



DOKUMENTATION ISG-kernel

Funktionsbeschreibung Konturvisualisierung

Kurzbezeichnung:
FCT-C17

Vorwort

Rechtliche Hinweise

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte und der Funktionsumfang werden jedoch ständig weiterentwickelt. Wir behalten uns das Recht vor, die Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern.

Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs-, Automatisierungs- und Antriebstechnik, das mit den geltenden Normen, der zugehörigen Dokumentation und der Aufgabenstellung vertraut ist.

Zur Installation und Inbetriebnahme ist die Beachtung der Dokumentation, der nachfolgenden Hinweise und Erklärungen unbedingt notwendig. Das Fachpersonal ist verpflichtet, für jede Installation und Inbetriebnahme die zum betreffenden Zeitpunkt veröffentlichte Dokumentation zu verwenden.

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbarer Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

Weiterführende Informationen

Unter den Links (DE)

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

bzw. (EN)

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

finden Sie neben der aktuellen Dokumentation weiterführende Informationen zu Meldungen aus dem NC-Kern, Onlinehilfen, SPS-Bibliotheken, Tools usw.

Haftungsausschluss

Änderungen der Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig.

Marken und Patente

Der Name ISG®, ISG kernel®, ISG virtuos®, ISG dirigent® und entsprechende Logos sind eingetragene und lizenzierte Marken der ISG Industrielle Steuerungstechnik GmbH.

Die Verwendung anderer in dieser Dokumentation enthaltene Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

Copyright

© ISG Industrielle Steuerungstechnik GmbH, Stuttgart, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster oder Geschmacksmustereintragung vorbehalten.

Allgemeine- und Sicherheitshinweise

Verwendete Symbole und ihre Bedeutung

In der vorliegenden Dokumentation werden die folgenden Symbole mit nebenstehendem Sicherheitshinweis und Text verwendet. Die (Sicherheits-) Hinweise sind aufmerksam zu lesen und unbedingt zu befolgen!

Symbole im Erklärtext

- Gibt eine Aktion an.
- ⇒ Gibt eine Handlungsanweisung an.



GEFAHR

Akute Verletzungsgefahr!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, besteht unmittelbare Gefahr für Leben und Gesundheit von Personen!



VORSICHT

Schädigung von Personen und Maschinen!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, können Personen und Maschinen geschädigt werden!



Achtung

Einschränkung oder Fehler

Dieses Symbol beschreibt Einschränkungen oder warnt vor Fehlern.



Hinweis

Tipps und weitere Hinweise

Dieses Symbol kennzeichnet Informationen, die zum grundsätzlichen Verständnis beitragen oder zusätzliche Hinweise geben.



Beispiel

Allgemeines Beispiel

Beispiel zu einem erklärten Sachverhalt.



Programmierbeispiel

NC-Programmierbeispiel

Programmierbeispiel (komplettes NC-Programm oder Programmsequenz) der beschriebenen Funktionalität bzw. des entsprechenden NC-Befehls.



Versionshinweis

Spezifischer Versionshinweis

Optionale, ggf. auch eingeschränkte Funktionalität. Die Verfügbarkeit dieser Funktionalität ist von der Konfiguration und dem Versionsumfang abhängig.

Inhaltsverzeichnis

Vorwort	2
Allgemeine- und Sicherheitshinweise	3
1 Übersicht	6
2 Beschreibung.....	7
3 Dry Run, Trockenlauf	10
4 Schnelle Konturvisualisierung	11
4.1 Beschreibung	11
4.2 Schnittstellenanbindung	16
4.2.1 HLI- Kanalbetriebsart bis CNC-Version V2.20xx	29
4.3 Applikationsbeispiele.....	31
5 Online Konturvisualisierung.....	38
6 Szene Konturvisualisierung	40
7 Parameter	42
7.1 Übersicht.....	42
7.2 Beschreibung	42
8 Anhang	44
8.1 Anregungen, Korrekturen und neueste Dokumentation.....	44
Stichwortverzeichnis.....	45

Abbildungsverzeichnis

Abb. 1:	Darstellung des verwendeten Koordinatensystems	9
Abb. 2:	Konturvisualisierung im Trockenlauf.....	10
Abb. 3:	Stützpunktraster für gekrümmte Konturelemente	13
Abb. 4:	Relativer und absoluter Sekantenfehler	13
Abb. 5:	Schnelle Konturvisualisierung	14
Abb. 6:	ADS-Zugriff via AmsAdsDebugger	31
Abb. 7:	Zugriff über CNC Objekt-Browser.....	32
Abb. 8:	Darstellung der DXF-Ausgabedatei in einem Viewer	37
Abb. 9:	Online Konturvisualisierung	39
Abb. 10:	Szene Konturvisualisierung	40
Abb. 11:	Beispiele für die Konturvisualisierung mit dem Szene-Modus.....	41

1 Übersicht

Aufgabe

Für die grafische Darstellung von Maschinenbewegungen können von der Steuerung die Achspositionen bereitgestellt werden und mittels eines Anwenderprogramms oder in der Bedienoberfläche visualisiert werden.

Dies kann erfolgen:

- additiv zum normalen Betrieb der Steuerung
- oder simulativ ohne reale Achsbewegungen.

Es wird unterschieden zwischen den 4 Modi:

1. Dry Run, Trockenlauf
2. Schnelle Konturvisualisierung
3. Online Konturvisualisierung
4. Szene Konturvisualisierung

Eigenschaften

Zur Aktivierung der Simulation muss vor dem Start eines NC-Programms der Bearbeitungsmodus auf Simulation geschaltet werden.

Dies ist möglich über:

- die Oberflächen- oder
- die SPS-Schnittstelle

Parametrierung

Zur Konfiguration der o.g. Modi müssen verschiedene Parameter [► 42] belegt werden.

Obligatorischer Hinweis zu Verweisen auf andere Dokumente

Zwecks Übersichtlichkeit wird eine verkürzte Darstellung der Verweise (Links) auf andere Dokumente bzw. Parameter gewählt, z.B. [PROG] für Programmieranleitung oder P-AXIS-00001 für einen Achsparameter.

Technisch bedingt funktionieren diese Verweise nur in der Online-Hilfe (HTML5, CHM), allerdings nicht in PDF-Dateien, da PDF keine dokumentenübergreifenden Verlinkungen unterstützt.

2 Beschreibung

Für die grafische Darstellung von Maschinenbewegungen können von der Steuerung die Achspositionen bereitgestellt und mittels eines Anwenderprogramms oder in der Bedienoberfläche visualisiert werden.

In der CNC werden im Normalbetrieb Achspositionen im Interpolationstakt als Anzeigedaten bereitgestellt. Zur Vereinfachung der Visualisierung kann die Menge der bereitgestellten Daten reduziert werden, indem nur für die Visualisierung relevante Positionen ausgegeben werden, z.B. der exakte Endpunkt eines Konturelements. So bleiben Ecken auch in den reduzierten Visualisierungsdaten als Ecken erkennbar. Die korrekte Darstellung der Ecken soll insbesondere auch dann möglich sein, wenn aufgrund einer möglichst schnellen Darstellung nur sehr wenige Punkte gemeldet werden.

Unterschiedliche Betriebsmodi der Konturvisualisierung

Dry Run, Trockenlauf

Im Modus **Dry Run** oder Trockenlauf wird das NC-Programm normal decodiert und die Positionen werden interpoliert. Die Bewegung der Achsen wird nicht an den Lageregler weitergegeben, sodass es zu keiner Achsbewegung kommt.

Schnelle Konturvisualisierung

Die Steuerung arbeitet im Simulationsbetrieb ohne reale Achsbewegung, die Abarbeitung des CNC-Programms erfolgt im Schnelldurchlauf. Dadurch werden die programmierten Konturen größer abgetastet, wobei jedoch Ecken erhalten bleiben.

Die Anzahl der Stützpunkte für die Visualisierung ist deutlich reduziert.

Es finden keine realen Achsbewegungen statt.

Online Konturvisualisierung

Die Steuerung arbeitet im Normalbetrieb, die Abarbeitung des CNC-Programms wird nicht beeinflusst. An der Schnittstelle der Konturvisualisierung werden Positionswerte in einem größeren Raster zur Visualisierung bereitgestellt.

Szene

Im NC-Programm wird die sequentielle kinematische Kette definiert. An jedem Koordinatensystem der kinematischen Kette (LINKPOINT) kann ein grafisches Objekt positioniert werden. Über eine Schnittstelle wird die Bewegung jedes Koordinatensystems protokolliert. Die Bewegung des grafischen Objektes kann u.a. in kernelCAM als Spur aufgezeichnet werden.



Hinweis

Die Funktionalität **Szene** ist unter TwinCAT nicht verfügbar.

Die folgende Tabelle stellt die Modi einander gegenüber:

Bearbeitungs- mode	Datenreduktion vor Interpolati- on	Datenreduktion nach Interpolati- on	Koordinaten- system der ausgegebenen Daten	Besonderheiten	Viewer
1. Trockenlauf, Dry run	- keine -	- keine -	PCS	Ohne reale Achsbewegung normale Pro- grammbearbei- tung	
2. Schnelle Kontur-visuali- sierung	geometrisches Gitter, abs./rel. Sekantenfehler	Keine Datenre- duktion nach In- terpolation, es werden keine Stützpunkte er- zeugt, die nicht auf dem Visuali- sierungsraster liegen.	WCS oder ACS	ohne reale Achs- bewegung mög- lich. Schnelle Pro- gramm-abarbei- tung	KernelCAM in Vorbereitung
3. Online Kon- tur-visualisie- rung	- keine -	geometrisches Gitter, abs./rel. Sekantenfehler	WCS oder ACS		KernelCAM in Vorbereitung
4. Szene	- keine -	Zeitliche Abtas- tung mit Bilder pro Sekunde	MCS=W0 beliebiger Punkt der kinemati- schen Kette, auch TCP	Für beliebige se- rielle Kinemati- ken verfügbar. Kinematische Kette muss im NC-Programm initialisiert wer- den	VirtuosV als vCAM

Koordinatensysteme

Bei den einzelnen Schnittstellen zur Visualisierung stehen unterschiedliche Koordinatensysteme zur Verfügung. Folgende Definition wird hier verwendet:

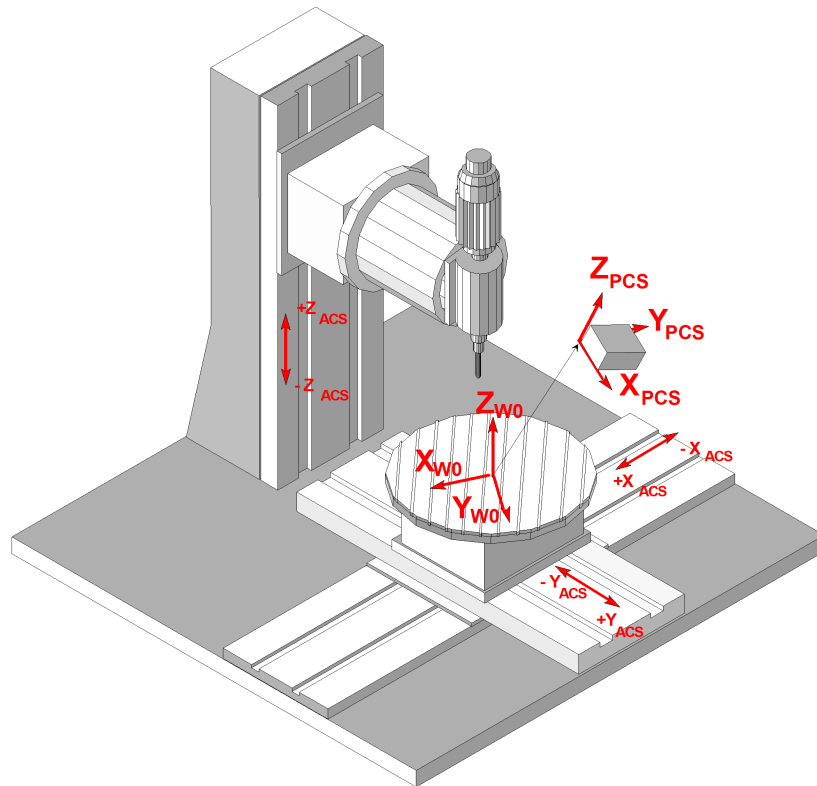


Abb. 1: Darstellung des verwendeten Koordinatensystems

ACS:	Axis Coordinate Sytem, Achskoordinatensystem
W0:	Base Workpiece Coordinate Sytem, kartesisches Grundkoordinatensystem der Maschine, bezogen auf den Werkstückspannplatz
PCS:	Programming Coordinate System, Programmierkoordinatensystem

3 Dry Run, Trockenlauf

Der Dry Run wird aktiviert, indem beim Programmstart die Programmstartoption **0x40 MACHINE_LOCK** auf dem HLI an die Steuerung übergeben wird (siehe Dokumentation zur HLI [► 18]).

Im Modus **Dry Run** oder Trockenlauf wird das NC-Programm normal decodiert und die Positionen werden interpoliert. Die Bewegung der Achsen wird nicht an den Lageregler weitergegeben, so dass es zu keiner Achsbewegung kommt.

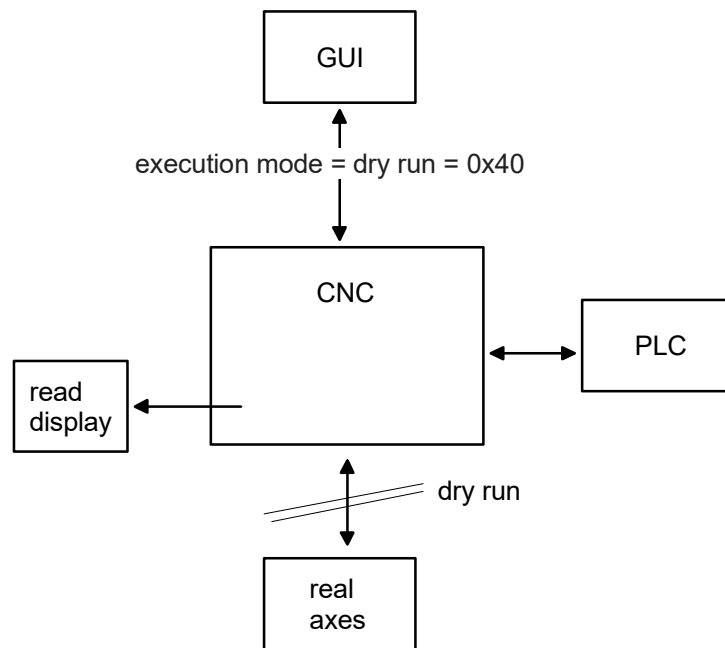


Abb. 2: Konturvisualisierung im Trockenlauf



Hinweis

Beim Wechsel der Betriebsart von Normalbetrieb in den Trockenlauf müssen alle Achsen und Spindeln im Stillstand sein.

Ist dies nicht der Fall wird bei bewegter Spindel der Fehler ID 60269 ausgegeben.

4 Schnelle Konturvisualisierung

4.1 Beschreibung

Aktivierung

Die **Schnelle Konturvisualisierung** wird aktiviert, indem beim Programmstart die Programmstartoption **SOLLKON** auf dem HLI an die Steuerung übergeben wird (siehe Dokumentation zur HLI [► 18]).

Bei der Schnellen Konturvisualisierung werden keine Achsbewegungen ausgeführt. Die Ausgabe der Visualisierungsdaten erfolgt in einem reduzierten Raster. Es ist das gewünschte Stützpunktraster bzw. der zulässige Sekantenfehler für die Interpolation anzugeben. Durch die gröbere Rasterung erfolgt eine schnellere Abarbeitung des NC-Programms.

Programmierte Verweilzeiten (G04, #TIME) werden ignoriert.

Anwendungen

Die Simulation kann u. a. für die folgenden Anwendungen verwendet werden:

- „Syntaxcheck“ unter Verwendung des gesamten NC-Kanals. Im Unterschied zur Betriebsart Syntaxcheck sind bei der Simulation, bis auf den Lageregler, alle Module des NC-Kanals aktiv. Dadurch können Fehler erkannt werden, die durch den Syntaxcheck nicht erfasst werden, z.B. Ausgleichsbewegungen bei der Werkzeugradiuskompensation oder überfahrene Softwareendschalter.
- Vorab-Visualisierung eines NC-Programms (offline).

Rasterung

Abhängig vom verwendeten Bewegungssatz (gerade/gekrümmt) kann das Stützpunktraster für die Interpolation entweder

- durch Angabe eines maximalen Stützpunktabstands
- oder eines maximalen Bahnfehlers angegeben werden.

Dies lässt sich in folgenden Parametern festlegen:

Parameter	Format	Beschreibung	Index-Group	Index-Offset
mc_contour_visu_grid_w mc_contour_visu_grid_r	UNS32	Streckenraster für Sollkonturvisualisierung für Linearsätze (G00/G01) in [0.1 µm]	0x2010<c> c Element [1; max. Kanal]	0x89, 0x8a
mc_contour_rel_curv_error_w	REAL64	Maximaler relativer Bahnfehler in [0.1%] für Sollkonturvisualisierung für Kreise oder Polynome	0x2010<c> c Element [1; max. Kanal]	0x8b
mc_contour_abs_curv_error_w	REAL64	Maximaler absoluter Bahnfehler in [0.1 µm] für Sollkonturvisualisierung	0x2010<c> c Element [1; max. Kanal]	0x8c

Die Zielpunkte jedes NC-Satzes werden immer ausgegeben.

Stützpunktraster für Linearsätze

Für Linearsätze wird der Stützpunktastand für die Interpolation direkt angegeben. In Folge werden die Achsdynamik und die programmierte Sollgeschwindigkeit nicht berücksichtigt.

Für jeden Linearsatz wird der programmierte Endpunkt auch dann ausgegeben, wenn er nicht auf dem eingestellten Stützpunktraster liegt. Hierdurch wird erreicht, dass Ecken einer Kontur stets angezeigt werden.

Falls ein Linearsatz kürzer als das eingestellte Stützpunktraster ist, erfolgt keine Ausgabe des Endpunktes.

Stützpunktraster für gekrümmte Konturelemente

Für gekrümmte Konturelemente (Kreise, Polynome) kann ein

- absoluter Sekantenfehler
- sowie ein relativer Sekantenfehler

angegeben werden.

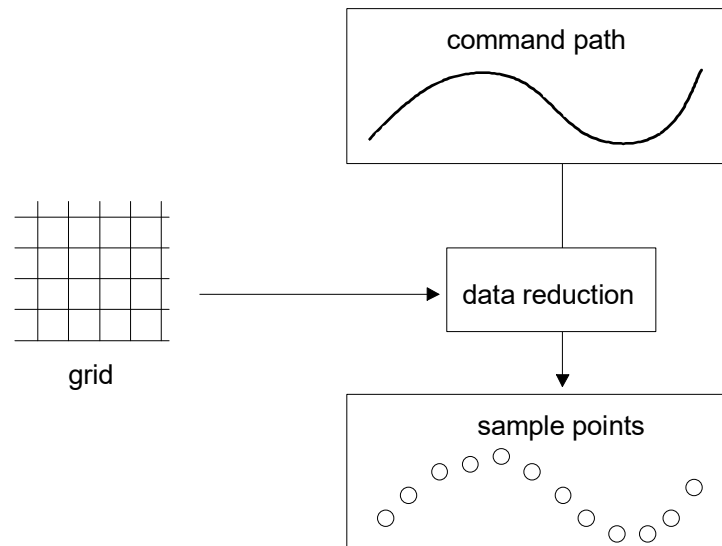


Abb. 3: Stützpunktraster für gekrümmte Konturelemente

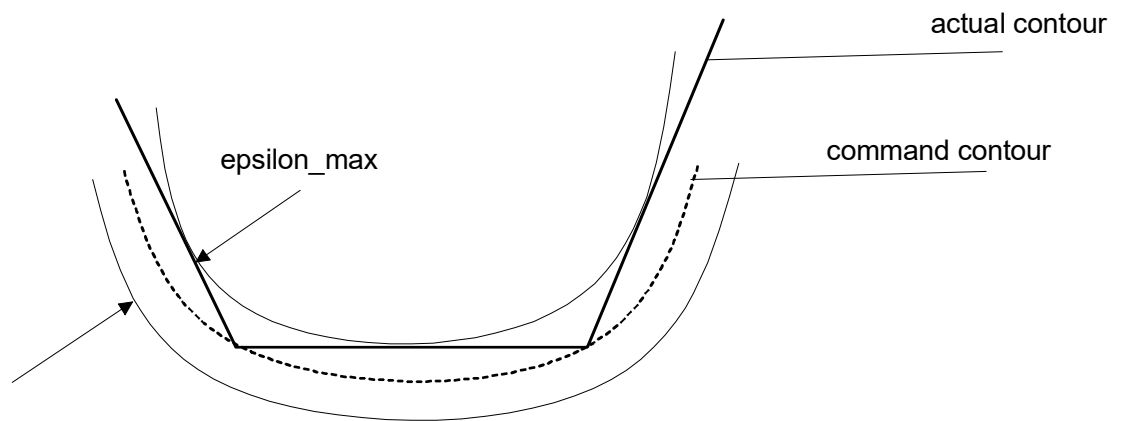


Abb. 4: Relativer und absoluter Sekantenfehler

$$\epsilon_{\max} = r \cdot \epsilon_{\text{rel}} \text{ für: } \epsilon_{\text{rel}} \leq \epsilon_{\text{abs}}$$

$$\epsilon_{\max} = \epsilon_{\text{abs}} \text{ für: } \epsilon_{\text{rel}} \geq \epsilon_{\text{abs}}$$

Der resultierende Sekantenfehler ist der kleinere der beiden Werte.

Stoppbedingungen

Die Abarbeitung eines NC-Programms kann durch interne oder externe Einflüsse angehalten werden.

Interne Stoppbedingungen sind NC-Befehle, die nur durch eine Benutzerinteraktion beendet werden. Beispiel ist ein programmierter Halt (M00). Durch den Kanalparameter P-CHAN-00183 wird das Stoppen der Programmabarbeitung verhindert.

Bei externen Stoppbedingungen bewirkt der Benutzer selbst ein Anhalten der Abarbeitung des NC-Programms. Beispiele sind:

- Vorschubstopp über die SPS-Schnittstelle
- Nichtquittieren einer Technologiefunktion

Externe Stoppbedingungen wirken immer. Der Benutzer hat daher Sorge zu tragen, dass die Programmabarbeitung nicht angehalten wird.

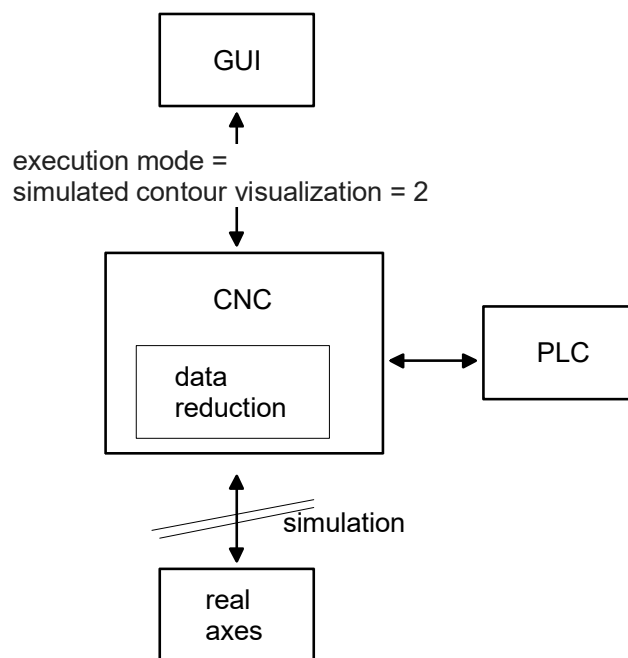


Abb. 5: Schnelle Konturvisualisierung



Hinweis

Im Gegensatz zum Syntaxcheck [siehe Funktionsbeschreibung FCT-C9] ist eine Fortsetzung der Programmabarbeitung nach Auftreten eines Fehlers nicht möglich.

Bei Auftreten eines Fehlers muss im Kanal ein Reset ausgelöst und das Programm nach Behebung des Fehlers erneut gestartet werden.

Ausgabe

Die erzeugten Visualisierungsdaten können über CNC Objekte gelesen werden. Entsprechend des eingestellten Rasters werden die Verfahrssätze unterteilt und die Achspositionen ausgegeben.

Die Achspositionen können auf 2 Arten ausgegeben werden:

- Anzeige der Achskoordinaten einschließlich Versätze (Maschinenkoordinaten).
- Anzeige der Absolutkoordinaten ohne Versätze (Programmierte Koordinaten).

Unter dem Hochlaufparameter P-STUP-00039 kann ausgewählt werden, welche Daten ausgegeben werden sollen.

4.2 Schnittstellenanbindung

Es gibt 2 Möglichkeiten der Schnittstellenanbindung (im Folgenden beschrieben):

- Anwahl über HMI oder über CNC Objekte
- Kommandos und Anzeige über HLI

Zugriff über CNC Objekte

Über CNC Objekte kann die Visualisierung parametrierbar und die Visualisierungsdaten abgefragt werden.

Alle Objektzugriffe erfolgen über die Task COM. Der Zugriff auf die einzelnen Daten/Parameter erfolgt über die folgenden Indexgroups/ -offsets.

Bei der Indexgroup ist für den Platzhalter <c> jeweils die Kanalnummer einzusetzen, wobei <c> innerhalb [1; max.Kanalanzahl] liegt.

Parameter	Format	Beschreibung	Index-Group	Index-Offset
mc_command_execution_mode_r, mc_command_execution_mode_w	UNS32	Anwahl der Sollkonturvisualisierung 0x0000 ISG_STANDARD Normalbetrieb 0x0002 SOLLKON Sollkonturvisualisierung 0x0004 ON_LINE Online-Visu 0x0008 SYNCHK Syntaxcheck	0x2010<c> c Element [1; max. Kanal]	0x40, 0x3f
mc_contour_visu_grid_w mc_contour_visu_grid_r	UNS32	Streckenraster für Sollkonturvisualisierung für Linearsätze (G00/G01) in [0.1 µm]	0x2010<c> c Element [1; max. Kanal]	0x89, 0x8a
mc_contour_rel_curv_error_w	REAL64	Maximaler relativer Bahnfehler in [0.1%] für Sollkonturvisualisierung für Kreise oder Polynome	0x2010<c> c Element [1; max. Kanal]	0x8b
mc_contour_abs_curv_error_w	REAL64	Maximaler absoluter Bahnfehler in [0.1 µm] für Sollkonturvisualisierung	0x2010<c> c Element [1; max. Kanal]	0x8c



Hinweis

Bei gekrümmten Konturelementen (Kreise, Polynome) wird der Parameter berücksichtigt, welcher zum kleinsten Streckenraster führt.

Kommandos und Anzeige über HLI

Die Konturvisualisierung kann über die HLI kommandiert und angezeigt werden. Die folgende Tabelle listet die Parameter auf, die hierbei belegt sein müssen.

Kanalbetriebsart			
Beschreibung	Anwahl einer speziellen Kanalbetriebsart, wie z.B. Syntaxcheck oder Fertigungszeitberechnung		
Datentyp	MC_CONTROL_SGN32_UNIT, s. Beschreibung Control Unit		
Zugriff	PLC liest request_r + state_r und schreibt command_w + enable_w		
ST-Pfad	gpCh[channel_idx]^*.decoder_mc_control.execution_mode		
Kommandierter, angeforderter und Rückgabewert			
ST-Element	.command_w .request_r .state_r		
Datentyp	DINT		
Wertebereich	Wert	Konstante	Bedeutung
	0x0000	ISG_STANDARD	Normalbetrieb
	0x0001	SV	Satzvorlauf
	0x0002	SOLLKON	Simulation Sollkonturvisualisierung mit Ausgabe der Visualisierungsdaten
	0x0802	SOLLKON_SUPPRESS_OUTPUT & SOLLKON	Simulation Sollkonturvisualisierung ohne Ausgabe der Visualisierungsdaten
	0x0004	ON_LINE	Simulation Onlinevisualisierung
	0x0008	SYNCHK	Simulation Syntaxcheck
	0x0010	PROD_TIME	Simulation Fertigungszeitberechnung
	0x0020	ONLINE_PROD_TIME	Simulation Online-Fertigungszeitberechnung
	0x0040	MACHINE_LOCK	Dry Run ohne Achsbewegung
	0x0080	ADD_MDI_BLOCK	Erweiterter Handsatzmodus: das Ende eines Handsatzes wird nicht als Programmende gewertet, sondern erlaubt es weitere Handsätze zu beauftragen.
	0x0100	KIN_TRAFO_OFF	Überschreibt die automatische Freischaltung für kinematische Transformationen durch eine in den Kanalparametern (sda_mds*.lis) definierte Kenngröße
	0x1000	BEARB_MODE_SCENE	Durch das Einschalten des SZENE-Modus wird die Ausgabe der #SCENE-Befehle auf der Schnittstelle eingeschaltet (s.a. [FCT-C17// Szene Konturvisualisierung [► 40]]). Die Anbindung eines weiteren Clients an diese Ausgabe findet über die DataFactory / CORBA statt.
	0x2000	SUPPRESS_TECHNO_OUTPUT	Ohne Ausgabe von Technofunktionen (M/H/T). Wird implizit gesetzt in Verbindung mit Syntaxcheck.
Umleitung			

ST-Element	.enable_w
------------	------------------

Ausgabe

Die Visualisierungsdaten können über folgende CNC Objekte gelesen werden.

Indexgroup	Indexoffset	Datentyp	Beschreibung
0x2010<c>	0x2000	SOLLKONT_VISU_PDU	Datensatz aus kanalspezifischem Ausgabe- puffer (Fifo).
0x2010<c>	0x2001	UNS32	Anzahl der Datensätze im kanalspezifischen Ausgabefifo.
0x2010<c>	0x2002	SOLLKONT_VISU_PDU	Datensatz aus globalem Ausgabefifo.
0x2010<c>	0x2003	UNS32	Anzahl der Datensätze im globalen Ausgabe- fifo

Das gelesene Datenpaket hat dabei folgenden Aufbau:

	SOLLKONT_VISU_PDU
SGN32	count, Anzahl der Strukturen SOLLKONT_VISU_DATA_V0 ... SOLLKONT_VISU_DATA_V5 in der aktuellen Nachricht
UNS32	Versionskennung der Visualisierungsdaten siehe P-STUP-00039
SOLLKONT_VISU_DATA_V0	v0[MAX_SOLLKONT_VISU_DATA_COUNT_V0 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 0 hat.
Oder SOLLKONT_VISU_DATA_V1	v1[MAX_SOLLKONT_VISU_DATA_COUNT_V1 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 1 hat.
Oder SOLLKONT_VISU_DATA_V2	V2[MAX_SOLLKONT_VISU_DATA_COUNT_V2 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 2 hat.
Oder SOLLKONT_VISU_DATA_V3	v3[MAX_SOLLKONT_VISU_DATA_COUNT_V3 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 3 hat.
Oder SOLLKONT_VISU_DATA_V4	v4[MAX_SOLLKONT_VISU_DATA_COUNT_V4 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 4 hat.
Oder SOLLKONT_VISU_DATA_V5	V5[MAX_SOLLKONT_VISU_DATA_COUNT_V5 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 5 hat.
Oder SOLLKONT_VISU_DATA_V6	v6[MAX_SOLLKONT_VISU_DATA_COUNT_V6 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 6 hat.
Oder SOLLKONT_VISU_DATA_V7	v7[MAX_SOLLKONT_VISU_DATA_COUNT_V7 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 7 hat.
Oder SOLLKONT_VISU_DATA_V8	V8[MAX_SOLLKONT_VISU_DATA_COUNT_V8 - 1] Struktur mit Visualisierungsdaten wenn P-STUP-00039 den Wert 8 hat.
Oder SOLLKONT_VISU_DATA_V9	V9[MAX_SOLLKONT_VISU_DATA_COUNT_V9 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 9 hat.
Oder SOLLKONT_VISU_DATA_V10	V10[MAX_SOLLKONT_VISU_DATA_COUNT_V10 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 10 hat.
Oder SOLLKONT_VISU_DATA_V11	V11[MAX_SOLLKONT_VISU_DATA_COUNT_V11 - 1] Struktur mit Visualisierungsdaten, wenn P-STUP-00039 den Wert 11 hat.

	SOLLKONT_VISU_DATA_V0
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
SOLLKONT_VISU_ACHS_DATA_STD	Simu_achs_data_std[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	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] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_DATA_V2
SOLLKONT_VISU_CH_DATA_V1	Visu_data_v1
IF_FILE_NAME	Dateiname der NC-Programmdatei
SOLLKONT_VISU_ACHS_DATA_STD	Simu_achs_data_std[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_DATA_V3
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
SOLLKONT_VISU_ACHS_DATA_V1	Simu_achs_data_v1[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_DATA_V4
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
IF_FILE_NAME	Dateiname der NC-Programmdatei
SOLLKONT_VISU_ACHS_DATA_V1	Simu_achs_data_v1[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_DATA_V5
SOLLKONT_VISU_CH_DATA_V1	Visu_data_v1
IF_FILE_NAME	Dateiname der NC-Programmdatei
SOLLKONT_VISU_ACHS_DATA_V1	Simu_achs_data_v1[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_DATA_V6
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
SOLLKONT_VI-SU_ACHS_DATA_V2	Simu_achs_data_v2[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_DATA_V7
SOLLKONT_VISU_CH_DATA_STD	Visu_data_std
IF_FILE_NAME	Dateiname der NC-Programmdatei
SOLLKONT_VI-SU_ACHS_DATA_V2	Simu_achs_data_v2[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_DATA_V8
SOLLKONT_VISU_CH_DATA_V1	Visu_data_v1
IF_FILE_NAME	Dateiname der NC-Programmdatei
SOLLKONT_VI-SU_ACHS_DATA_V2	Simu_achs_data_v2[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_DATA_V9
SOLLKONT_VISU_CH_DATA_V2	Visu_data_v2
IF_FILE_NAME	Dateiname der NC-Programmdatei
SOLLKONT_VI-SU_ACHS_DATA_STD	Simu_achs_data_std[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_DATA_V10
SOLLKONT_VISU_CH_DATA_V2	Visu_data_v2
IF_FILE_NAME	Dateiname der NC-Programmdatei
SOLLKONT_VI-SU_ACHS_DATA_V1	Simu_achs_data_v1[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_DATA_V11
SOLLKONT_VISU_CH_DATA_V2	Visu_data_v2
IF_FILE_NAME	Dateiname der NC-Programmdatei
SOLLKONT_VISU_ACHS_DATA_V2	Simu_achs_data_v2[ANZ_SIMU_KOORD] Achsspezifische Visualisierungsdaten.

	SOLLKONT_VISU_CH_DATA_STD
SGN32	nc_satz_nr, Satznummer im NC-Programm
SGN32	fileoffset, Dateioffset von Dateianfang in Bytes >= 0 : gültiger Dateioffset bei aktivem Programm == -1 : Offset nicht gültig, da kein Programm aktiv
UNS16	channel_nr, Kanalnummer
SGN16	g_function >= 0 : G-Funktion : G0, G1, G2, G3, G61 für Polynomsätze == -1 : keine G-Funktion aktiv
UNS32	circle_radius, Radius in [0.1 µm] für G2 / G3-Sätze
REAL64	circle_center_point[2] Absolutposition des Kreismittelpunkts in der aktiven Bearbeitungsebene (G17,G18,G19) in [0.1 µm] für G2 / G3 Sätze (ab CNC-Version V2.10.1032.03 bzw. V2.10.1505.05)

	SOLLKONT_VISU_CH_DATA_V2
SGN32	Nc_satz_nr, Satznummer im NC-Programm
SGN32	fileoffset, Dateioffset von Dateianfang in Bytes >= 0 : gültiger Dateioffset bei aktivem Programm == -1 : Offset nicht gültig, da kein Programm aktiv
UNS16	channel_nr, Kanalnummer
SGN16	g_function >= 0 : G-Funktion : G0, G1, G2, G3, G61 für Polynomsätze == -1 : keine G-Funktion aktiv
UNS32	circle_radius, Radius in [0.1 µm] für G2 / G3-Sätze
REAL64	circle_center_point[2] Absolutposition des Kreismittelpunkts in der aktiven Bearbeitungsebene (G17,G18,G19) in [0.1 µm] für G2 / G3 Sätze (ab CNC-Version V2.10.1032.03 bzw. V2.10.1505.05)
SGN32	vb_prog, programmierte Bahngeschwindigkeit
SOLLKONT_VISU_DATA_TECHNO_V1	Techno_v1, Technologieinformationen
UNS32	Fillup (Alignmentdaten)

	SOLLKONT_VISU_DATA_TECHNO_V1
UNS16	Axis_number, Achsnummer der Achse, auf der die Technologieinformationen ausgegeben wurden. Eine Achsnummer von 0 bedeutet, dass die Technologieinformationen auf der Kanalschnittstelle ausgegeben wurden
UNS16	Fillup, benutzt um Strukturausrichtung zu erzwingen
UNS32	m_h_count, Anzahl der belegten Einträge im Vektor m_h_data[]
SOLLKONT_M_H_PROCESS_V1	M_h_data_v1[MAX_M_H_DATA] Vektor mit Informationen über M/H-Funktionen
UNS32	S_count, Anzahl der Einträge im Vektor s_proc[]
SOLLKONT_S_PROCESS	s_proc[] Vektor mit Informationen über Spindel-Funktionen.
SGN32	vb_prog, programmierte Bahngeschwindigkeit



Hinweis

Über den Dateioffset kann erkannt werden, ob ein Programm in Bearbeitung ist bzw. beendet wurde.

Eine ungültige G-Funktion (-1) wird z.B. durch eine NC-Zeile mit einer M-Funktion ausgelöst.

	IF_FILE_NAME
ISG_CHAR	file_name[128] Dateiname des aktuellen NC-Programms. Für die zusätzliche Ausgabe des Dateinamens muss als Versionskennung der Anzeigedaten „contour_visu_ifc_version“ (P-STUP-00039) auf den Wert 1, 2, 4, oder 5 gesetzt sein (ab CNC-Version V2.10.1032.08 bzw. V2.10.1507.06).

	SOLLKONT_VISU_ACHS_DATA_STD
SGN32	Akt_sollwert, aktueller Sollposition der Achse in [0.1 µm]
UNS16	Log_achs_nr, logische Achsnummer
UNS16	<alignment bytes>

	SOLLKONT_VISU_ACHS_DATA_V1
SGN32	Akt_sollwert, aktueller Sollposition der Achse in [0.1 µm]
SGN32	Akt_sollwert_wcs0, aktueller Sollposition der Achse im WCS0-System in [0.1 µm]. Dieser Wert wird nur berechnet, wenn Kanalparameter P-CHAN-00145 den Wert 1 hat und dem Kanalparameter P-CHAN-00032 ein Wert > 0 zugewiesen wird.
UNS16	Log_achs_nr, logische Achsnummer
UNS16	<alignment bytes>

	SOLLKONT_VISU_ACHS_DATA_V2
SGN32	Akt_sollwert, aktuelle Sollposition der Achse in [0.1 µm]
SGN32	Akt_sollwert_wcs0, aktuelle Sollposition der Achse im WCS0-System in [0.1 µm]. Dieser Wert wird nur berechnet, wenn Kanalparameter P-CHAN-00145 den Wert 1 hat und dem Kanalparameter P-CHAN-00032 ein Wert > 0 zugewiesen wird.
SGN32	Akt_sollwert_wcs, aktuelle Sollposition der Achse im WCS-System in [0.1 µm]. Dieser Wert wird nur berechnet, wenn Kanalparameter P-CHAN-00145 den Wert 1 hat und dem Kanalparameter P-CHAN-00032 ein Wert > 0 zugewiesen wird.
UNS16	Log_achs_nr, logische Achsnummer
UNS16	<alignment bytes>

	SOLLKONT_VISU_CH_DATA_V1
SGN32	Nc_satz_nr, Satznummer im NC-Programm
SGN32	fileoffset, Dateioffset von Dateianfang in Bytes >= 0 : gültiger Dateioffset bei aktivem Programm == -1 : Offset nicht gültig, da kein Programm aktiv
UNS16	channel_nr, Kanalnummer
SGN16	g_function >= 0 : G-Funktion : G0, G1, G2, G3, G61 für Polynomsätze == -1 : keine G-Funktion aktive
UNS32	circle_radius, Radius in [0.1 µm] für G2 / G3-Sätze
REAL64	circle_center_point[2] Absolutposition des Kreismittelpunkts in der aktiven Bearbeitungsebene (G17,G18,G19) in [0.1 µm] für G2 / G3 Sätze (ab CNC-Version V2.10.1032.03 bzw. V2.10.1505.05)
SGN32	vb_prog, programmierte Bahngeschwindigkeit
SOLLKONT_VISU_DATA_TECHNO	techno, Technologieinformationen

	SOLLKONT_VISU_DATA_TECHNO
UNS16	Axis_number, Achsnummer der Achse, auf der die Technologieinformationen ausgegeben wurden. Eine Achsnummer von 0 bedeutet, dass die Technologieinformationen auf der Kanalschnittstelle ausgegeben wurden
UNS16	Fillup, benutzt um Strukturausrichtung zu erzwingen
UNS32	m_h_count, Anzahl der belegten Einträge im Vektor m_h_data[]
SOLLKONT_M_H_PROCESS	M_h_data[MAX_M_H_DATA] Vektor mit Informationen über M/H-Funktionen
UNS32	S_count, Anzahl der Einträge im Vektor s_proc[]
SOLLKONT_S_PROCESS	s_proc[] Vektor mit Informationen über Spindel-Funktionen.
SOLLKONT_TOOL_ID	Tool, Informationen über die aktuell gültige Werkzeugnummer

	SOLLKONT_M_H_PROCESS
UNS32	nr, Nummer der M-/H-Funktion
UNS32	sync, Synchronisationsart der M-/H-Funktion, siehe [CHAN// Konfiguration von SPS Funktionen]
UNS32	type, 1 = M-Funktion, 2 = H-Funktion

	SOLLKONT_S_PROCESS	
UNS16	Ax_nr, Achsnummer der Spindelachse	
UNS16	Cmd, Spindelkommando:	
	Wert	Kommando
	3	M3
	4	M4
	5	M5
	19	M19
UNS32	Sync, Synchronisationsart der Spindelfunktion	
SGN32	Position; Zielposition in 0,1 um bei Positionierbewegung	
SGN32	Revolution, Sollgeschwindigkeit der Spindel in 10E-3°/s bzw.um/s.	

	SOLLKONT_TOOL_ID	
SGN32	Basic, Basisnummer des Werkzeugs	
SGN32	Sister, Schwesternummer des Werkzeugs, -1 bedeutet nicht belegt.	
SGN32	Variant, Variantennummer des Werkzeugs, -1 bedeutet nicht belegt.	

	SOLLKONT_M_H_PROCESS_V1	
UNS32	nr, Nummer der M-/H-Funktion	
UNS32	sync, Synchronisationsart der M-/H-Funktion, siehe [CHAN// Konfiguration von SPS Funktionen]	
UNS32	type, 1 = M-Funktion, 2 = H-Funktion	
SGN32	Add_value, In M/H-Funktion programmierter zusätzlicher wert.	

Konstanten

Konstante	Wert
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

Datentyp	C Datentyp	Beschreibung
SGN16	signed short	Vorzeichenbehaftete 16 Bit Ganzzahl
UNS16	unsigned short	Vorzeichenlose 16 Bit Ganzzahl
SGN32	signed long	Vorzeichenbehaftete 32 Bit Ganzzahl
UNS32	unsigned long	Vorzeichenlose 32 Bit Ganzzahl
REAL64	double	64 Bit Fließkommazahl
ISG_CHAR	char	8 Bit Text-Zeichen

4.2.1

HLI- Kanalbetriebsart bis CNC-Version V2.20xx

Kanalbetriebsart			
Beschreibung	Anwahl einer speziellen Kanalbetriebsart, wie z.B. Syntaxcheck oder Fertigungszeitberechnung		
Datentyp	MCControlSGN32Unit, s. Beschreibung Control Unit		
Zugriff	PLC liest Request + State und schreibt Command + Enable		
ST-Pfad	pMC[channel_idx]^^.addr^.MCControlDecoder_Data. MCControlSGN32Unit_Execution-Mode		
Kommandierter, angeforderter und Rückgabewert			
ST-Element	.D_Command .D_Request .D_State		
Datentyp	DINT		
Wertebereich	Wert	Konstante	Bedeutung
	0x0000	ISG_STANDARD	Normalbetrieb
	0x0001	SV	Satzvorlauf
	0x0002	SOLLKON	Simulation Sollkonturvisualisierung mit Ausgabe der Visualisierungsdaten
	0x0802	SOLLKON_SU-PRESS_OUTPUT & SOLLKON	Simulation Sollkonturvisualisierung ohne Ausgabe der Visualisierungsdaten
	0x0004	ON_LINE	Simulation Onlinevisualisierung
	0x0008	SYNCHK	Simulation Syntaxcheck
	0x0010	PROD_TIME	Simulation Fertigungszeitberechnung (bei TwinCAT ohne Funktion)
	0x0020	ONLINE_PROD_TIME	Simulation Online-Fertigungszeitberechnung
	0x0040	MACHINE_LOCK	Dry Run ohne Achsbewegung
	0x0080	ADD_MDI_BLOCK	Erweiterter Handsatzmodus: das Ende eines Handsatzes wird nicht als Programmende gewertet, sondern erlaubt es weitere Handsätze zu beauftragen.
	0x0100	KIN_TRAFO_OFF	Überschreibt die automatische Freischaltung für kinematische Transformationen durch eine in den Kanalparametern (sda_mds*.lis) definierte Kenngröße
	0x1000	BEARB_MODE_SCENE	Durch das Einschalten des SZENE-Modus wird die Ausgabe der #SCENE-Befehle auf der Schnittstelle eingeschaltet (s.a. [FCT-C17// Szene Konturvisualisierung [► 40]]). Die Anbindung eines weiteren Clients an diese Ausgabe findet über die DataFactory / CORBA statt.

	0x2000	SUPPRESS_TECH- NO_OUTPUT	Ohne Ausgabe von Technofunktionen (M/H/T). Wird implizit gesetzt in Verbindung mit Syntax- check
Umleitung			
ST-Element	.X_Enable		

4.3 Applikationsbeispiele

Die im vorhergehenden Abschnitt beschriebenen Visualisierungsdaten können zum Beispiel mit Hilfe der folgenden Applikationen gelesen werden.

ADS-Zugriff via AmsAdsDebugger

An einer laufenden TwinCAT-Steuerung können über den AmsAdsViewer die einzelnen Parameter der Simulation direkt getestet werden.

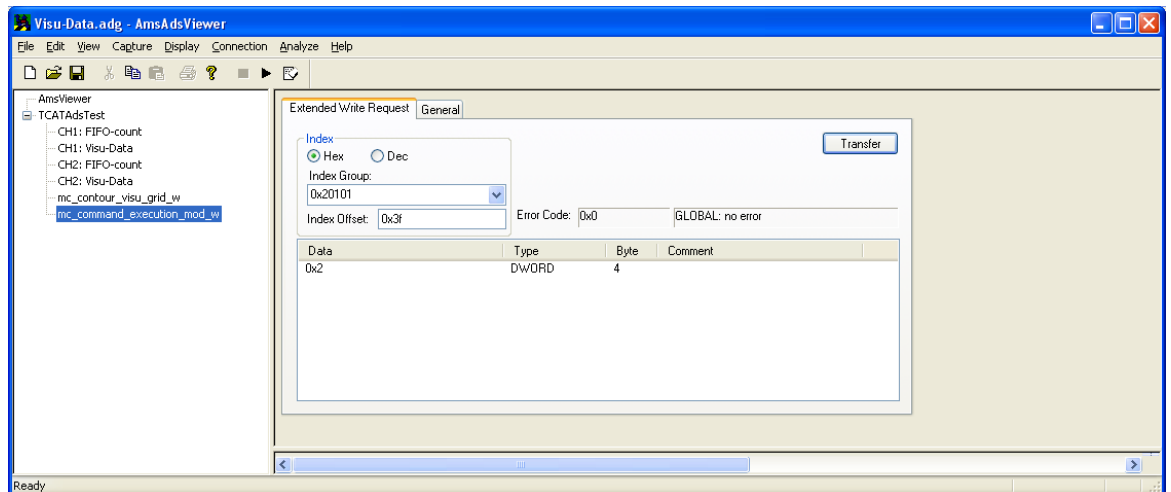


Abb. 6: ADS-Zugriff via AmsAdsDebugger

Zugriffe über den CNC Objekt-Browser

Die Sollkonturvisualisierung wird als Bearbeitungsmodus vor dem Programmstart durch die Bedienung eingestellt. Diese Einstellung wird über eine sogenannte Control-Unit an die SPS weitergegeben, welche diese zulassen oder ablehnen kann.

Ebenso hat die SPS hierdurch die Möglichkeit, den Bearbeitungsmodus = Schnelle Konturvisualisierung ohne vorherige HMI-Anforderung selbst anzuwählen.

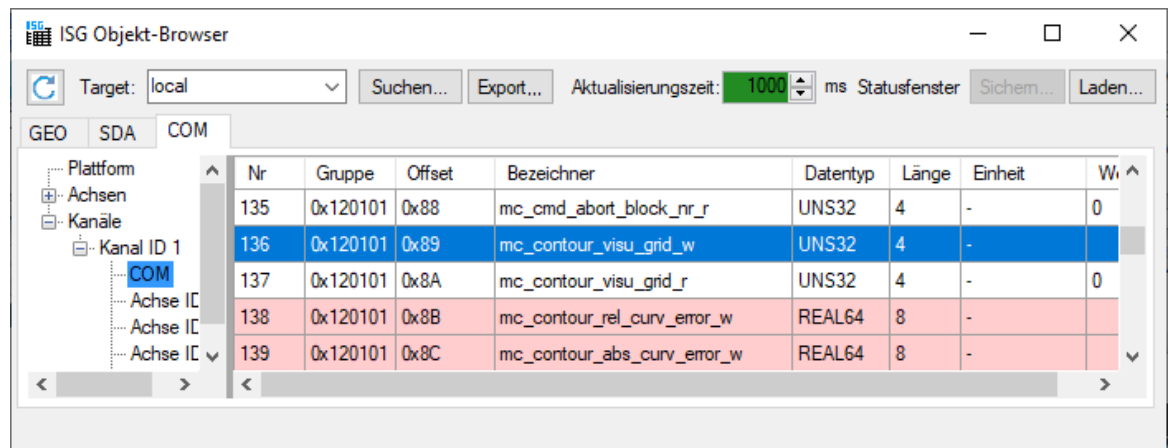


Abb. 7: Zugriff über CNC Objekt-Browser

ADS-Zugriff via Win32 Application



Programmierbeispiel

ADS-Zugriff 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 output fifo
    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;
}
```

Anzeigen von Achspositionen mit DXF Format



Programmierbeispiel

Anzeigen von Achspositionen mit 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
```

Die ausgelesenen Achspositionen können genutzt werden, um die abgefahrte Kontur im DXF-Format darzustellen.

DXF-Ausgabedatei

```
0
SECTION
2
HEADER
999
isg.dxf
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
...
```

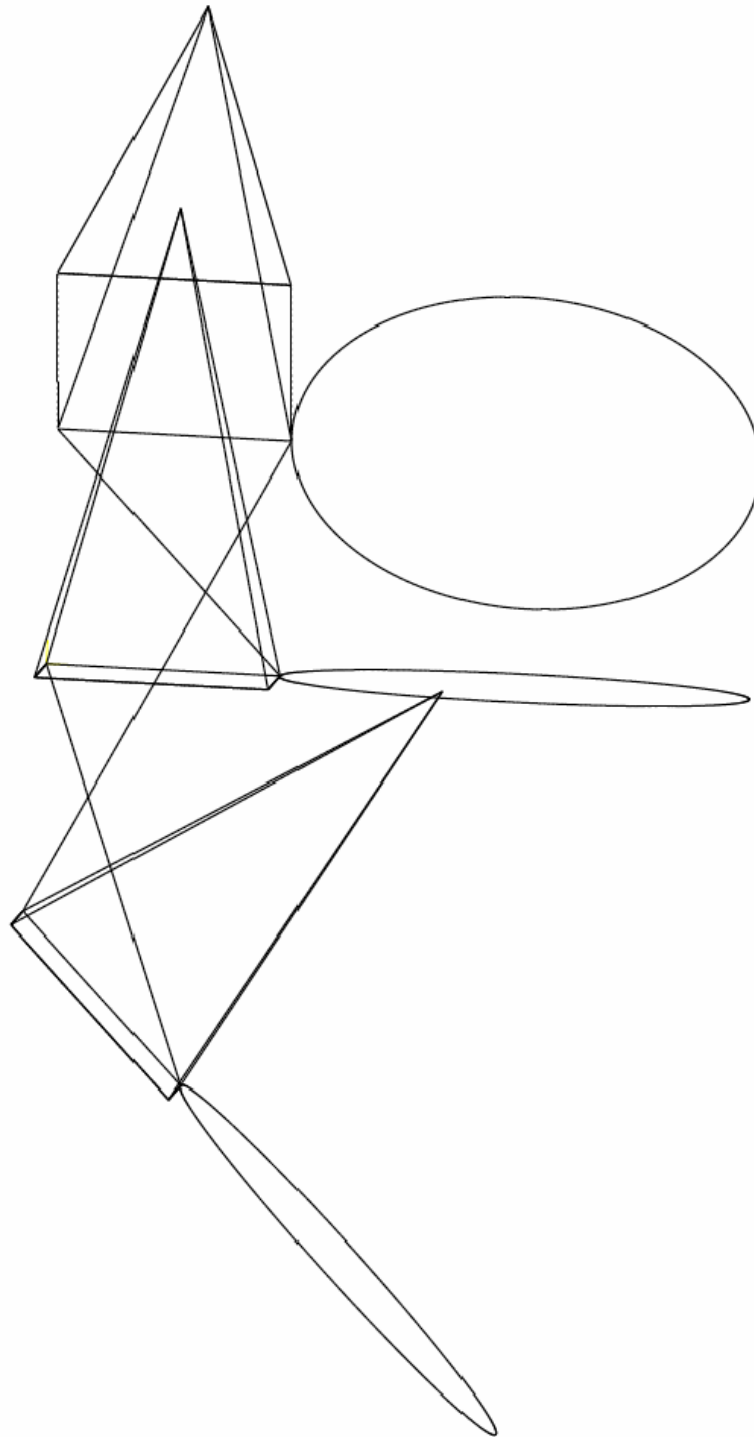


Abb. 8: Darstellung der DXF-Ausgabedatei in einem Viewer

5 Online Konturvisualisierung

Aktivierung

Die **Online Konturvisualisierung** wird aktiviert, indem beim Programmstart die Programmstartoption **ON_LINE** auf dem HLI an die Steuerung übergeben wird (siehe Dokumentation zur HLI [► 18]).

Im Gegensatz zur Schnellen Konturvisualisierung erfolgt im Modus **Online Konturvisualisierung** eine reale Abarbeitung des NC-Programms. Die ausgegebenen Werte werden gefiltert, um eine performante Datenübertragung zu erhalten.

Rasterung

In dieser Betriebsart kann eine Datenreduktion erfolgen. Dazu kann – abhängig vom verwendeten Bewegungssatz (gerade/gekrümmt) – das Stützpunktraster für die Interpolation entweder

- durch Angabe eines maximalen Stützpunktabstands
- oder eines maximalen Bahnfehlers angegeben werden (siehe unter „Anwahl über HMI / ADS“).

Dies lässt sich in folgenden Parametern festlegen:

Parameter	Format	Beschreibung	Index-Group	Index-Offset
mc_contour_visu_grid_w mc_contour_visu_grid_r	UNS32	Streckenraster für Sollkonturvisualisierung für Linearsätze (G00/G01) in [0.1 µm]	0x2010<c> c Element [1; max. Kanal]	0x89, 0x8a
mc_contour_rel_curv_error_w	REAL64	Maximaler relativer Bahnfehler in [0.1%] für Sollkonturvisualisierung für Kreise oder Polynome	0x2010<c> c Element [1; max. Kanal]	0x8b
mc_contour_abs_curv_error_w	REAL64	Maximaler absoluter Bahnfehler in [0.1 µm] für Sollkonturvisualisierung	0x2010<c> c Element [1; max. Kanal]	0x8c

Parametrierung

Die Parametrierung erfolgt analog zur Schnellen Konturvisualisierung (siehe „Parameter [► 42]“.)

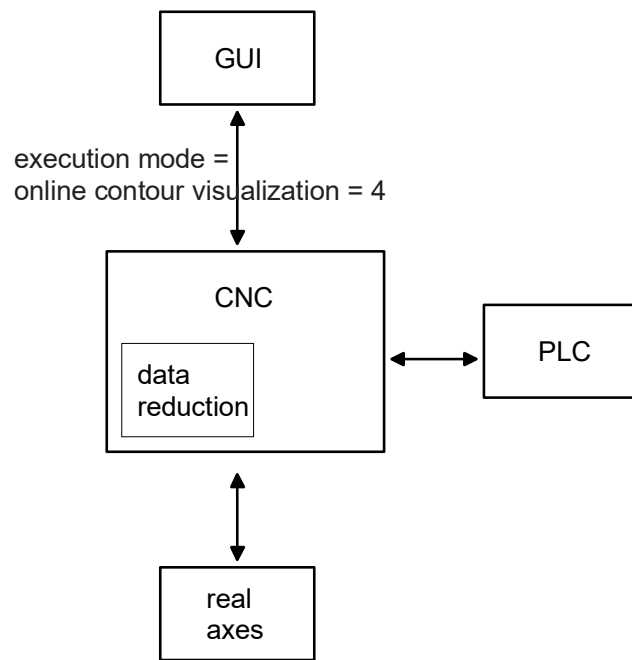


Abb. 9: Online Konturvisualisierung

6 Szene Konturvisualisierung

Im **Szene-Modus** erfolgt eine reale Abarbeitung des NC-Programms, wobei die ausgegebenen Werte zeitlich gefiltert werden. Es kann die gewünschte Datenrate als "Bilder pro Sekunde" angegeben werden.

Aktivierung

Die **Szene Konturvisualisierung** wird aktiviert, indem bei Programmstart die Programmstartoption **BEARB_MODE_SCENE** auf dem HLI oder der Oberfläche an die Steuerung übergeben wird.

Protokollierung

Bei der **Szene**-Anzeige werden die Bewegungen jedes Koordinatensystems der kinematischen Kette protokolliert. Somit kann die Bewegung jedes grafischen Körpers visualisiert werden. Zusätzlich lässt sich diese Bewegung als Spur visualisieren.

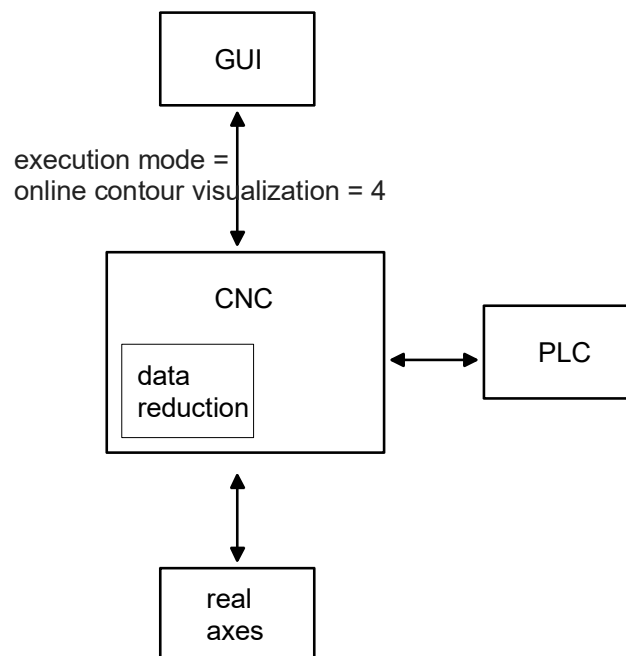


Abb. 10: Szene Konturvisualisierung

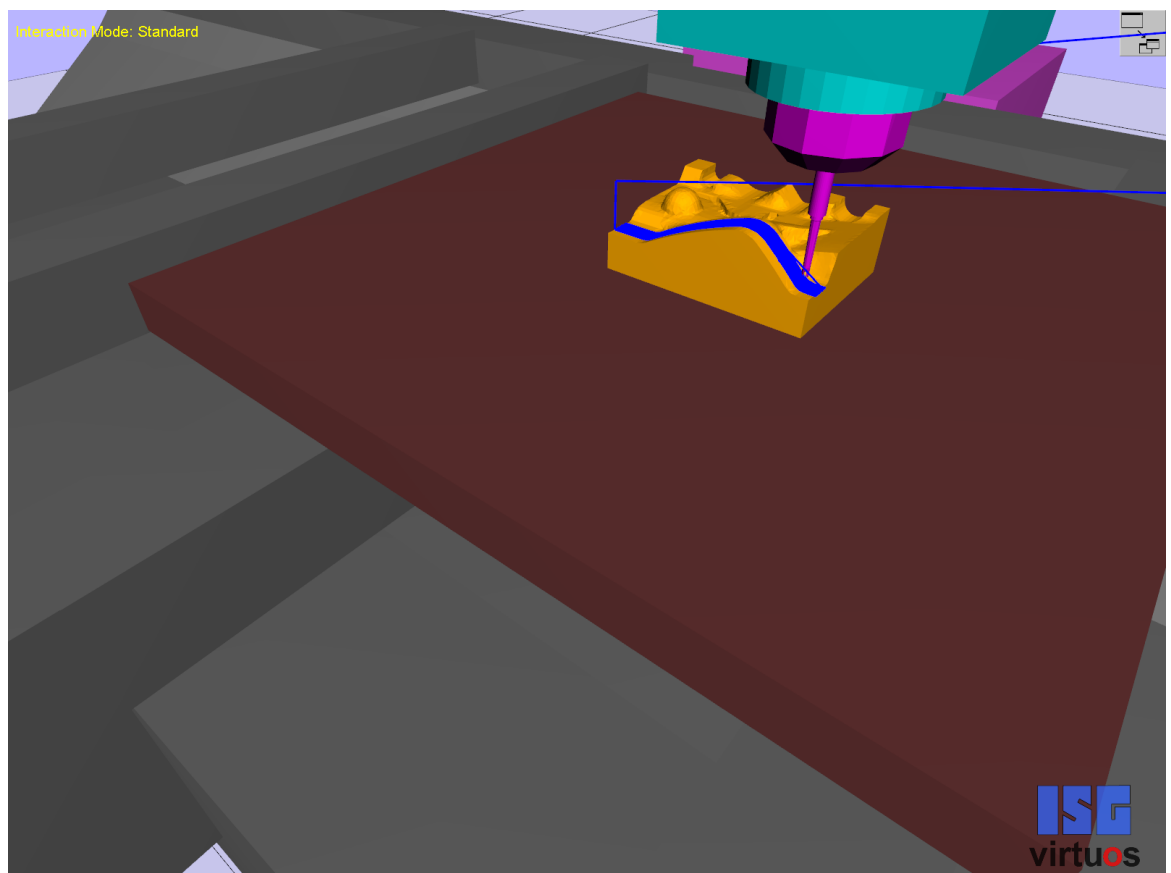
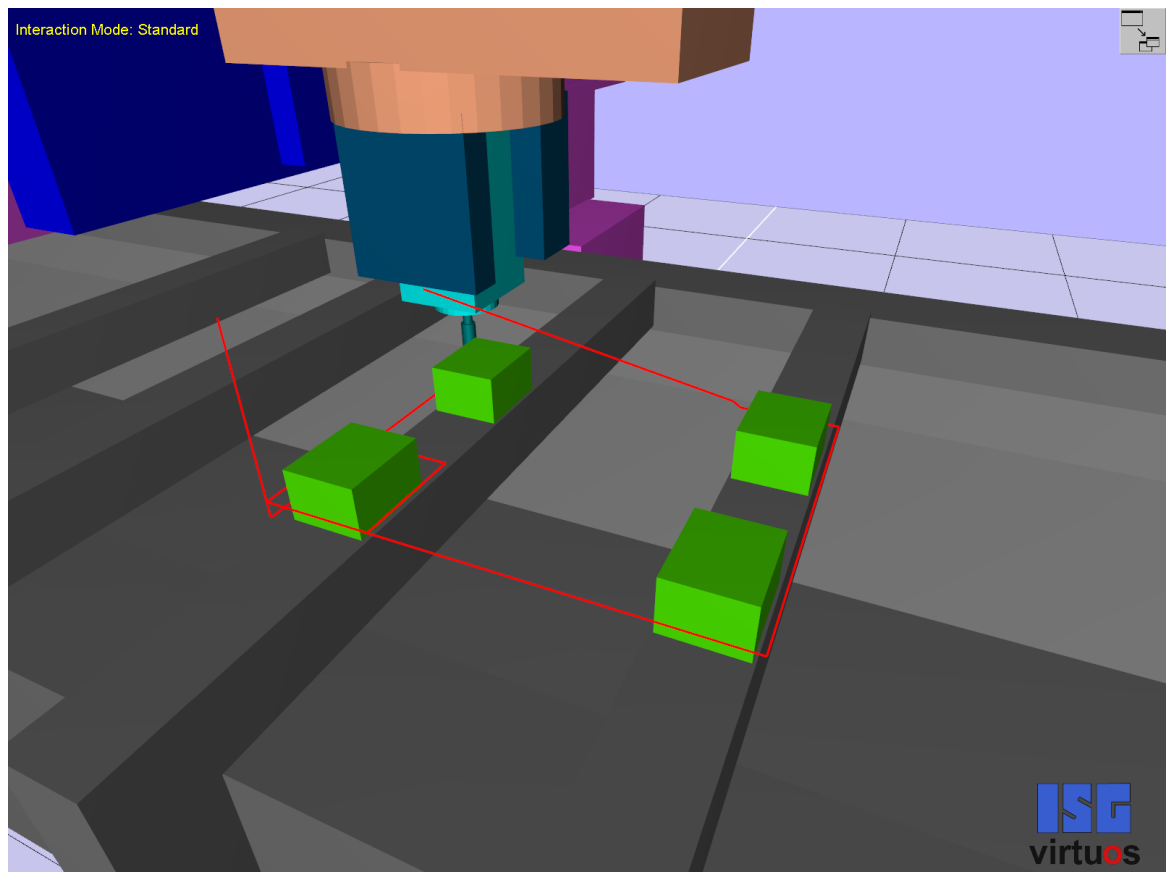


Abb. 11: Beispiele für die Konturvisualisierung mit dem Szene-Modus

7 Parameter

7.1 Übersicht

ID	Parameter	Beschreibung
P-CHAN-00121	simu_output_wcs	Anzeigeformat bei der Bearbeitungssimulation
P-CHAN-00183	simu_ignore_internal_stop_cond	Ignorieren der internen Stoppbedingungen bei schneller Konturvisualisierung
P-STUP-00040	single_protocol_fifo	Sammel- oder kanalspezifische Ausgabe der Anzeigedaten
P-STUP-00039	contour_visu_ifc_version	Versionskennung der Visualisierungsdaten

7.2 Beschreibung

P-CHAN-00121	Anzeigeformat bei der Bearbeitungssimulation
Beschreibung	Durch den Parameter kann das Format der Anzeigedaten an der Schnittstelle zur Bearbeitungssimulation bzgl. des Koordinatensystems umgeschaltet werden.
Parameter	simu_output_wcs
Datentyp	BOOLEAN
Datenbereich	0: Anzeige der Achskoordinaten einschließlich Versätze (Maschinenkoordinaten). 1: Anzeige der Absolutkoordinaten ohne Versätze (Programmierte Koordinaten).
Dimension	----
Standardwert	0
Anmerkungen	

P-CHAN-00183	Ignorieren der internen Stoppbedingungen bei schneller Konturvisualisierung
Beschreibung	Durch den Parameter kann das Anhalten des CNC-Programms aufgrund von internen Stoppbedingungen (z. B. M00) bei der schnellen Konturvisualisierung verhindert werden.
Parameter	simu_ignore_internal_stop_cond
Datentyp	BOOLEAN
Datenbereich	0: Interne Stoppbedingungen sind wirksam (Standard). 1: Interne Stoppbedingungen werden ignoriert.
Dimension	----
Standardwert	0
Anmerkungen	

P-STUP-00040	Sammel- oder kanalspezifische Ausgabe der Anzeigedaten
Beschreibung	Der Parameter definiert, ob die Visualisierungsdaten pro Kanal in einen Ausgabefifo geschrieben werden, oder ob die Visualisierungsdaten aller Kanäle in einen globalen Ausgabefifo geschrieben werden.
Parameter	single_protocol_fifo
Datentyp	BOOLEAN
Datenbereich	0: Kanalspezifische Ausgabe der Visualisierungsdaten. 1: Gemeinsame Ausgabe der Visualisierungsdaten.
Dimension	----
Standardwert	0 *
Anmerkungen	* 1 ab CNC-Version V3.01.3038

P-STUP-00039	Versionskennung der Visualisierungsdaten	
Beschreibung	<p>Durch den Parameter kann der Typ der Datenstruktur, den die Konturvisualisierung ([FCT-C17 ▶ 6]) liefert eingestellt werden.</p> <p>Abhängig von der gewählten Einstellung werden mehr oder weniger Visualisierungsdaten erzeugt.</p> <p>Eine Übersicht der vorhandenen Datenstrukturen findet sich in [FCT-C17 ▶ 6].</p>	
Parameter	contour_visu_ifc_version	
Datentyp	UNS32	
Datenbereich	contour_visu_ifc_version	Datenstruktur
	0	SOLLKONT_VISU_DATA_V0 (Standard)
	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	----	
Standardwert	0	
Anmerkungen		

8 Anhang

8.1 Anregungen, Korrekturen und neueste Dokumentation

Sie finden Fehler, haben Anregungen oder konstruktive Kritik? Gerne können Sie uns unter documentation@isg-stuttgart.de kontaktieren. Die aktuellste Dokumentation finden Sie in unserer Onlinehilfe (DE/EN):



QR-Code Link: <https://www.isg-stuttgart.de/documentation-kernel/>

Der o.g. Link ist eine Weiterleitung zu:

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



Hinweis

Mögliche Änderung von Favoritenlinks im Browser:

Technische Änderungen der Webseitenstruktur betreffend der Ordnerpfade oder ein Wechsel des HTML-Frameworks und damit der Linkstruktur können nie ausgeschlossen werden.

Wir empfehlen, den o.g. „QR-Code Link“ als primären Favoritenlink zu speichern.

PDFs zum 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

Stichwortverzeichnis

B

Betriebsart

Kanal 18, 30

K

Kanal

Betriebsart 18, 30

P

P-CHAN-00121 42

P-CHAN-00183 42

P-STUP-00039 43

P-STUP-00040 43



© Copyright
ISG Industrielle Steuerungstechnik GmbH
STEP, Gropiusplatz 10
D-70563 Stuttgart
Alle Rechte vorbehalten
www.isg-stuttgart.de
support@isg-stuttgart.de

