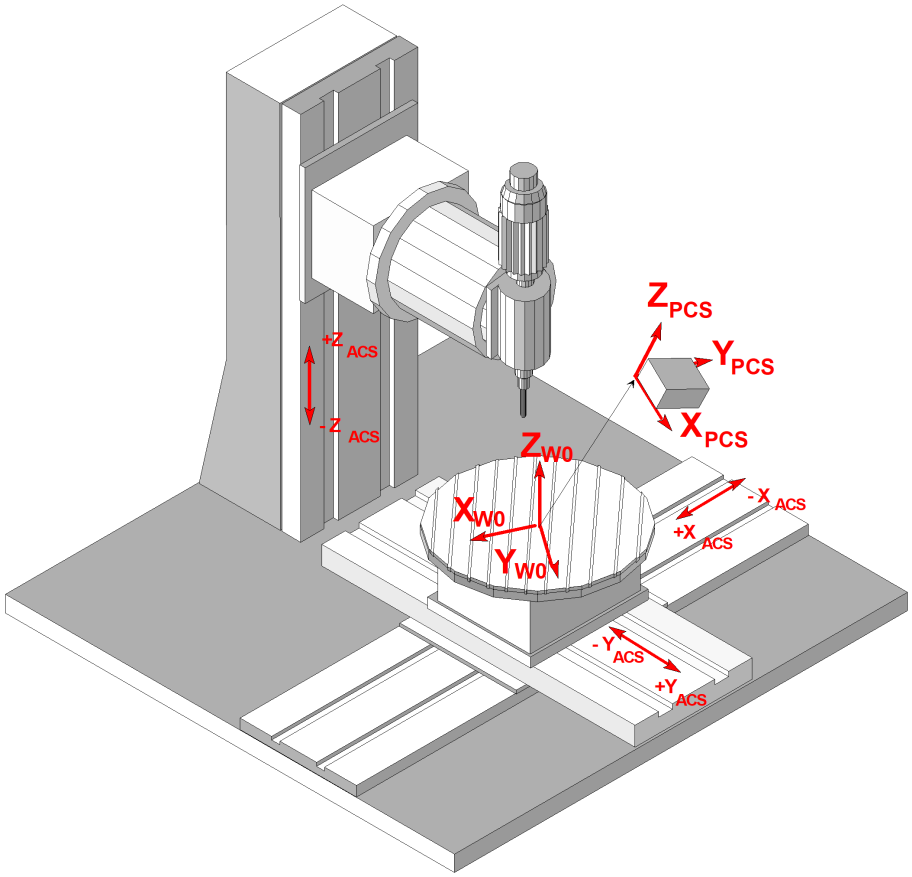


Fig. 1: Display of coordinate system used

ACS:	Axis Coordinate System
W0:	Base Workpiece Coordinate System, Cartesian base coordinate system of the machine referred to workpiece clamp position
PCS:	Programming coordinate system

3 Dry run

Dry Run is activated by transferring the program start option **0x40 MACHINE_LOCK** on the HLI to the controller at program start (see documentation on the HLI [▶ 19]).

In **Dry Run** mode, the NC program is decoded normally and the positions are interpreted. Axis motions are not forwarded to the position controller, meaning that there is no axis motion.

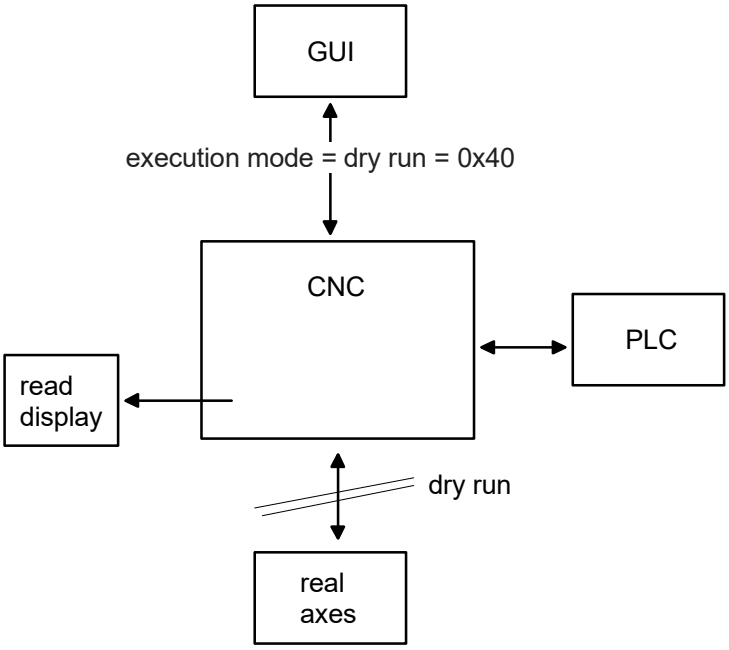


Fig. 2: Contour visualisation in Dry Run

When the operation mode is changed from normal mode to dry reun, all axes and spindles must be at standstill.

If this is not the case, the error ID 60269 is output when the spindle is selected.

4 Rapid contour visualisation

4.1 Description

Activation

Rapid contour visualisation is activated by transferring the program start option SOLLKON on the HLI to the controller at program start (see documentation on the HLI [▶ 19]).

No axis motions are executed with Rapid Contour Visualisation. Visualisation data is output in a reduced grid. The required interpolation point grid or the permitted secant error must be specified for the interpolation. The NC program is executed faster as a result of the sample grid.

Programmed dwell times (G04, #TIME) are ignored.

Applications

Simulation can be used for following applications, among others:

- “Syntax check“ using the entire CNC channel. As opposed to the syntax check mode, all modules in the NC channel are active during the simulation except for the position controller. This permits the detection of errors that are not detected during the syntax check, e.g. compensation motions during tool radius compensation or crossed software limit switches.
- Advance visualisation of an NC program (offline).

Sample grid

Depending on the motion block used (straight/curved), the interpolation point grid can be specified for the interpolation either

- by specifying a maximum interpolation point interval
- or by specifying a maximum path error

This can be defined in the following parameters:

Parameter	Format:	Description	Index-Group	Index-Offset
mc_contour_visu_grid_w mc_contour_visu_grid_r	UNS32	Output grid for nominal contour visualisation for linear blocks (G00/G01) in [0.1 µm]	0x2010<c> c element [1; max. channel]	0x89, 0x8a
mc_contour_rel_curv_error_w	REAL64	Maximum relative path error in [0.1%] for nominal contour visualisation of circles or polynomials	0x2010<c> c element [1; max. channel]	0x8b
mc_contour_abs_curv_error_w	REAL64	Maximum absolute path error in [0.1 µm] for nominal contour visualisation of circles and polynomials	0x2010<c> c element [1; max. channel]	0x8c

The target points of every NC block are always output.

Interpolation point grid for linear blocks

For linear blocks the interpolation point interval for interpolation is specified directly. As a consequence the axis dynamics and the programmed commanded velocity are not considered.

The programmed linear block is also output for each linear block if it does not lie on the set interpolation point grid. This means that the corners of a contour are always displayed.

If a linear block is shorter than the set interpolation point grid, the end point is not output.

Interpolation point grid for curved contour elements

An

- absolute secant error
- and a relative secant error

can be specified for curved contour elements (circles, polynomials).

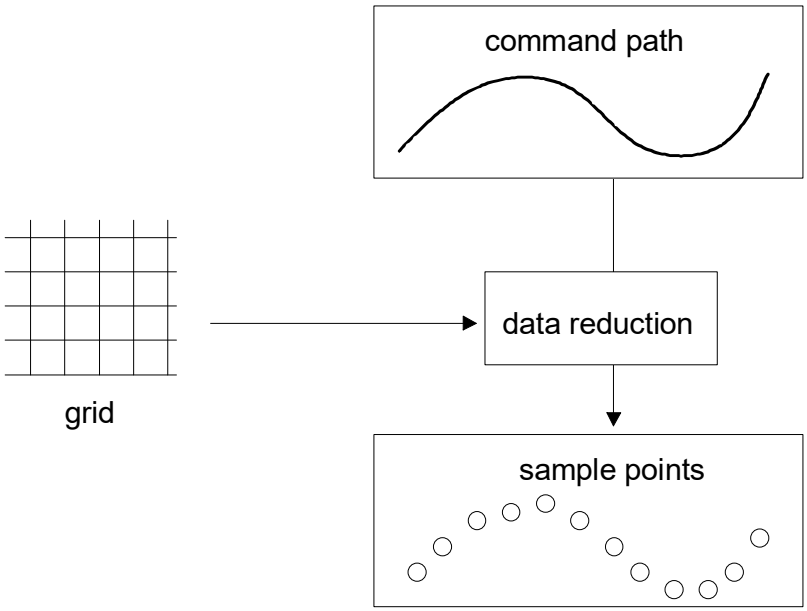


Fig. 3: Interpolation point grid for curved contour elements

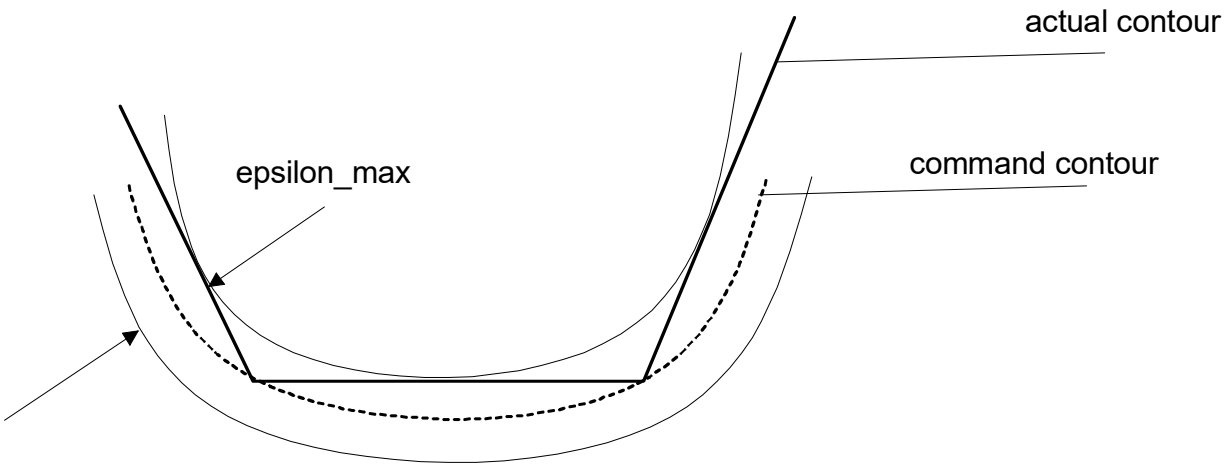


Fig. 4: Relative and absolute secant errors

$\varepsilon_{max} = r^*\varepsilon_{rel}$ for: $\varepsilon_{rel} <= \varepsilon_{abs}$

$\varepsilon_{max} = \varepsilon_{abs}$ for: $\varepsilon_{rel} >= \varepsilon_{abs}$

The resulting second error is the smaller of the two values.

Stop conditions

The execution of an NC program can be stopped by internal and external influences.

Internal stop conditions are NC commands which can only be terminated by user interaction. One example is a programmed stop (M00). The channel parameter P-CHAN-00183 prevents program execution from being stopped.

In case of external stop conditions, the user himself initiates the stop of an NC program execution. Examples include:

- Feedhold via the PLC interface
- Technology function not acknowledged

External stop conditions are always effective. The user must therefore make sure that program execution is not stopped.

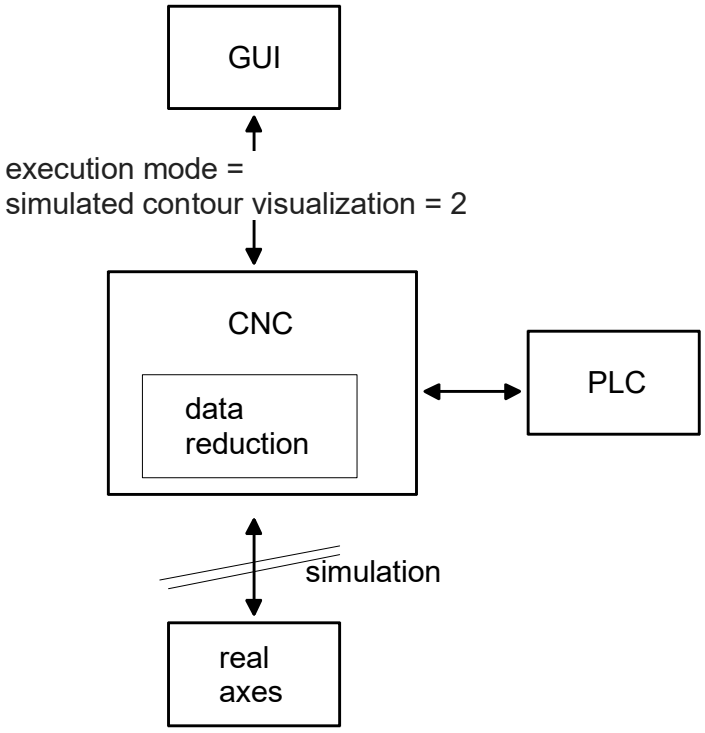


Fig. 5: Rapid contour visualisation

As opposed to Syntax Check [see functional description FCT-C9] it is not possible to continue program execution after an error occurs.

When an error occurs, a reset must be triggered in the channel and the program must be restarted after the error is rectified.

Output

Generated visualisation data can be read by CNC objects. Motion blocks are divided and axis positions are output depending on the grid set.

Axis positions can be output in 2 ways:

- Display of axis coordinates including offsets (machine coordinates)
- Display of absolute coordinates without offsets (programmed coordinates)

Select the data to be output in the Start-up parameters P-STUP-00039.

4.2 Interfacing

There are 2 options for interface connection (described below):

- Selection via HMI or via CNC objects
- Commands and display via HLI

Commands via CNC objects

Visualisation can be parameterised and visualisation data can be queried using CNC objects.

All object accesses are made using the COM task. The individual data/parameters are accessed via the following index groups/offsets.

For the index group, the channel number must be used for the placeholder <c>, whereby <c> lies within [1; max. number of channels].

Parameter	Format:	Description	Index-Group	Index-Offset
mc_command_execution_mode_r, mc_command_execution_mode_w	UNS32	Select nominal contour visualisation 0x0000 ISG_STANDARD Normal mode **0x0002 SOLLKON Nominal contour visualisation** 0x0004 ON_LINE Online-Visu 0x0008 SYNCHK Syntax check	0x2010<c> c element [1; max. channel]	0x40, 0x3f
mc_contour_visu_grid_w mc_contour_visu_grid_r	UNS32	Output grid for nominal contour visualisation for linear blocks (G00/G01) in [0.1 µm]	0x2010<c> c element [1; max. channel]	0x89, 0x8a
mc_contour_rel_curv_error_w	REAL64	Maximum relative path error in [0.1%] for nominal contour visualisation of circles or polynomials	0x2010<c> c element [1; max. channel]	0x8b
mc_contour_abs_curv_error_w	REAL64	Maximum absolute path error in [0.1 µm] for nominal contour visualisation of circles and polynomials	0x2010<c> c element [1; max. channel]	0x8c

For curved contour elements (circles, polynomials), the parameter that results in the smallest output grid is used.

Commands and display via HLI

Contour visualisation can be commanded and displayed via the HLI. The table below lists the parameters that must be assigned.

Channel mode			
Description	Selection of a special channel mode such as syntax check or machining time calculation		
Data type	MC_CONTROL_SGN32_UNIT, see description of Control Unit		
Access	PLC reads request_r + state_r and writes command_w + enable_w		
ST Path	gpCh[*channel_idx*]^.decoder_mc_control.**execution_mode**		
Commanded, requested and return values			
ST Element	**.command_w** **.request_r** **.state_r**		
Data type	DINT		
Value range	**Value**	**Constant**	**Meaning**
	0x0000	ISG_STANDARD	Normal mode
	0x0001	SOLLKON	Block search
	0x0002	SOLLKON	Nominal contour visualisation simulation with output of visualisation data
	0x0802	SOLLKON_SUPRESS_OUTPUT & 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
	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 [▶ 37]]). An additional client is linked to this output via DataFactory / CORBA.
	0x2000	SUPPRESS_TECHNO_OUTPUT	Without output of technology functions (M/H/T). Set implicitly in connection with syntax check.
Redirection			
ST element	**.enable_w**		

Output

The visualisation data can be read via CNC objects.

Indexgroup	**Index offset**	**Data type**	**Description**

0x2010<c>	0x2000	SOLLKONT_VISU_PDU	Data record from channel-specific output buffer (FIFO).
0x2010<c>	0x2001	UNS32	Number of data records in the channel-specific output FIFO
0x2010<c>	0x2002	SOLLKONT_VISU_PDU	Data record from global output FIFO
0x2010<c>	0x2003	UNS32	Number of data records in the global output FIFO

The data package read has the following structure:

	SOLLKONT_VISU_PDU
SGN32	count, number of structures SOLLKONT_VISU_DATA_V0 … SOLLKONT_VISU_DATA_V5 in the current message
UNS32	Version identifier of visualisation data P-STUP-00039
SOLLKONT_VISU_DATA_V0	v0[MAX_SOLLKONT_VISU_DATA_COUNT_V0 - 1] Structure with visualisation data if P-STUP-00039 has the value 0.
or SOLLKONT_VISU_DATA_V1	v1[MAX_SOLLKONT_VISU_DATA_COUNT_V1 - 1] Structure with visualisation data if P-STUP-00039 hat the value 1.
or SOLLKONT_VISU_DATA_V2	V2[MAX_SOLLKONT_VISU_DATA_COUNT_V2 - 1] Structure with visualisation data if P-STUP-00039 hat the value 2.
or SOLLKONT_VISU_DATA_V3	v3[MAX_SOLLKONT_VISU_DATA_COUNT_V3 - 1] Structure with visualisation data if P-STUP-00039 hat the value 3.
or SOLLKONT_VISU_DATA_V4	v4[MAX_SOLLKONT_VISU_DATA_COUNT_V4 - 1] Structure with visualisation data if P-STUP-00039 hat the value 4.
or SOLLKONT_VISU_DATA_V5	V5[MAX_SOLLKONT_VISU_DATA_COUNT_V5 - 1] Structure with visualisation data if P-STUP-00039 hat the value 5.
or SOLLKONT_VISU_DATA_V6	v6[MAX_SOLLKONT_VISU_DATA_COUNT_V6 - 1] Structure with visualisation data if P-STUP-00039 hat the value 6.
or SOLLKONT_VISU_DATA_V7	v7[MAX_SOLLKONT_VISU_DATA_COUNT_V7 - 1] Structure with visualisation data if P-STUP-00039 hat the value 7.
or SOLLKONT_VISU_DATA_V8	V8[MAX_SOLLKONT_VISU_DATA_COUNT_V8 - 1] Structure with visualisation data if P-STUP-00039 has the value 8.
or SOLLKONT_VISU_DATA_V9	V9[MAX_SOLLKONT_VISU_DATA_COUNT_V9 - 1] Structure with visualisation data if P-STUP-00039 has the value 9.
or SOLLKONT_VISU_DATA_V10	V10[MAX_SOLLKONT_VISU_DATA_COUNT_V10 - 1] Structure with visualisation data if P-STUP-00039 has the value 10.
or SOLLKONT_VISU_DATA_V11	V11[MAX_SOLLKONT_VISU_DATA_COUNT_V11 - 1] Structure with visualisation data if P-STUP-00039 has the value 11.

	SOLLKONT_VISU_DATA_V0
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
SOLLKONT_VISU_ACHS_DATA_STD	Simu_achs_data_std[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V1
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
IF_FILE_NAME	File_name

SOLLKONT_VISU_ACHS_DATA_STD	Simu_achs_data_std[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V2
SOLLKONT_VISU_CH_DATA_V1	Visu_data_v1
IF_FILE_NAME	Filename of the NC main program
SOLLKONT_VISU_ACHS_DATA_STD	Simu_achs_data_std[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V3
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
SOLLKONT_VISU_ACHS_DATA_V1	Simu_achs_data_v1[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V4
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
IF_FILE_NAME	Filename of the NC main program
SOLLKONT_VISU_ACHS_DATA_V1	Simu_achs_data_v1[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V5
SOLLKONT_VISU_CH_DATA_V1	Visu_data_v1
IF_FILE_NAME	Filename of the NC main program
SOLLKONT_VISU_ACHS_DATA_V1	Simu_achs_data_v1[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V6
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
SOLLKONT_VISU_ACHS_DATA_V2	Simu_achs_data_v2[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V7
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
IF_FILE_NAME	Filename of the NC main program
SOLLKONT_VISU_ACHS_DATA_V2	Simu_achs_data_v2[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V8
SOLLKONT_VISU_CH_DATA_V1	Visu_data_v1
IF_FILE_NAME	Filename of the NC main program
SOLLKONT_VISU_ACHS_DATA_V2	Simu_achs_data_v2[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V9

SOLLKONT_VISU_CH_DATA_V2	Visu_data_v2
IF_FILE_NAME	Filename of the NC main program
SOLLKONT_VISU_ACHS_DATA_STD	Simu_achs_data_std[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V10
SOLLKONT_VISU_CH_DATA_V2	Visu_data_v2
IF_FILE_NAME	Filename of the NC main program
SOLLKONT_VISU_ACHS_DATA_V1	Simu_achs_data_v1[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_DATA_V11
SOLLKONT_VISU_CH_DATA_V2	Visu_data_v2
IF_FILE_NAME	Filename of the NC main program
SOLLKONT_VISU_ACHS_DATA_V2	Simu_achs_data_v2[ANZ_SIMU_KOORD] Axis-specific visualisation data.

	SOLLKONT_VISU_CH_DATA_STD
SGN32	nc_satz_nr, block number in the NC program
SGN32	fileoffset, file offset from file start in bytes >= 0 : valid data offset when program is active == -1 : Offset not valid since no program is active
UNS16	channel_nr, channel number
SGN16	g_function >= 0 : G function : G0, G1, G2, G3, G61 for polynomial blocks == -1 : no G function active
UNS32	circle_radius, radius in [0.1 µm] for G2 / G3 blocks
REAL64	circle_center_point[2] Absolute position of circle centre point in the active machining plane (G17,G18,G19) in [0.1 µm] for G2 / G3 blocks (as of CNC Build V2.10.1032.03 and V2.10.1505.05)

	SOLLKONT_VISU_CH_DATA_V2
SGN32	Nc_satz_nr, block number in the NC program
SGN32	fileoffset, file offset from file start in bytes >= 0 : valid data offset when program is active == -1 : Offset not valid since no program is active
UNS16	channel_nr, channel number
SGN16	g_function >= 0 : G function : G0, G1, G2, G3, G61 for polynomial blocks == -1 : no G function active
UNS32	circle_radius, radius in [0.1 µm] for G2 / G3 blocks
REAL64	circle_center_point[2] Absolute position of circle centre point in the active machining plane (G17,G18,G19) in [0.1 µm] for G2 / G3 blocks (as of CNC Build V2.10.1032.03 and V2.10.1505.05)
SGN32	vb_prog, programmed path velocity

SOLLKONT_VISU_DATA_TECHNO_V1	Techno_v1, technology information
UNS32	Fillup (alignment data)

	SOLLKONT_VISU_DATA_TECHNO_V1
UNS16	Axis_number, axis number of the axis to which technology information was output. An axis number of 0 means that technology information was output to the channel interface
UNS16	Fillup, used to force structure alignment
UNS32	m_h_count, number of assigned entries in the vector m_h_data[]
SOLLKONT_M_H_PROCESS_V1	M_h_data_v1[MAX_M_H_DATA] Vector containing information about M/H functions
UNS32	S_count, number of entries in the vector s_proc[]
SOLLKONT_S_PROCESS	s_proc[] Vector containing information about spindle functions
SGN32	vb_prog, programmed path velocity

Data offset indicates whether a program is being edited or was terminated.

An invalid G function (-1) is triggered by an NC line containing an M function, for example.

	IF_FILE_NAME
ISG_CHAR	file_name[128] Filename of the current NC program. To obtain the additional output of the filename, the version identifier of the display data “contour_visu_ifc_version” (P-STUP-00039) must be set to the value 1, 2, 4, or 5 (as of CNC Build V2.10.1032.08 and V2.10.1507. 06).

	SOLLKONT_VISU_ACHS_DATA_STD
SGN32	Akt_sollwert, current command position of the axis in [0.1 µm]
UNS16	Log_achs_nr, logical axis number
UNS16	<alignment bytes>

	SOLLKONT_VISU_ACHS_DATA_V1
SGN32	Akt_sollwert, current command position of the axis in [0.1 µm]
SGN32	Akt_sollwert_wcs0, current command position of the axis in the WCS0 system in [0.1 µm]. This value is only calculated if channel parameter P-CHAN-00145 has the value 1 and the channel parameter P-CHAN-00032 is assigned a value > 0.
UNS16	Log_achs_nr, logical axis number
UNS16	<alignment bytes>

	SOLLKONT_VISU_ACHS_DATA_V2
SGN32	Akt_sollwert, current command position of the axis in [0.1 µm]
SGN32	Akt_sollwert_wcs0, current command position of the axis in the WCS0 system in [0.1 µm]. This value is only calculated if channel parameter P-CHAN-00145 has the value 1 and the channel parameter P-CHAN-00032 is assigned a value > 0.
SGN32	Akt_sollwert_wcs, current command position of the axis in the WCS system in [0.1 µm].

	This value is only calculated if channel parameter P-CHAN-00145 has the value 1 and the channel parameter P-CHAN-00032 is assigned a value > 0.
UNS16	Log_achs_nr, logical axis number
UNS16	<alignment bytes>

	SOLLKONT_VISU_CH_DATA_V1
SGN32	Nc_satz_nr, block number in the NC program
SGN32	fileoffset, file offset from file start in bytes >= 0 : valid data offset when program is active == -1 : Offset not valid since no program is active
UNS16	channel_nr, channel number
SGN16	g_function >= 0 : G function : G0, G1, G2, G3, G61 for polynomial blocks == -1 : no G function active
UNS32	circle_radius, radius in [0.1 µm] for G2 / G3 blocks
REAL64	circle_center_point[2] Absolute position of circle centre point in the active machining plane (G17,G18,G19) in [0.1 µm] for G2 / G3 blocks (as of CNC Build V2.10.1032.03 and V2.10.1505.05)
SGN32	vb_prog, programmed path velocity
SOLLKONT_VISU_DATA_TECHNO	techno, technology information

	SOLLKONT_VISU_DATA_TECHNO
UNS16	Axis_number, axis number of the axis to which technology information was output. An axis number of 0 means that technology information was output to the channel interface
UNS16	Fillup, used to force structure alignment
UNS32	m_h_count, number of assigned entries in the vector m_h_data[]
SOLLKONT_M_H_PROCESS	M_h_data[MAX_M_H_DATA] Vector containing information about M/H functions
UNS32	S_count, number of entries in the vector s_proc[]
SOLLKONT_S_PROCESS	s_proc[] Vector containing information about spindle functions
SOLLKONT_TOOL_ID	Tool, information about the current valid tool number

	SOLLKONT_M_H_PROCESS
UNS32	nr, number of the M/H function
UNS32	sync, synchronisation type of the M/H function, see [CHAN// Configuration of PLC functions]
UNS32	type, 1 = M function, 2 = H function

	SOLLKONT_S_PROCESS	
UNS16	Ax_nr, axis number of the spindle axis	
UNS16	Cmd, spindle command:	
	Value	**Command**
	3	**M3**
	4	**M4**
	5	**M5**

	19	M19
UNS32	Sync, synchronisation type of the spindle function	
SGN32	Position; target position in 0.1 µm if position moves	
SGN32	Revolution, command speed of the spindle in 10E-3°/s or µm/s.	

	SOLLKONT_TOOL_ID
SGN32	Basic, basic number of the tool
SGN32	Sister, sister number of the tool, -1 means unassigned.
SGN32	Variant, variant number of the tool, -1 means unassigned.

	SOLLKONT_M_H_PROCESS_V1
UNS32	nr, number of the M/H function
UNS32	sync, synchronisation type of the M/H function, see [CHAN// Configuration of PLC functions]
UNS32	type, 1 = M function, 2 = H function
SGN32	Add_value, additional value programmed in M/H function.

Constants

Constant	Value
MAX_SOLLKONT_VISU_DATA_COUNT_V0	15
MAX_SOLLKONT_VISU_DATA_COUNT_V1	10
MAX_SOLLKONT_VISU_DATA_COUNT_V2	5
MAX_SOLLKONT_VISU_DATA_COUNT_V3	10
MAX_SOLLKONT_VISU_DATA_COUNT_V4	7
MAX_SOLLKONT_VISU_DATA_COUNT_V5	4
MAX_SOLLKONT_VISU_DATA_COUNT_V6	7
MAX_SOLLKONT_VISU_DATA_COUNT_V7	6
MAX_SOLLKONT_VISU_DATA_COUNT_V8	4
MAX_SOLLKONT_VISU_DATA_COUNT_V9	5
MAX_SOLLKONT_VISU_DATA_COUNT_V10	4
MAX_SOLLKONT_VISU_DATA_COUNT_V11	3
ANZ_SIMU_KOORD	32
MAX_M_H_DATA	20
MAX_SPINDLE_DATA	6

Datentypen

Data type	C Datentyp	Description
SGN16	signed short	Signed 16 bit integer
UNS16	unsigned short	Unsigned 16 bit integer
SGN32	signed long	Signed 32 bit integer

UNS32	unsigned long	Unsigned 32 bit integer
REAL64	double	64-bit floating point number
ISG_CHAR	char	8-bit text character

4.2.1 HLI parameters up to CNC Build V2.20xx

Channel mode			
Description	Selection of a special channel mode such as syntax check or machining time calculation		
Data type	MCControlSGN32Unit, see description of Control Unit		
Access	PLC reads Request + State and writes Command + Enable		
ST Path	pMC[*channel_idx*]^.addr^.MCControlDecoder_Data.**MCControlSGN32Unit_ExecutionMode**		
Commanded, requested and return values			
ST Element	**.D_Command** **.D_Request** **.D_State**		
Data type	DINT		
Value range	**Value**	**Constant**	**Meaning**
	0x0000	ISG_STANDARD	Normal mode
	0x0001	SOLLKON	Block search
	0x0002	SOLLKON	Nominal contour visualisation simulation with output of visualisation data
	0x0802	SOLLKON_SUPRESS_OUTPUT & 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 [▶ 37]]). An additional client is linked to this output via DataFactory / CORBA.
	0x2000	SUPPRESS_TECHNO_OUTPUT	Without output of technology functions (M/H/T). Set implicitly in connection with syntax check
Redirection			
ST element	**.X_Enable**		

4.3 Application examples

The visualisation data described in the previous section can be read using the following applications, for example.

ADS access via AmsAdsDebugger

The AmsAdsViewer can execute a direct check of the individual parameters of the simulation on a running TwinCAT controller.

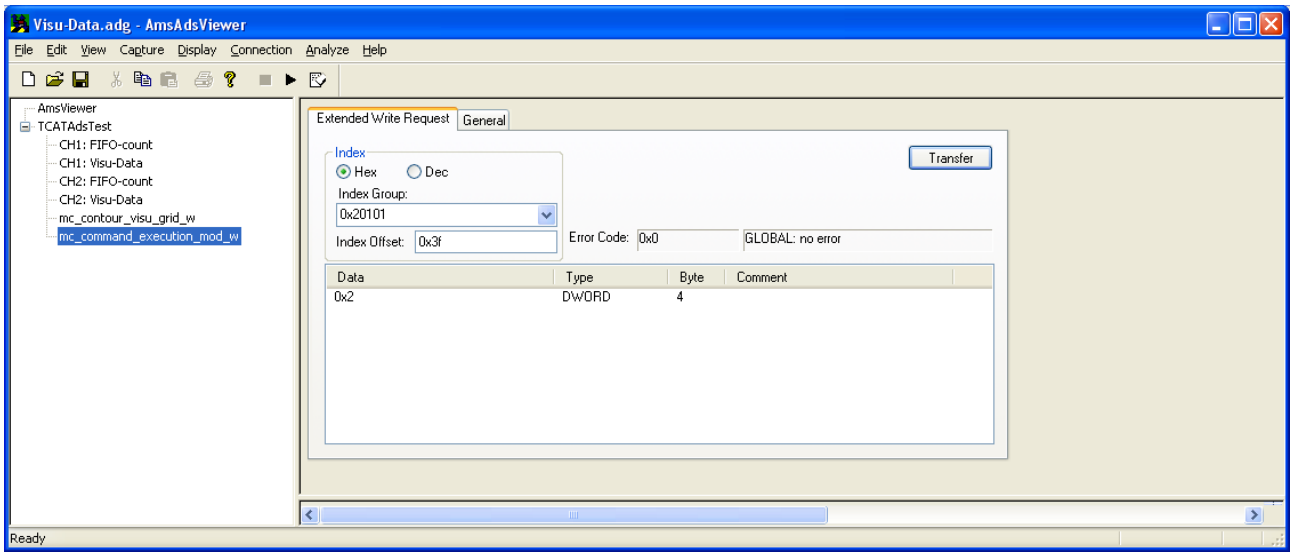


Fig. 6: ADS access via AmsAdsDebugger

ADS access via Object Browser

The operator selects contour visualisation as processing mode before program start. This setting is forwarded to the PLC via a so-called controller which the PLC can permit or reject.

In the same way, the PLC also has the option to select the processing mode = rapid contour visualisation without previous HMI request.

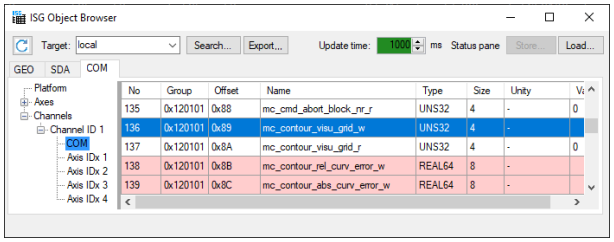


Fig. 7: ADS access via Object Browser

ADS access via Win32 application

ADS access via Win32 application

```
static BOOLEAN writeContourParameters(
UNS16 channel_nr, UNS32 grid, REAL64 abs_error, REAL64 rel_error)
{
  SGN32 result;
  SGN32 idx_group = 0x20100 + channel_nr;

  if ((channel_nr < 1) || (channel_nr > SYS_KANAL_MAX))
    return FALSE;

  result = AdsSyncWriteReq( &amsCom,     // Ams address of ADS server
                            idx_group,   // index group:
                            0x89,        // index offset:
                            sizeof(grid),// count of bytes
                                         to read
                            &grid);      // pointer to the
                                         client buffer
  if (0 != result)
    return FALSE;

  result = AdsSyncWriteReq( &amsCom,     // Ams address of
                                         ADS server
                            idx_group,   // index group:
                            0x8c,        // index offset:
                            sizeof(abs_error),// count of
                                              bytes to
                                              read
                            &abs_error);  // pointer to the
                                          client buffer
  if (0 != result)
    return FALSE;

  result = AdsSyncWriteReq( &amsCom,     // Ams address of ADS server
                            idx_group,     // index group:
                            0x8B,          // index offset:
                            sizeof(rel_error),// count of
                                              bytes to
                                              read
                            &rel_error);  // pointer to the
                                          client buffer
  if (0 != result)
    return FALSE;

  return TRUE;
}
static BOOLEAN activateContourVisu( UNS16 channel_nr)
{
  SGN32 result;
  SGN32 idx_group = 0x20100 + channel_nr;
  UNS32 execution_mode = SOLLKON;

  if ((channel_nr < 1) || (channel_nr > SYS_KANAL_MAX))
    return FALSE;

  result = AdsSyncWriteReq( &amsCom,      // Ams address of ADS server
                            idx_group,     // index group:
                            0x3f,          // index offset:
                            sizeof(execution_mode),
                            &execution_mode);
  if (0 != result)
    return FALSE;
  return TRUE;
}



static BOOLEAN readContourData (
SOLLVISU_PDU_CHAN *p_visu_pdu, UNS16 channel_nr)
{
  SGN32 result;
  UNS32 count;
  UNS32 fifo_count;
  SGN32 idx_group = 0x20100 + channel_nr;

  if ((channel_nr < 1) || (channel_nr > SYS_KANAL_MAX))
```

```
    return FALSE;

  // Read number of entries in visualisation FIFO output
  result = AdsSyncReadReqEx( &amsCom, // Ams address of ADS server
                             idx_group,  // index group:
                             0x2001,     // index offset:
                             sizeof(fifo_count),
                             &fifo_count,
                             &count);

  if (0 != fifo_count)
{
    // Data present, read via COM
    result = AdsSyncReadReqEx( &amsCom,   // Ams address of ADS server
                           idx_group,     // index group:
                              0x2000,        // index offset:
                              sizeof(*p_visu_pdu),
                              p_visu_pdu,
                              &count)
    if (0 == result)
  return TRUE;
}
    return FALSE;
}
```

Display of axis positions in DXF format

Display of axis positions in DXF format

```
%contour_visu

N001 G01 G90 X0 Y0 Z0 F1000
N100 X100
N200         Y100
N300 X0
N400 Y0
N500 X50 Y50 Z200
N500 X100 Y100 Z0
N600 X0
N700 X50 Y50 Z200
N800 X100 Y0 Z0
N900 G02 I100


N1000 #CS ON[0,0,100, 45 ,0,0]
N1001 G01 G90 X0 Y0 Z0 F1000
N1100 X100
N1200       Y100
N1300 X0
N1400 Y0
N1500 X50 Y50 Z200
N1500 X100 Y100 Z0
N1600 X0
N1700 X50 Y50 Z200
N1800 X100 Y0 Z0
N1900 G02 I100
N1500 #CS OFF

N2000 #CS ON[0,100,-100, 0, 45,0]
N2001 G01 G90 X0 Y0 Z0 F1000
N2100 X100
N2200       Y100
N2300 X0
N2400 Y0
N2500 X50 Y50 Z200
N2500 X100 Y100 Z0
N2600 X0
N2700 X50 Y50 Z200
N2800 X100 Y0 Z0
N2900 G02 I100
N2500 #CS OFF

M30
```

The read-out axis positions can be used to display the actual path motion in DXF format.

DXF output file

```
0
SECTION
  2
HEADER
999
isg.dxf created by TwinCAT CNC
0
ENDSEC
0
SECTION
  2
TABLES
0
ENDSEC
0
SECTION
  2
BLOCKS
0
ENDSEC
0
SECTION
  2
ENTITIES
0
LINE
  8
0
62
  2
10
0.000000
20
0.000000
30
0.000000
11
10.000000
21
0.000000
31
0.000000
0
...
```

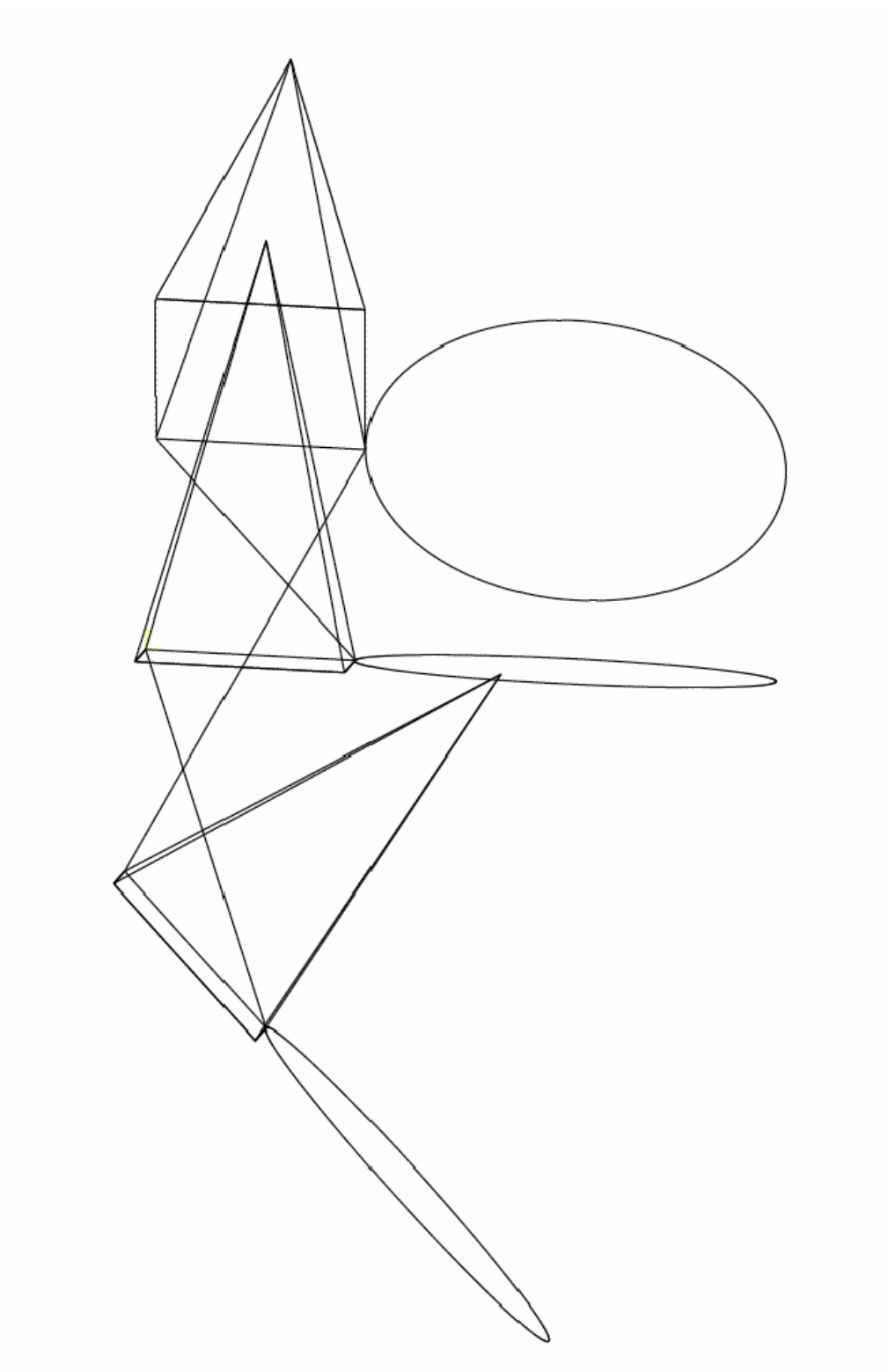


Fig. 8: Display of the DXF output file in a viewer

5 Online contour visualisation

Activation

Online contour visualisation is activated by transferring the program start option **ON_LINE** on the HLI to the controller at program start (see documentation on the HLI [▶ 19]).

As opposed to rapid contour visualisation, the **online contour visualisation** mode executes the real processing of the NC program. The read-out values are filtered to obtain a high-performance data transfer.

Sample grid

Data reduction can be applied in this operating mode. Depending on the motion block used (straight/curved), the interpolation point grid can be specified for the interpolation either

- by specifying a maximum interpolation point interval
- or a maximum path error can be specified (see “Selection via HMI / ADS“).

This can be defined in the following parameters:

Parameter	Format:	Description	Index-Group	Index-Offset
mc_contour_visu_grid_w mc_contour_visu_grid_r	UNS32	Output grid for nominal contour visualisation for linear blocks (G00/ G01) in [0.1 µm]	0x2010<c> c element [1; max. channel]	0x89, 0x8a
mc_contour_rel_curv_error_w	REAL64	Maximum relative path error in [0.1%] for nominal contour visualisation of circles or polynomials	0x2010<c> c element [1; max. channel]	0x8b
mc_contour_abs_curv_error_w	REAL64	Maximum absolute path error in [0.1 µm] for nominal contour visualisation of circles and polynomials	0x2010<c> c element [1; max. channel]	0x8c

Parameterisation

Parameterisation takes place analogously to Rapid Contour Visualisation (see “Parameters [▶ 40]”.)

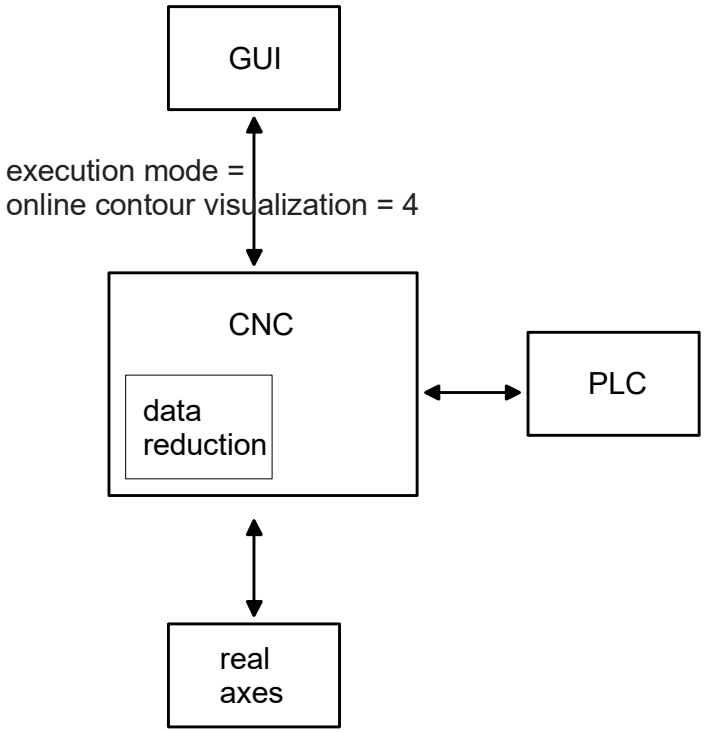


Fig. 9: Online contour visualisation

6 Scene contour visualisation

In **scene mode** the CNC program is actually executed, i.e. the values output are filtered in time. The required data rate can be specified as “frames per second”.

Activation

The **scene contour visualisation** is activated by transferring the program type option **BEARB_MODE_SCENE** to the HLI or the user interface to the controller at program start.

Logging

In the **scene** display, all the motions of every coordinate system of the kinematic chain are logged. This visualises the motion of each graphical object. In addition, this motion can be visualised as a track.

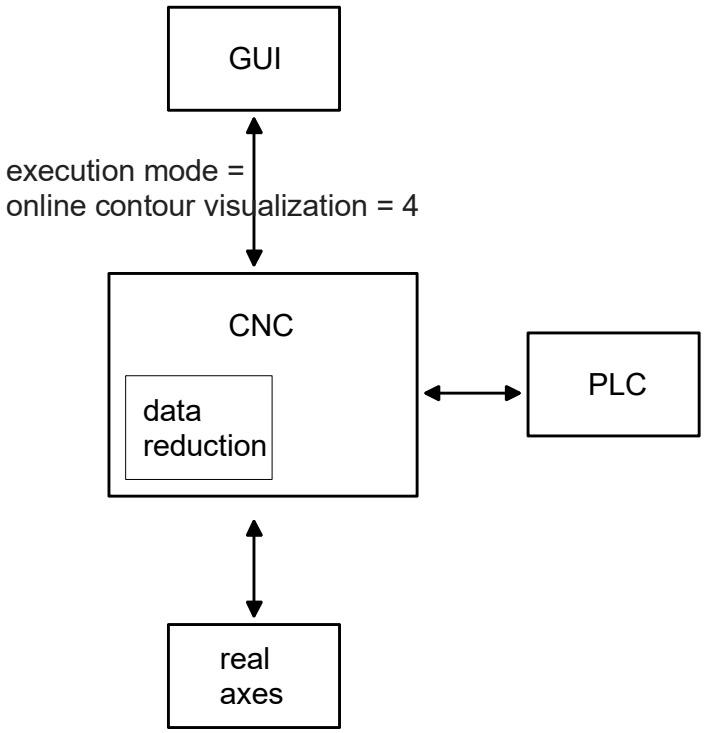
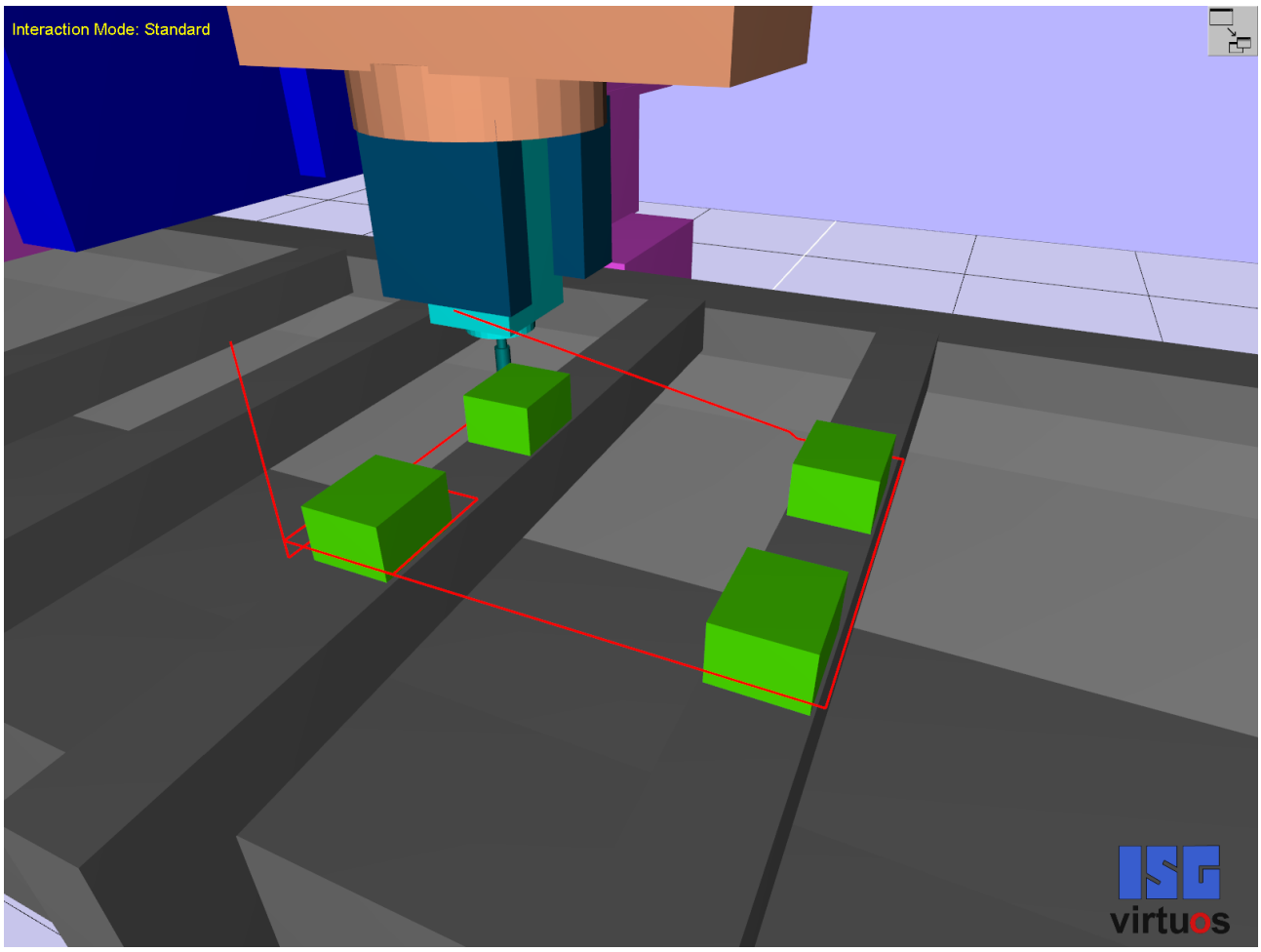


Fig. 10: Scene contour visualisation


Interaction Mode: Standard
ISG
virtuos

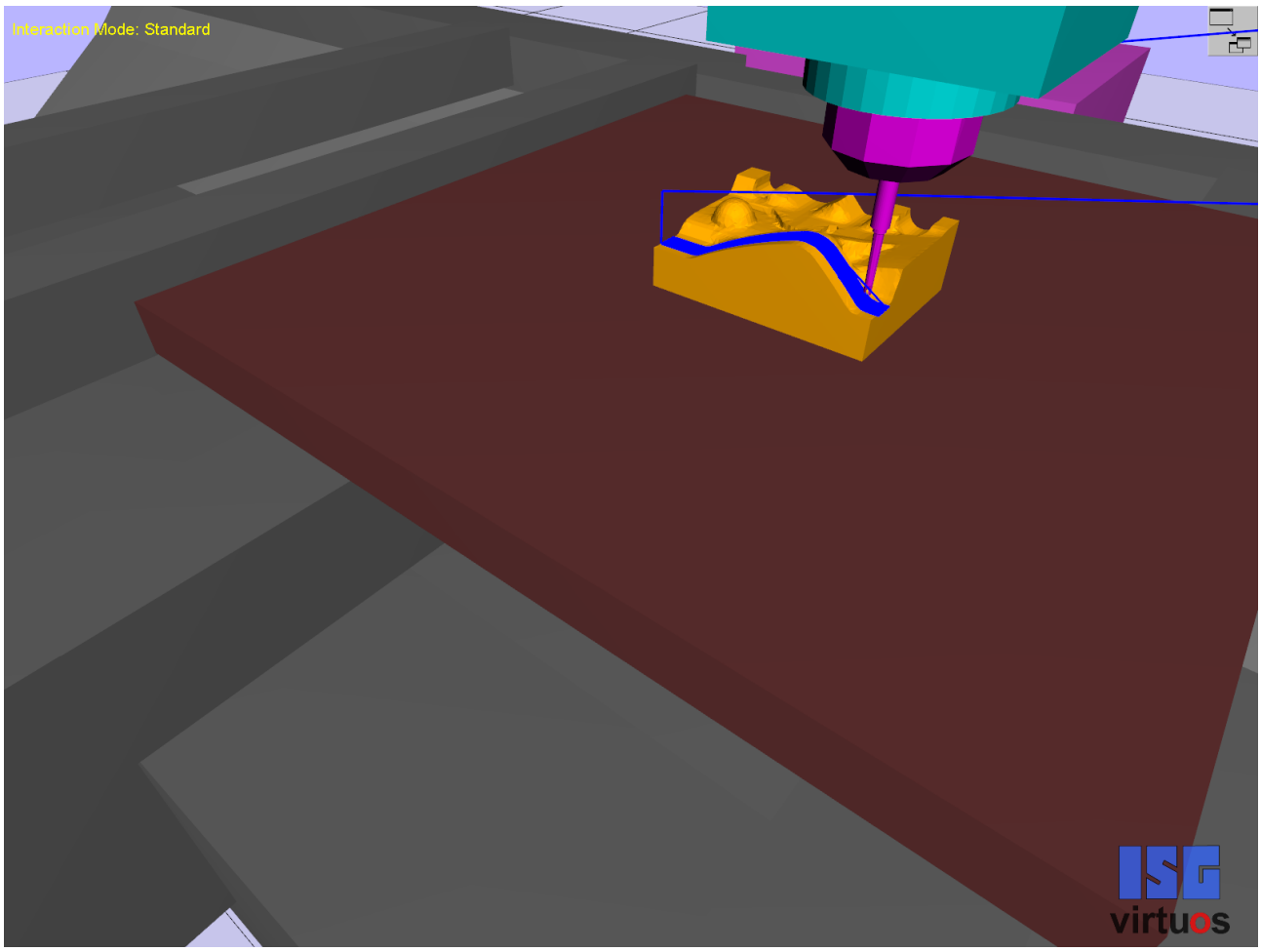


Fig. 11: Examples of contour visualisation with the Scene mode

7 Parameter

7.1 Overview

ID	Parameter	Description
P-CHAN-00121	simu_output_wcs	Display format during machining simulation
P-CHAN-00183	simu_ignore_internal_stop_cond	Ignore internal stop conditions with rapid contour visualisation
P-STUP-00040	single_protocol_fifo	Global or channel-specific output of display data
P-STUP-00039	contour_visu_ifc_version	Version identifier of visualisation data

7.2 Description

P-CHAN-00121	**Display format during machining simulation**
Description	This parameter switches over the format of display data for the coordinate system at the interface of the machining simulation.
Parameter	simu_output_wcs
Data type	BOOLEAN
Data range	0: Display of axis coordinates including offsets (machine coordinates) 1: Display of absolute coordinates without offsets (programmed coordinates)
Dimension	----
Default value	0
Remarks	

P-CHAN-00183	**Ignore internal stop conditions with rapid contour visualisation**
Description	This parameter prevents the NC program from stopping because of internal stop conditions (e. g. M00) during rapid contour visualisation.
Parameter	simu_ignore_internal_stop_cond
Data type	BOOLEAN
Data range	0: Internal stop conditions are effective (default). 1: Internal stop conditions are ignored.
Dimension	----
Default value	0
Remarks	

P-STUP-00040	**Global or channel-specific output of display data**
Description	This parameter defines whether visualisation data is written to a FIFO output for each channel or whether the visualisation data of all channels is written to a global FIFO output.
Parameter	single_protocol_fifo
Data type	BOOLEAN
Data range	0: Channel-specific output of visualisation data 1: Common output of visualisation data.
Dimension	----
Default value	0 *
Remarks	* 1 as of CNC Build V3.1.3038

P-STUP-00039	**Version identifier of visualisation data**

Description	The parameter sets the type of data structure which the contour visualisation ([FCT-C17 [▶ 8]]) supplies. Depending on the setting selected, more or less visualisation data is generated. An overview of existing data structures is contained in [FCT-C17 [▶ 8]].	
Parameter	contour_visu_ifc_version	
Data type	UNS32	
Data range	contour_visu_ifc_version	Data structure
	0	SOLLKONT_VISU_DATA_V0 (default)
	1	SOLLKONT_VISU_DATA_V1
	2	SOLLKONT_VISU_DATA_V2
	3	SOLLKONT_VISU_DATA_V3
	4	SOLLKONT_VISU_DATA_V4
	5	SOLLKONT_VISU_DATA_V5
	6	SOLLKONT_VISU_DATA_V6
	7	SOLLKONT_VISU_DATA_V7
	8	SOLLKONT_VISU_DATA_V8
	9	SOLLKONT_VISU_DATA_V9
	10	SOLLKONT_VISU_DATA_V10
	11	SOLLKONT_VISU_DATA_V11
Dimension	----	
Default value	0	
Remarks		

8 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

C

O

P

More Information:
www.beckhoff.com/TF5200

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