



DOKUMENTATION ISG-kernel

McCOM - Online Adaptation

Kurzbezeichnung:
McCOM - OAD

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

Dokumentation Version: 1.01
01.12.2025

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

ISG®, ISG kernel®, ISG virtuos®, ISG dirigent® und TwinStore® sowie die entsprechenden 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	7
2 Online Werkzeugradiuskompensation	8
2.1 Übersicht	8
2.2 Beschreibung	9
2.2.1 Befehlsübersicht	13
2.2.2 Eigenschaften der Werkzeugradiuskorrektur	14
2.2.2.1 Verhalten an Satzübergängen mit Winkeln größer 180°	15
2.2.2.2 Verhalten an Satzübergängen mit Winkeln kleiner 180°	18
2.3 Anbindung einer Online-Werkzeugradiuskompensation via TcCOM	20
2.3.1 Methoden der Schnittstelle	20
2.3.2 Instanzdaten	20
2.3.3 Konfigurieren und Anmelden des TcCOM-Objekts	22
2.4 Programmierung	23
2.4.1 TRC Optionen für Online-TRC und 2-Pfad	24
2.4.2 TRC Option GEN_CIR_BLOCK_IN_CORNER	27
2.4.3 TRC Option G236_LIN	27
2.4.4 TRC Option PERPENDICULAR_RADIUS_CHANGE	28
2.5 Parameter	33
2.6 Fehlermeldungen	34
3 Geometrische Vorschubanpassung	35
3.1 Übersicht	35
3.2 Beschreibung	36
3.2.1 Integration in den NC-Kanal	37
3.3 Anbindung einer geometrischen Vorschubanpassung via TcCOM	38
3.3.1 Methoden der Schnittstelle	38
3.3.2 Instanzdaten	38
3.3.3 Konfigurieren und Anmelden des TcCOM-Objekts	39
3.4 Programmierung	40
3.5 Parameter	41
3.6 Fehlermeldungen	43
4 Dynamische Konturvorsteuerung	44
4.1 Übersicht	44
4.2 Beschreibung	45
4.2.1 Externe Berechnungsmodus	47
4.3 Programmierung	49
4.4 Anbindung einer dynamischen Konturvorsteuerung via TcCOM	50
4.4.1 Methoden der Schnittstelle	50
4.4.2 Instanzdaten	51
4.4.3 Konfigurieren und Anmelden des TcCOM-Objekts	52
4.5 Parameter	53
4.5.1 Parametrierbeispiel	54

4.6	Fehlermeldungen	55
5	Erstellungsablauf eines TcCOM-Objekts	56
5.1	Neues Projekt anlegen	56
5.2	Geometrische Vorschubanpassung	59
5.2.1	Erstellen eines Objekts für geometrische Vorschubanpassung	59
5.2.2	Objekt der geometrischen Vorschubanpassung einbinden	61
5.3	Dynamische Konturvorsteuerung	63
5.3.1	Erstellen eines Objekts für dynamische Konturvorsteuerung	63
5.3.2	Objekt der dynamischen Konturvorsteuerung einbinden	65
5.4	Online Werkzeugradiuskompensation	68
5.4.1	Erstellen eines Objekts für Online-Werkzeugradiuskompensation	68
5.4.2	Objekt der Online-Werkzeugradiuskorrektur einbinden	71
6	Anhang	73
6.1	Anregungen, Korrekturen und neueste Dokumentation	73
	Stichwortverzeichnis	74

Abbildungsverzeichnis

Abb. 1:	Platzierung der Online-Werkzeugradiuskorrektur im NC-Kanal	9
Abb. 2:	Prinzipieller Ablauf im NC-Kanal	10
Abb. 3:	Ablauf 2-Pfadprogrammierung mit Online TRC	11
Abb. 4:	2-Pfad-Anwendung mit schräg stehendem Draht	12
Abb. 5:	Werkstück mit statischer Werkzeugradiuskorrektur	14
Abb. 6:	Tangentendrehung an der Außenecke einer Geometrie	15
Abb. 7:	Online TRC Bahn mit schräg stehendem Werkzeug an einer Außenecke	16
Abb. 8:	Änderung des Werkzeugradius an einer Außenecke	17
Abb. 9:	Verhalten an der Innenecke einer Geometrie an Satzübergängen mit Winkel 90°	18
Abb. 10:	Änderung des Werkzeugradius an der Innenecke einer Geometrie an Satzübergängen mit Winkel kleiner 180°	19
Abb. 11:	Orthogonales Ausfahren der Änderung des Werkzeugradius	28
Abb. 12:	Geometrische Vorschubanpassung im NC-Kanal	36
Abb. 13:	Integration des Objekts im NC-Kanal	37
Abb. 14:	Geometrische Wirkungsweise der programmierten und korrigierten Bahn	45
Abb. 15:	Anordnung der dynamischen Konturvorststeuerung im System	46
Abb. 16:	Dynamische Konturvorststeuerung mit impliziter Überlagerung	47
Abb. 17:	Dynamische Konturvorststeuerung mit expliziter Offsetüberlagerung in erster Kinematikstufe	48
Abb. 18:	Erstellung eines neuen Projekts	56
Abb. 19:	Konfiguration des neuen Projekts	57
Abb. 20:	Anlegen einer CNC-Konfiguration	57
Abb. 21:	Anlegen eines Kanals	58
Abb. 22:	Anlegen einer Achse	58
Abb. 23:	Treiberprojekt für geometrische Vorschubanpassung	59
Abb. 24:	Klasse festlegen	59
Abb. 25:	Benennung der Klasse	60
Abb. 26:	Treiber erstellen	60
Abb. 27:	Integration des TcCOM-Objekts	61
Abb. 28:	Eigenschaften des TcCOM-Objekts	61
Abb. 29:	Treiberprojekt für dynamische Kontursteuerung anlegen	63
Abb. 30:	Klasse festlegen	64
Abb. 31:	Benennung der Klasse	64
Abb. 32:	Treiber erstellen	65
Abb. 33:	Integration des TcCOM-Objekts	65
Abb. 34:	Eigenschaften des TcCOM-Objekts	66
Abb. 35:	Treiberprojekt für Online-Werkzeugradiuskompensation	68
Abb. 36:	Transformationsklasse festlegen	69
Abb. 37:	Benennen der Transformationsklasse	69
Abb. 38:	Treiber erstellen	70
Abb. 39:	Integration des TcCOM-Objekts	71
Abb. 40:	Eigenschaften des TcCOM-Objekts	71

1 Übersicht

Das Dokument stellt eine Zusammenfassung der folgenden drei Funktionalitäten dar:

- Online Werkzeugradiuskompensation
- Dynamische Konturvorsteuerung
- Geometrische Vorschubanpassung

Aufgabe

Die Online Werkzeugradiuskompensation ermöglicht die Integration technologiespezifischer Werkzeugkorrekturen in Echtzeit, besonders geeignet für Drahterodieren mit schräg stehendem Draht in 2-Pfad-Anwendungen.

Die „Dynamische Konturvorsteuerung“ (DCC) kompensiert Abweichungen zwischen der programmierten und der real gefertigten Kontur, die beispielsweise durch physikalische Verformungen eines Erodierdrahts entstehen können. Ziel ist es, die Istkontur an die Sollkontur anzupassen.

Die „Geometrische Vorschubanpassung“ ermöglicht es dem Anwender, den Vorschub durch eigene (geometrieabhängige) Algorithmen zu beeinflussen. Dies dient insbesondere beim Drahterodieren dazu, einen konstanten Flächenabtrag sicherzustellen.

Eigenschaften

Online Werkzeugradiuskompensation: Wirksam nur bei aktiver Werkzeugradiuskorrektur (G41/G42). Erfordert eine C2-stetige Parallelbahn aus der statischen Werkzeugradiuskorrektur. Unterstützt 2-Pfad-Konfigurationen mit unabhängiger Dynamikplanung und synchronisierter Ausgabe.

Dynamische Konturvorsteuerung: Modifiziert die interpolierte Werkzeugmittelbahn basierend auf aktuellen und vorherigen Konturelementen, Werkzeugparametern und SPS-Einfluss (z.B. Geschwindigkeitsabhängigkeit).

Geometrische Vorschubanpassung: Berechnung eines Faktors, der mit der aktuellen Sollgeschwindigkeit und dem Overridefaktor multipliziert wird. Die Berechnung erfolgt auf Basis einer zyklisch korrigierten Radiusmittelpunktbahn, wobei der Werkzeugradius aus dem effektiven Werkzeugradius und einem Spaltwert besteht.

Programmierung

Die genaue Programmierung bitte den jeweiligen Unterkapiteln entnehmen:

- Online Werkzeugradiuskompensation [▶ 23]
- Dynamische Konturvorsteuerung [▶ 49]
- Geometrische Vorschubanpassung [▶ 40]

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 Online Werkzeugradiuskompensation

2.1 Übersicht

Aufgabe

Mit der Funktionalität Online-Werkzeugradiuskompensation hat der Anwender die Möglichkeit, technologiespezifische Werkzeugkorrekturen zu integrieren.

Speziell im Bereich Drahterodieren bei schräg stehendem Draht in 2-Pfadanwendungen ist diese Funktionalität geeignet.



Versionshinweis

Diese Funktionalität ist ab CNC-Version V3.1.3108 verfügbar.



Hinweis

Diese Funktionalität ist Bestandteil einer lizenzpflichtigen Zusatzoption.

Eigenschaft

Die Online-TRC (TRC = Tool Radius Compensation) ist nur wirksam bei aktiver Werkzeugradiuskorrektur. Die Aktivierung erfolgt mit G41 oder G42, die Deaktivierung mit G40.

Parameter

Der Parameter P-CHAN-00550 [▶ 33] muss bei Verwendung einer 2-Pfadanwendung gesetzt werden.

Programmierung

Die Aktivierung und Deaktivierung der Funktionalität erfolgt mit den Standardbefehlen der Werkzeugradiuskorrektur (G40/G41/G42). Die entsprechende Definition der Online-Werkzeugradiuskorrektur erfolgt mit dem #TRC-Befehl [▶ 24].

Verweise 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.

2.2 Beschreibung

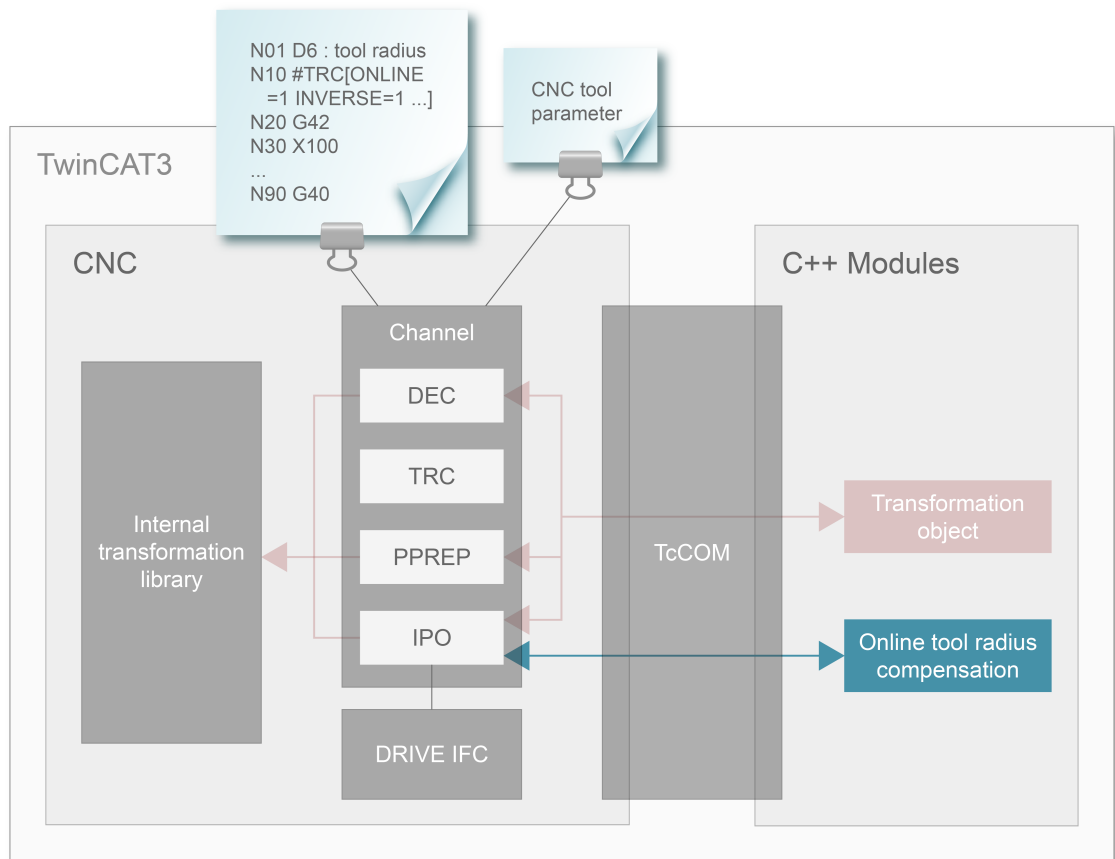


Abb. 1: Platzierung der Online-Werkzeugradiuskorrektur im NC-Kanal

Voraussetzung:

Für die Nutzung der Online-Werkzeugradiuskorrektur muss die erzeugte Parallelbahn der statischen Werkzeugradiuskorrektur im Nicht-Echtzeitteil der CNC eine C2-stetige Kontur (d.h. krümmungsstetig) zur Verfügung stellen. Aus diesem Grund ist die Befehlsauswahl im Bereich der statischen Werkzeugradiuskorrektur eingeschränkt.

Die Auflistung verfügbarer und nicht zulässiger Befehle für die Werkzeugradiuskorrektur ist im Unterkapitel Befehlsübersicht [▶ 13] zu finden.

Ablauf im NC-Kanal

Anhand der nachfolgenden Abbildungen wird der prinzipielle Ablauf erläutert.

Über das erstellte NC-Programm wird im Nicht-Echtzeitteil der CNC eine Parallelbahn bestimmt, für die eine Dynamikplanung durchgeführt wird.

Die Dynamikplanung mit dieser Parallelbahn ist für die Online TRC nur bei kleinen Radien (ca. $< 1 \text{ mm}$) und kleinen Schrägstellungen des Werkzeugs (ca. $< 15^\circ$) ausreichend genau. Die durch die Online TRC generierte abweichende Bahn wird dynamisch nicht berücksichtigt. Im Extremfall können dadurch Beschleunigungsüberschreitungen auftreten.

Im Echtzeitteil der CNC wird, ausgehend von dieser dynamisch geplanten Bahn, erneut auf die programmierte Bahn zurückgerechnet. Auf diese folgt dann die Online-Werkzeugradiuskorrektur, die in jedem Takt die Korrekturen berechnet.

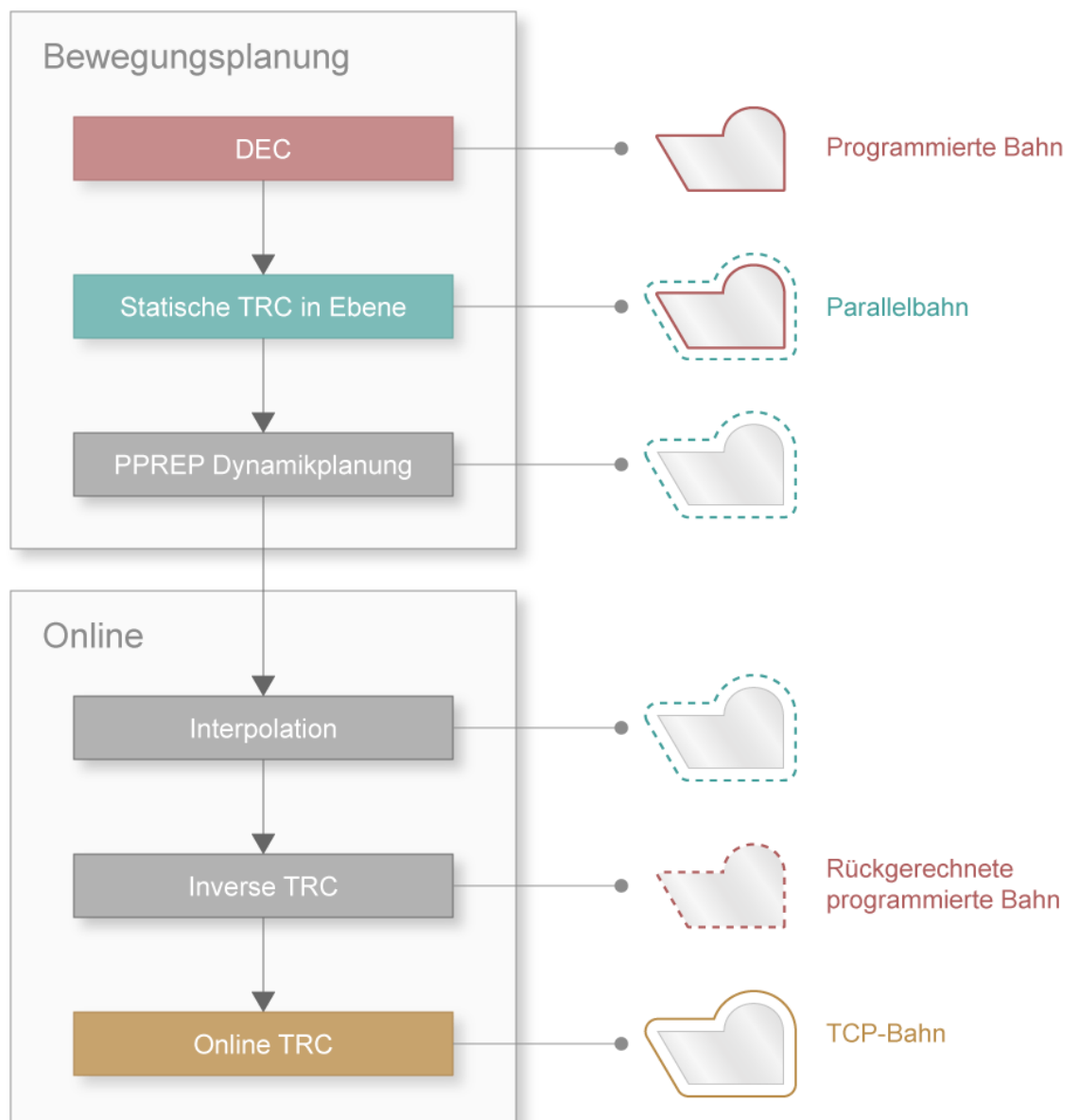


Abb. 2: Prinzipieller Ablauf im NC-Kanal

Mehrfad-Konfiguration und Werkzeugradiuskorrektur

Die Werkzeugradiuskorrektur erlaubt die Programmierung von 2 Pfaden in einer NC-Zeile, siehe PROG Kapitel: 2-Pfadprogrammierung. In jedem Pfad werden die Parallelbahn und die Dynamikplanung unabhängig vom jeweils anderen Pfad durchgeführt. Zum Zeitpunkt der Interpolation werden beide Pfade wieder synchronisiert ausgegeben.

Voraussetzung dafür ist der gesetzte Parameter P-CHAN-00550 [▶ 33]:

```
configuration.tool_radius_comp.function MULTI_PATH
```

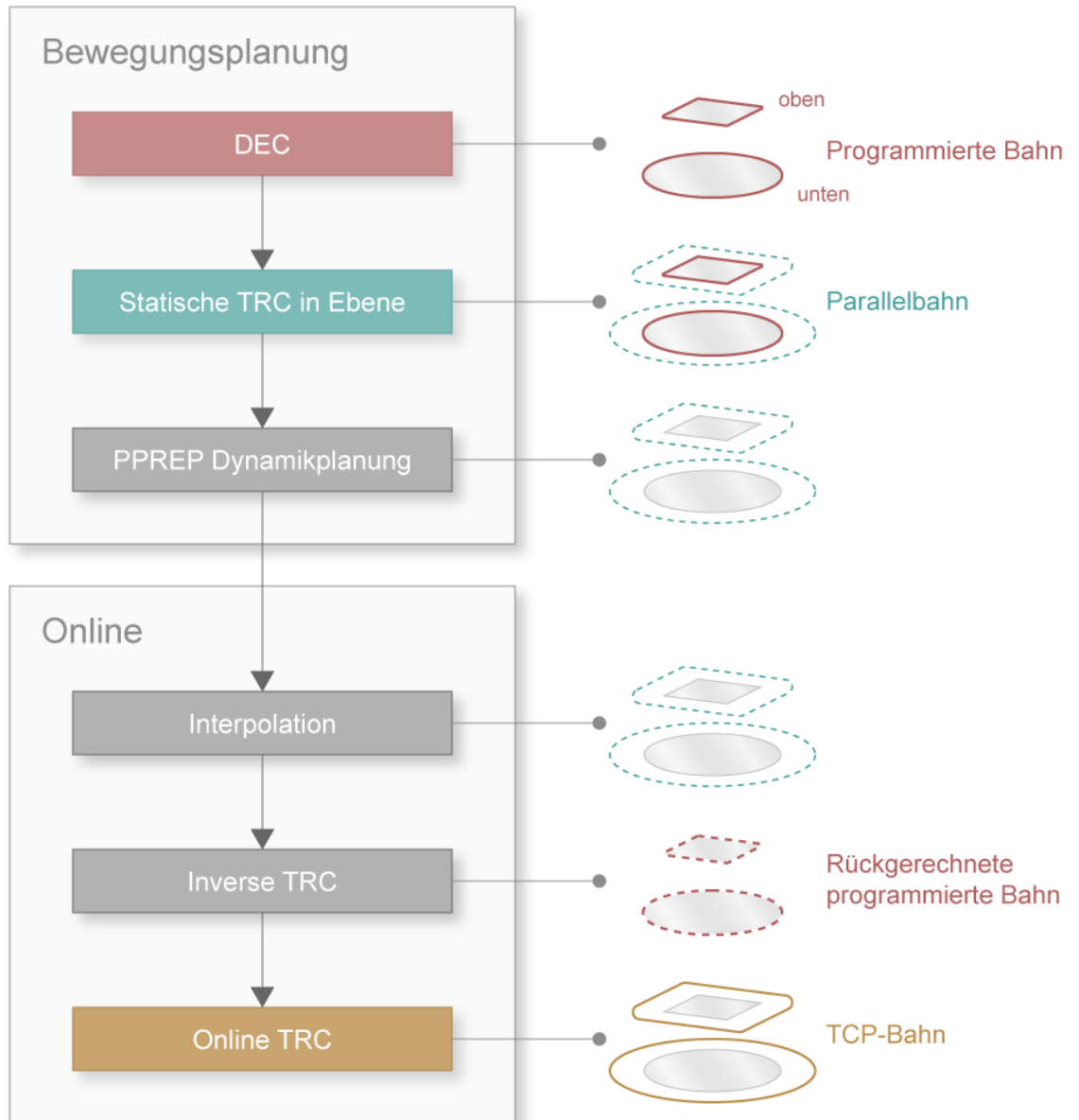


Abb. 3: Ablauf 2-Pfadprogrammierung mit Online TRC

Basierend auf der „Rückgerechneten programmierten Bahn“ kann der Anwender die eigene Online-Werkzeugradiuskorrektur einbringen.

Anwendungsbeispiel

Die Nutzung des TcCOM-Objekts in Verbindung mit einer 2-Pfad-Anwendung ist eine Einsatzmöglichkeit. Bei schräg stehendem Werkzeug (Draht) wird dabei in der Ebene aus dem Werkzeugradius eine Ellipse.

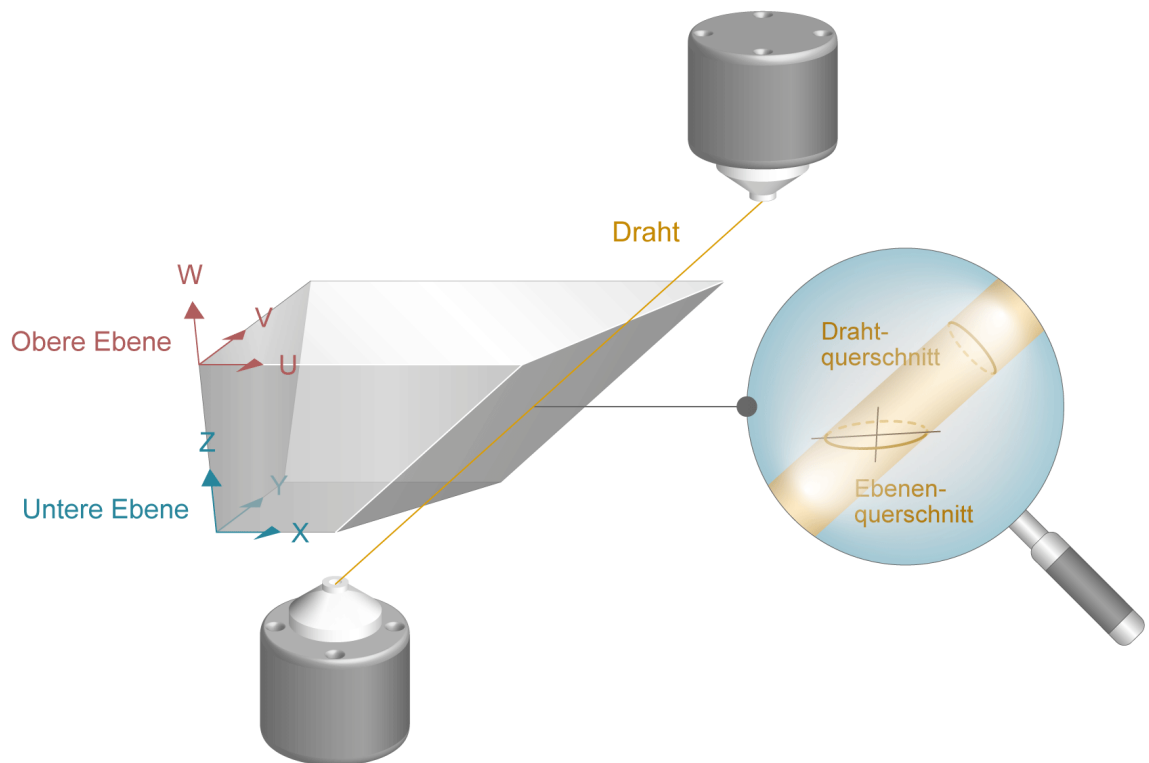


Abb. 4: 2-Pfad-Anwendung mit schräg stehendem Draht

2.2.1 Befehlsübersicht

Nachfolgend sind die G-Befehle, die die Werkzeugradiuskorrektur im NC-Programm betreffen, in zulässige und unzulässige Befehle aufgelistet:

Zulässige Befehle

NC-Befehl	Bedeutung
G26	Einfügen zirkularer Übergangssätze
G40	Abwahl Werkzeugradiuskorrektur
G41	Anwahl Werkzeugradiuskorrektur links der Kontur
G42	Anwahl Werkzeugradiuskorrektur rechts der Kontur
G138	Modus: Direkte Anwahl
G139	Modus: Indirekte Anwahl
G140	Abwahl der Konturausblendung
G141	Anwahl der Konturausblendung
G236	Modus: Direkte An-/Abwahl auf die Bahn
G237	Modus: Lotrechte An-/Abwahl
G239	Modus: Direkte An-/Abwahl ohne Satz

Unzulässige Befehle

NC-Befehl	Bedeutung
G05	Tangentiale An-/ Abwahl der Werkzeugradiuskorrektur
G25	Einfügen linearer Übergangssätze
G238	Modus: Inneneckanwahl der Werkzeugradiuskorrektur
#TRC [REMOVE_MASKED_BLOCKS]	Löschen von Konturschleifen

Bei Verwenden eines unzulässigen Befehls wird der Fehler mit ID 90166 ausgegeben.

2.2.2 Eigenschaften der Werkzeugradiuskorrektur

Dem McCOM-Objekt der Online-Werkzeugradiuskorrektur wird immer folgendes übergeben:

- Der aktuelle Werkzeugradius,
- die Position,
- der Bahntangentenvektor und der
- Werkzeugrichtungsvektor

Das Verhalten bei Satzübergängen und An- und Abwahl der statischen Werkzeugradiuskorrektur dient nur dem Verständnis.

Ausgangslage für die Online-Werkzeugradiuskorrektur ist die Berechnung der statischen Werkzeugradiuskorrektur.

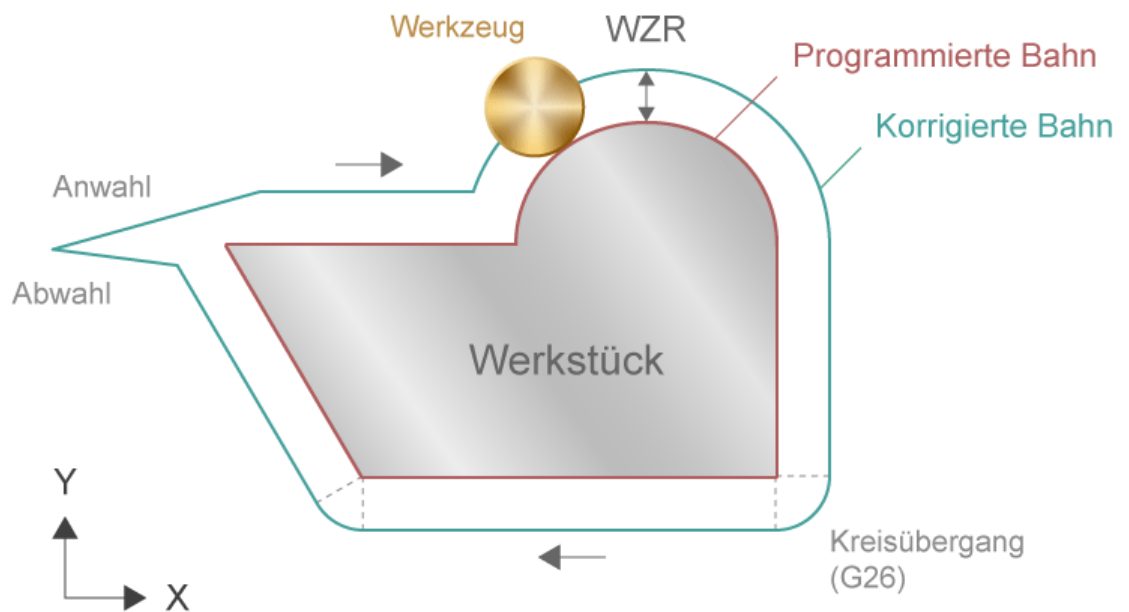


Abb. 5: Werkstück mit statischer Werkzeugradiuskorrektur

2.2.2.1 Verhalten an Satzübergängen mit Winkeln größer 180°

Bei G26 und Satzübergängen mit Übergangswinkeln größer als 180° fügt die statische Werkzeugradiuskorrektur Kreiselemente ein um die Kontur zu schließen. Mit dem eingefügten Kreiselement wird die Parallelbahn tangentialstetig.

Die Konturtangente ändert sich kontinuierlich in diesem Kreiselement.

Die Zeit für Änderung der Konturtangente ist näherungsweise der zurückgelegte Kreisfahrweg geteilt durch die erreichte Geschwindigkeit.

Während der Interpolation eines eingefügten Kreiselements wird die inverse TRC auf einen Punkt reduziert

Auf Basis der inversen TRC-Bahn kann die Online-Werkzeugradiuskorrektur die notwendige TRC-Korrektur in jedem Abtastzyklus berechnen.

Senkrecht stehendes Werkzeug

Bei senkrecht stehendem Werkzeug ist die resultierende Bahn der Online TRC identisch mit der geplanten 2D WRK Bahn.

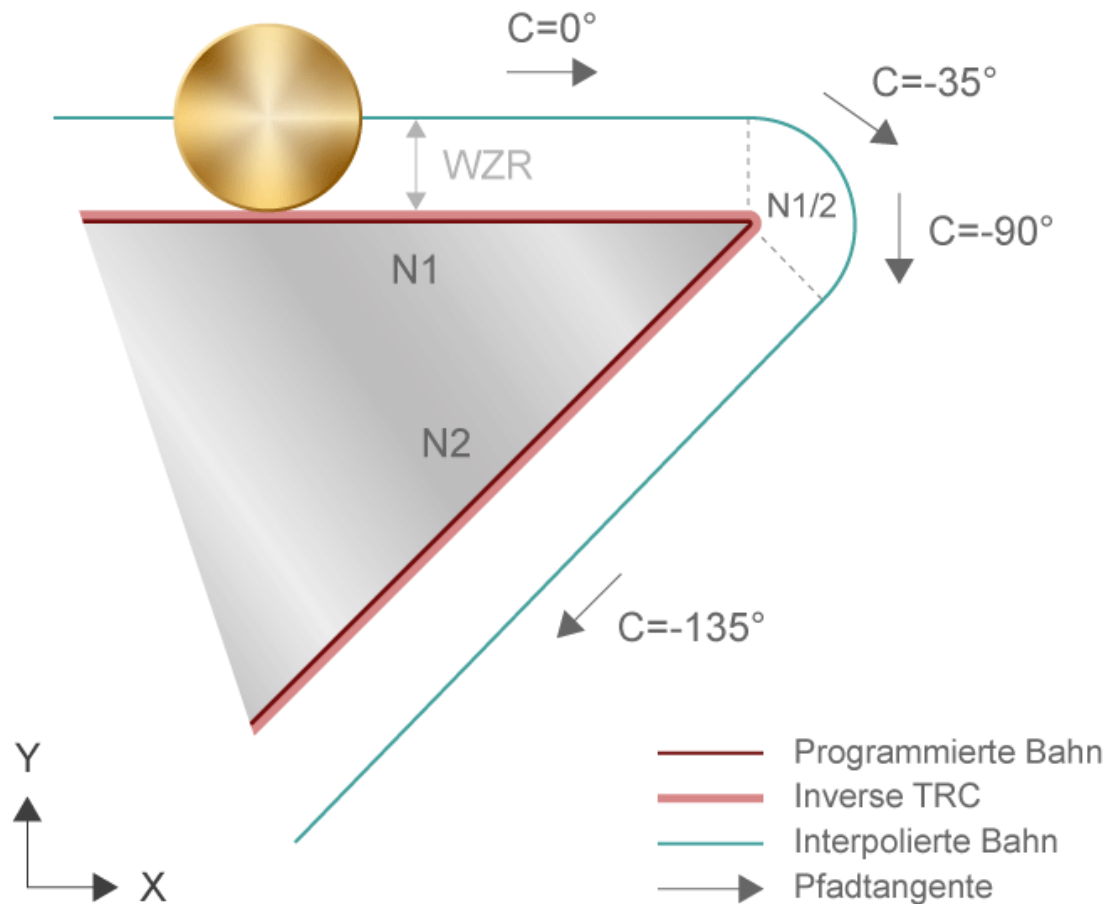


Abb. 6: Tangentendrehung an der Außenecke einer Geometrie

Verhalten bei schräg stehendem Werkzeug

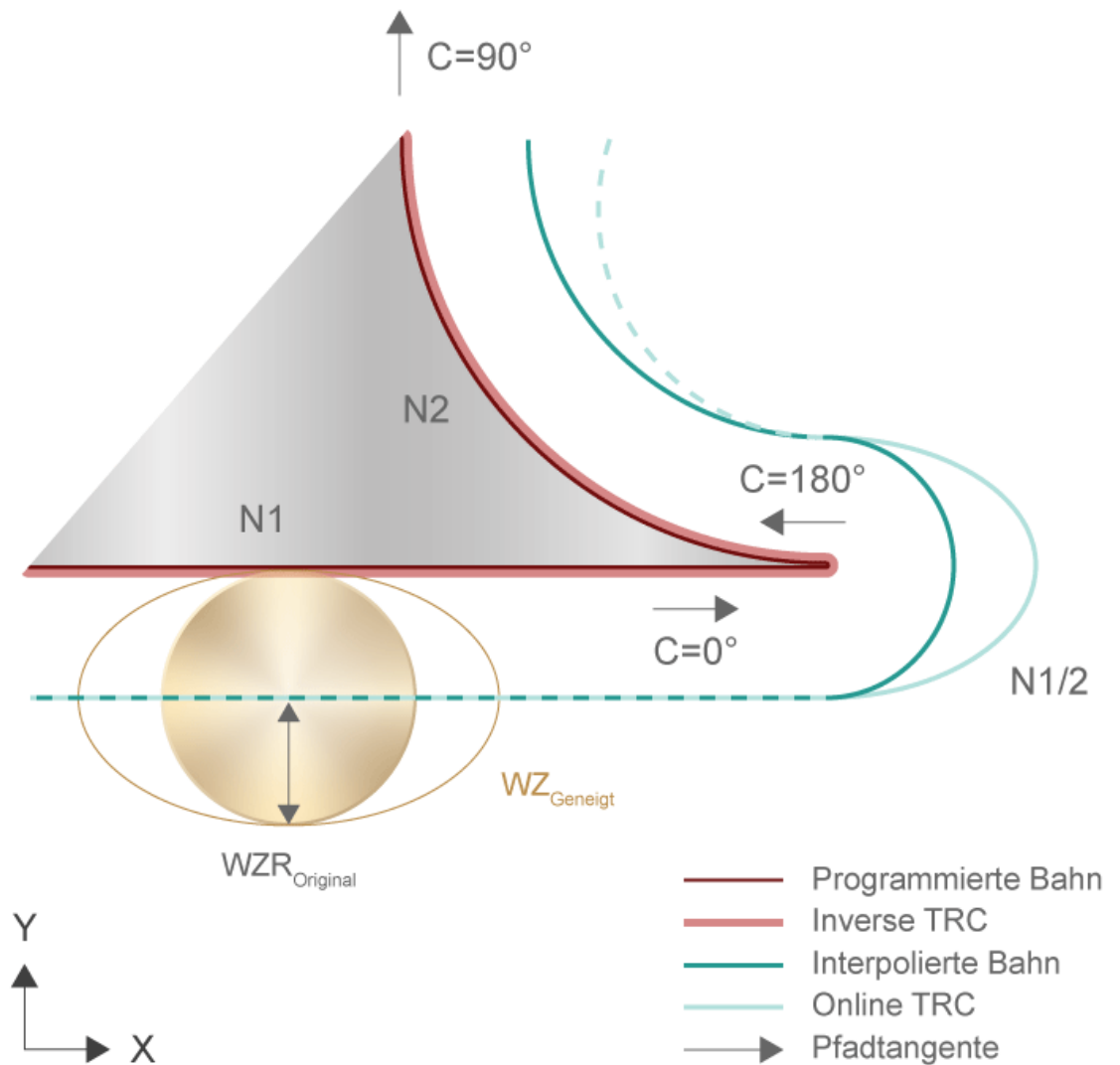


Abb. 7: Online TRC Bahn mit schräg stehendem Werkzeug an einer Außenecke

Weitere Satzübergänge sind unter [PROG, Kapitel: Generierung von Korrektursätzen] zu finden.

Änderung des Werkzeugradius

Wird der Werkzeugradius bei aktiver Werkzeugradiuskorrektur geändert, so kann dies über die TRC-Option „PERPENDICULAR_RADIUS_CHANGE“ [▶ 28] beeinflusst werden.

Im Standardfall, bei inaktiver Option, wird der geänderte Werkzeugradius im nachfolgenden Bewegungssatz ausgefahren (siehe Verhalten bei Änderung des Werkzeugradius).

Anschließend wird der Werkzeugradius mit gesetzter Option an der Außenecke geändert.

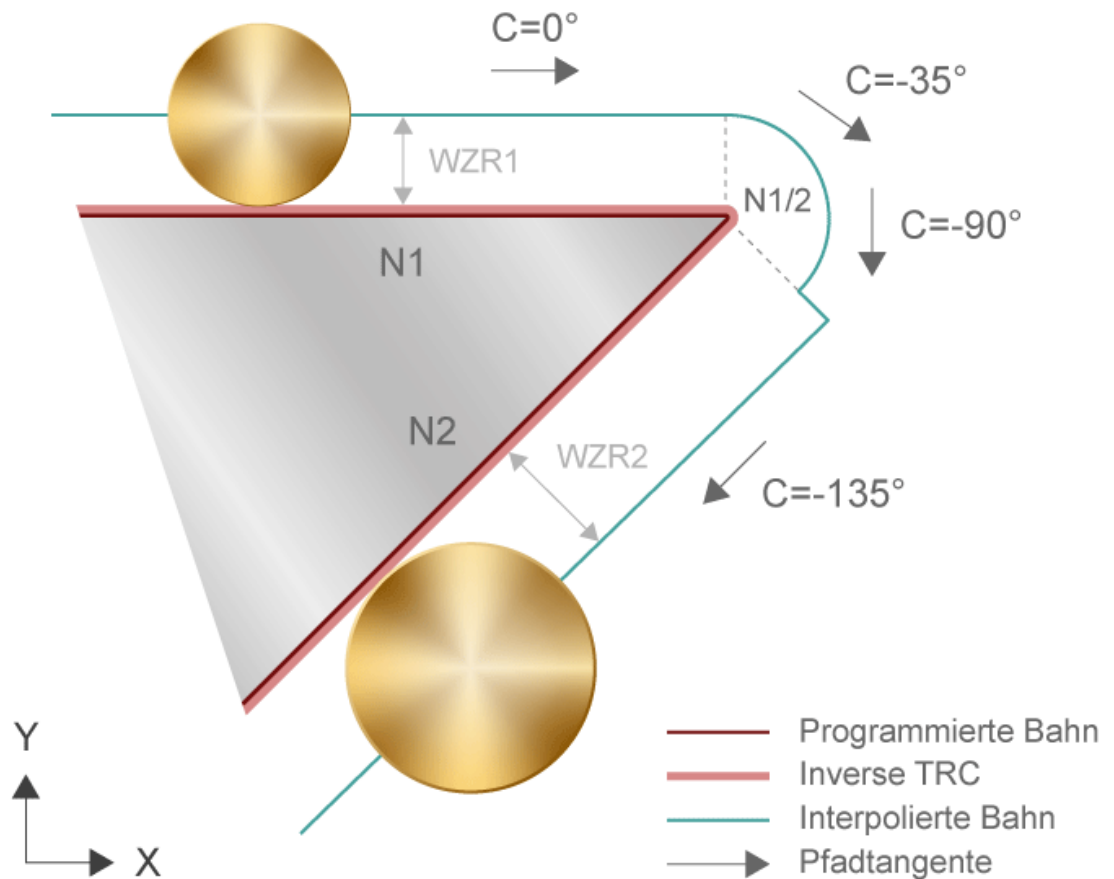


Abb. 8: Änderung des Werkzeugradius an einer Außenecke

2.2.2.2 Verhalten an Satzübergängen mit Winkeln kleiner 180°

Bei Satzübergängen mit Übergangswinkeln kleiner als 180° werden die beiden Satzelemente verkürzt.

Die inverse TRC verschiebt die beiden beteiligten Sätze zurück auf die programmierte Kontur. Der Verlauf der inversen TRC ist an dieser Stelle weder C0- noch C1-stetig. Während der Interpolation springt der Verlauf der inversen TRC-Bahn vom Satzende des ersten Satzes zum Satz-anfang des zweiten Satzes.

Auf Basis der inversen TRC-Bahn kann die Online-Werkzeugradiuskorrektur den realen TRC-Weg berechnen.

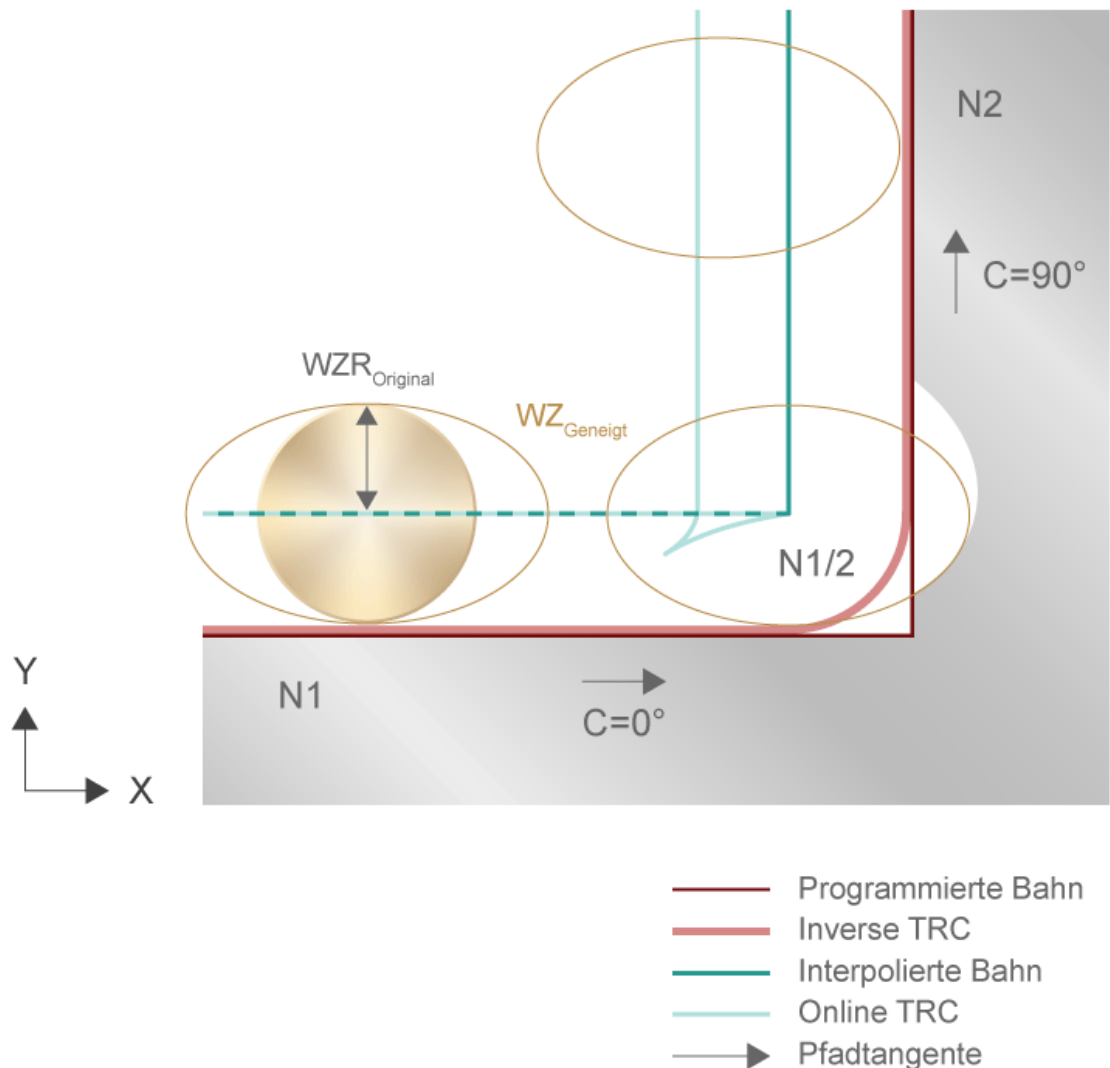


Abb. 9: Verhalten an der Innenecke einer Geometrie an Satzübergängen mit Winkel 90°

Weitere Satzübergänge sind unter [PROG, Kapitel: Generierung von Korrektursätzen] zu finden.

Änderung des Werkzeugradius

Wird der Werkzeugradius bei aktiver Werkzeugradiuskorrektur geändert, so kann dies über die TRC-Option „PERPENDICULAR_RADIUS_CHANGE“ [▶ 28] beeinflusst werden.

Im Standardfall, bei inaktiver Option, wird der geänderte Werkzeugradius im nachfolgenden Bewegungssatz ausgefahren (siehe Verhalten bei Änderung des Werkzeugradius).

Ist die Option gesetzt, so ergibt sich folgender Verlauf an einer Innenecke.

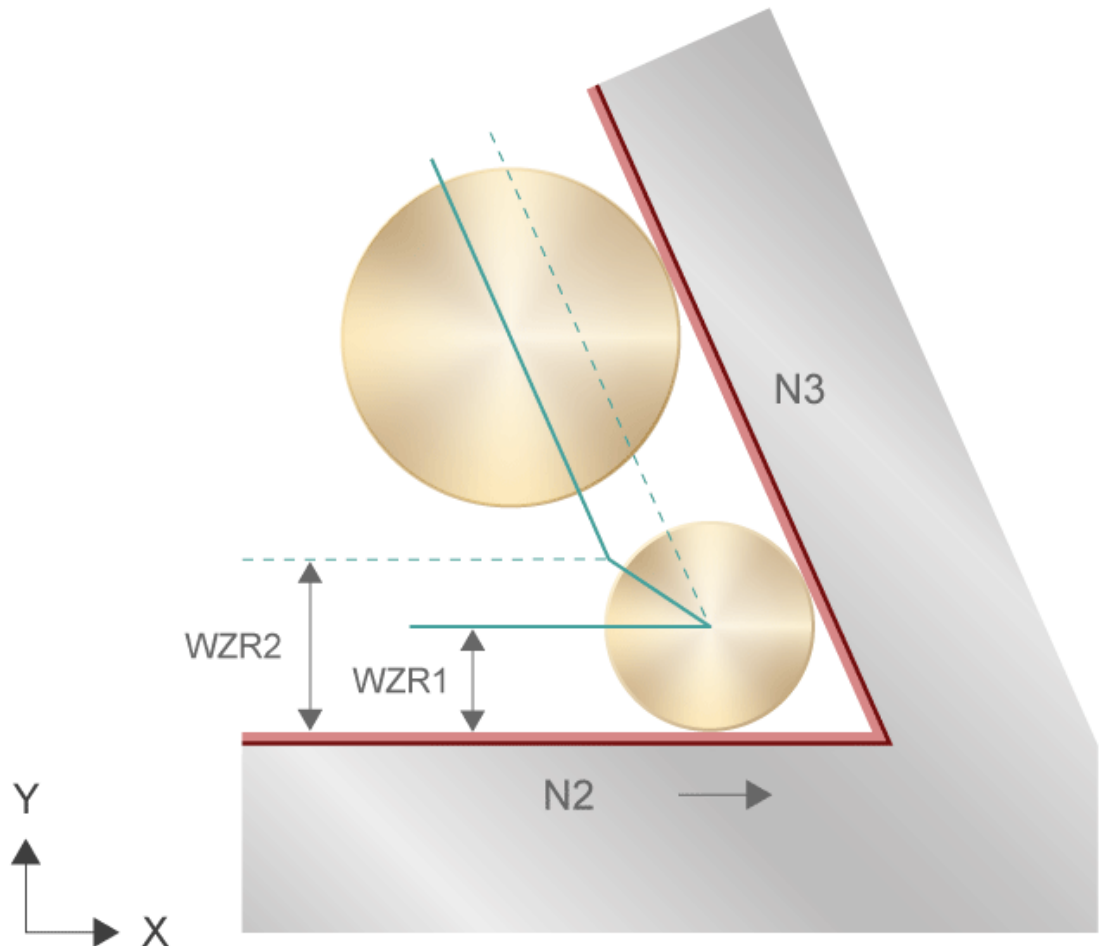


Abb. 10: Änderung des Werkzeugradius an der Innenecke einer Geometrie an Satzübergängen mit Winkel kleiner 180°

2.3 Anbindung einer Online-Werkzeugradiuskompensation via TcCOM

2.3.1 Methoden der Schnittstelle

Zu implementierende Methoden

Folgende Methode ist für eine Online-Werkzeugradiuskompensation zu implementieren (TcCnc-ToolRadiusCompInterfaces.h):

- virtual HRESULT TCOMAPI CalculateTrcPath (PTcToolRadiusCompParam trc)=0;

CalculateTrcPath	Berechnung der kompensierten Bahn anhand des gegebenen Werkzeugradius im Programmierkoordinatensystem. Input: Aktuellen Parameter der Werkzeugradiuskompensation.
-------------------------	---



Hinweis

Die Member-Variablen im Konstruktor der Klasse müssen für die Dimensionierung der Ein- und Ausgangskoordinaten initialisiert werden.

2.3.2 Instanzdaten

Arbeitsdaten – TcToolRadiusCompParam

Parameter der Methoden

Die Parameter für die einzelnen Methoden werden in gekapselter Form über die Struktur TcCncDynContourCtrlParam (TcCncDynContourCtrlInterfaces.h) übergeben:

```

struct TcToolRadiusCompParam
{
    EcToolRadiusComp    type;

    int dimension; // dimension of positions, 3 or 6 for 2 pathes
    int dim_radius; // dimension of radius, 1 or 2 for 2 pathes

    const double * i_path; // input: position of path, see dimension
    const double * i_tangent; // input: tangent of path, see dimension
    const double * i_tool_direction; // input: direction of tool, see dimension
    const double * i_radius; // input: actual tool radius, see dim_radius

    double * o_path; // output: position of compensated path
}
  
```

Struktur TcCncParam

Die folgenden Parameter werden im Allgemeinen in jeder TcCOM-Schnittstelle verwendet:

```

struct TcCncParam
{
    // Input data
    EcCncCallerID caller_id; // identification of caller
    unsigned long block_number; // block number of actual executed
                                // CNC program, MDI(not yet supported)
    unsigned short cnc_channel; // calling CNC channel
    // Output data
    double ret_value1; // additional error value
    double ret_value2; // additional error value
    char ret_text[24]; // out: additional error text, max. 24
byte
}
  
```

Identifikation der Aufrufer (Caller-ID)

Die online Werkzeugradiuskompensation wird an unterschiedlichen Stellen im Interpolationsmodul aufgerufen. Zur Identifikation sind die folgenden Kennungen verfügbar:

Identifikationsnummer	Caller-ID
1000	EcCncTrcCallerID_Display
1001	EcCncTrcCallerID_Interpolation
1002	EcCncTrcCallerID_GeometricFeedAdaption
1003	EcCncTrcCallerID_TargetPos

Das TcCOM-Objekt kann die Caller-IDs verwenden, um unterschiedliche Algorithmen für die Interpolation und für die Anzeigefunktionalität zu implementieren, z.B.:

- Bei der Interpolation wird in beiden Pfaden der individuelle Radius verwendet.
- Für die Anzeige in beiden Pfaden wird der gleiche (z.B. sekundäre) Radius verwendet.

2.3.3 Konfigurieren und Anmelden des TcCOM-Objekts

Registrieren in TwinCAT

Zum Anmelden eines TcCOM-Objekts der Online-Werkzeugradiuskompensation werden die folgenden Daten verwendet (TcCncServices.h):

- Type 3 (CCNC_REGISTEROBJECT_TYPE_TOOL_RADIUS_COMP) fest vorgegeben.
- Verwendet das TcCOM-Objekt keine instanzspezifischen Variablen, so wird der Eintrag Group auf 0 gesetzt .
- Verwendet das TcCOM-Objekt instanzspezifischen Variablen, so wird der Eintrag Group auf die jeweilige Kanannummer [1;12] gesetzt, zu dem das Objekt zugeordnet ist. Maximal ein Objekt pro Kanal.
- Index wird nicht verwendet.

Die Anmeldung des Objekts der online Werkzeugradiuskompensation erfolgt über nachfolgendes TcCOM-Interface, welches in der Datei TcCncInterfaces.h definiert ist:

- virtual HRESULT TCOMAPI RegisterObject(TcCncRegisterObject& id, ITcUnknown* ipUnk)=0;
- virtual HRESULT TCOMAPI UnregisterObject(TcCncRegisterObject& id)=0;

Bereitstellung des TcCOM-Objekts

Nach der Generierung der Online-Werkzeugradiuskompensation müssen 2 Dateien vorhanden sein:

1. TMC-Datei
2. Treiberdatei

Die Beschreibung der Online-Werkzeugradiuskompensation ist in der TMC-Datei enthalten, z.B. TcCncMyOnlineToolRadiusComp.tmc.

Diese ist im Arbeitsverzeichnis der Solution.

Das Verzeichnis der Treiberdatei ist abhängig von Release oder Debug:

- <TwinCAT> \3.1\sdk_products\TwinCAT RT (x64)\Release bzw.
- <TwinCAT> \3.1\sdk_products\TwinCAT RT (x64)\Debug

Mit der Aktivierung der Konfiguration wird die jeweilige Treiberdatei automatisch in das Verzeichnis <TwinCAT>\3.1\Driver\AutoInstall kopiert.

Anhand des obigen Beispielnamens: TcCncMyOnlineToolRadiusComp.sys



Hinweis

Der Anwender muss nur die Generierung (Debug/Release) auslösen und die entsprechende Konfiguration aktivieren.

Die Vorgehensweise zum Debuggen der erstellten Online-Werkzeugradiuskompensation ist analog zum Debuggen einer McCOM-Transformation. Diese Vorgehensweise wird in [McCOM-Trafo, Kapitel: Transformation debuggen] geschildert.

Laden des Objekts

Das Laden des Objekts für die Online-Werkzeugradiuskompensation ist in „Objekt der Online-Werkzeugradiuskorrektur einbinden [71]“ veranschaulicht.

2.4 Programmierung

Die Nutzung der Online-Werkzeugradiuskompensation (auch Online-TRC) wird über die Programmierung der Optionen INVERSE und ONLINE [▶ 24] festgelegt. Wirksam wird die gewählte Einstellung mit der Aktivierung der Werkzeugradiuskorrektur (WRK) mit G41 oder G42. Die Abwahl der Online-TRC erfolgt mit der Abwahl der WRK mit G40.



Achtung

Für die Nutzung der Online-TRC müssen beide Optionen #TRC [INVERSE=1 ONLINE=1] gesetzt sein.

Es wird sonst die Fehlermeldung mit der ID 22125 ausgegeben.



Programmierbeispiel

WRK-Anwahl mit gesetzten Optionen

```

N10 V.G.WZ_AKT.R = 0.3
N20 G26
N30 G138
; ...
N70 #TRC [INVERSE=1 ONLINE=1] ( Setzen der Optionen)

N100 G0 X0 Y0 U0 V0

N280 G41      ( Anwahl WRK)
( Kontur)
; ...
N500 G40      ( Abwahl WRK)
; ...
N550 M30
  
```

Programmierung des Werkzeugradius

Gleicher Werkzeugradius in beiden Pfaden

```
N.. V.G.WZR=0.15
```

alternativ

```
N.. V.G.WZR=0.15 : X100 Y20: U100 V20
```

Unterschiedliche Werkzeugradien in beiden Pfaden

```
N.. : V.G.WZR=0.139 : V.G.WZR=0.15
```

alternativ

```
N.. : V.G.WZR=0.139 X100 Y20 : V.G.WZR=0.15 U100 V20
```

Festlegen des Werkzeugradius im Referenzpfad

```
N.. : V.G.WZR=0.134 :
```

Festlegen des Werkzeugradius im zweiten Pfad

```
N.. : : V.G.WZR=0.151
```

2.4.1 TRC Optionen für Online-TRC und 2-Pfad

Online-TRC

Mit den nachfolgenden Parametern kann Einfluss auf die Online-Werkzeugradiuskorrektur genommen werden. Diese kann nur gesetzt werden mit aktiver Werkzeugradiuskorrektur (G41/G42).

Mit dem Parameter INVERSE wird die durch die Werkzeugradiuskorrektur entstandene verschobene Parallelbahn auf die ursprünglich programmierte Bahn zurückgerechnet.

Der Parameter ONLINE ermöglicht bei gesetztem Parameter INVERSE Einstellungen für die Online-Werkzeugradiuskompensation. Diese Online-Werkzeugradiuskompensation kann über ein TcCOM-Objekt realisiert werden.



Achtung

Der Parameter ONLINE darf nur auf einen Wert ungleich 0 gesetzt werden, wenn der Parameter INVERSE aktiv ist. Bei inaktivem INVERSE wird der Fehler ID 22125 ausgegeben.

Syntax:

#TRC [[INVERSE=..] [ONLINE=..] [ONLINE_BY_VECTOR=..]

INVERSE=..	Rückrechnen der durch die Werkzeugradiuskorrektur entstandenen Parallelbahn. 0: kein Rückrechnen (Standard). 1: Rückrechnen der Parallelbahn.
ONLINE=..	Einstellung der Online-Werkzeugradiuskompensation: 0: Keine Berechnung der Online-Werkzeugradiuskompensation. 1: Berechnung der kompensierten Bahn via TcCOM-Schnittstelle. 2: Einfache Berechnung der Parallelbahn für jede Ebene in der CNC (entspricht der Bahn aus der Werkzeugradiuskorrektur, nur zum Test) 3: Berechnung der Parallelbahn unter Berücksichtigung der Werkzeugorientierung in der CNC.
ONLINE_BY_VECTOR=..	Dieser Parameter bestimmt die Methode, mit der eine 3D-TRC online korrigierte Position auf die Koordinatensystemebene zurücktransformiert wird. 0: Die Verschiebung entlang der Drehrichtung muss in einem TcCOM Objekt der kinematischen Transformation implementiert werden. 1: Die Verschiebung entlang der Drehrichtung wird direkt nach der 3D-TRC-Korrektur durch die CNC ausgeführt. 2: Kombinationslösung; die zyklische Sollwerterzeugung verwendet die Implementierung im TcCOM Objekt der kinematischen Transformation, die Anzeige nutzt die Berechnung durch die CNC direkt nach der 3D-TRC-Korrektur.

Basisradius für Werkzeugradiuskorrektur

Mit dieser Option kann festgelegt werden, ob die Werkzeugradiuskorrektur für die einzelnen Pfade mit dem individuell für diesen Pfad festgelegten Werkzeugradius die Parallelbahn berechnen soll. Alternativ mit dem Werkzeugradius des Pfades, der über diese Option bestimmt werden kann.

Diese Option kann beim Drahterodieren dazu verwendet werden auf einem HMI die ideale Bahn anzuzeigen, während der Drahtverschleiß bei der Berechnung der Sollgeometrie berücksichtigt wird.

Syntax:

#TRC [[MULTI_PATH_RADIUS=..]]

MULTI_PATH_RADIUS=.. Mit diesem Parameter kann festgelegt werden, mit welchem Werkzeugradius die statische Werkzeugradiuskorrektur die Parallelbahn berechnen soll.

INDIVIDUAL/ Die Werkzeugradien, die in dem jeweiligen Pfad programmiert
 DEFAULT sind, werden verwendet (Standard).

REFERENCE Der Werkzeugradius des Referenzpfades wird verwendet.

SECONDARY Der Werkzeugradius des zweiten Pfades wird verwendet.

Dynamische Offsetüberlagerung

Die dynamische Offsetüberlagerung ist eine anwenderspezifische Überlagerung der TCP Bahn, die in der Online-TRC wirkt.

Die überlagerten Achsbewegungen können deshalb nur eingeschränkt bei der Planung der maximalen Geschwindigkeit und Beschleunigung in der CNC berücksichtigt werden.

Speziell an Innenecken können die Überlagerungen zu Beschleunigungsüberschreitungen führen. Um dies zu verhindern, analysiert die CNC die Tangentendrehung an Innenecken und bestimmt daraus die maximal zulässige Dynamik.

Die An/Abwahl der Funktionalität in der Online-TRC wird über die SPS geschaltet. Die Berechnung der zulässigen Dynamik bei der Offsetüberlagerung an Innenecken wird mit nachfolgendem Programmierbefehl aktiviert:

Syntax:

#TRC [[DYNAMIC_VARIATION_MAX_OFFSET=..]]

DYNAMIC_VARIATION_MAX_OFFSET =.. Definition des maximalen dynamischen Offsets in [0,1µm]

0: Abwahl Dynamische Offsetüberlagerung (Standard).

> 0: Anwahl, maximaler Offset wird für Dynamikberechnung verwendet.

TAPERLINK

Die Option TAPERLINK ermöglicht bei einer 2-Pfadkonfiguration die Synchronisation zwischen dem Referenzpfad und dem Sekundärpfad, um die optimale Drahtneigung zu erhalten. Siehe [FCT-C49, Kapitel: Beschreibung].

Voraussetzung für die Funktionalität ist eine 2-Pfadkonfiguration und die Anwahl der Werkzeugradiuskorrektur mit G41 oder G42.

Syntax:

#TRC [[TAPERLINK=..]]

TAPERLINK =.. Modus für Funktionalität Taper-Link festlegen.

0: Taper-Link Funktionalität inaktiv (Standard).

1: Taper-Link aktiv: Kompensation ist auf beiden Pfaden aktiv, automatische Erkennung.

2: Taper-Link aktiv: Referenzpfad kompensiert den 2. Pfad.

3: Taper-Link aktiv: 2. Pfad kompensiert den Referenzpfad.

2.4.2 TRC Option GEN_CIR_BLOCK_IN_CORNER

Mit dieser Option kann bei aktiver Werkzeugradiuskorrektur in sogenannten Innenecken ein virtueller Kreis (Radius 0) integriert werden. Innenecken sind Übergänge zwischen Bewegungssätzen, deren Übergangswinkel kleiner als 180° ist.

Nur möglich mit einem Werkzeugradius ungleich 0.

Diese Option muss zur Vermeidung von Beschleunigungsüberschreitungen verwendet werden.

Syntax:

#TRC [[GEN_CIR_BLOCK_IN_CORNER=..]]

GEN_CIR_BLOCK_IN_CORNER=.. Einfügen von virtuellen Kreissätzen mit Radius 0 bei Innenecken der Werkzeugradiuskorrektur.

0: Kein Einfügen von virtuellen Kreissätzen (Standard)

1: Einfügen von virtuellen Kreissätzen

2.4.3 TRC Option G236_LIN

Diese Option ist nur wirksam bei Verwendung des Anwahlmodus G236 und einem Übergangswinkel im An- oder Abwahlbereich der Werkzeugradiuskorrektur zwischen 90° und 180° .

Syntax:

#TRC [[G236_LIN =..]]

G236_LIN=.. Festlegung, ob im beschriebenen Winkelbereich 90° bis 180° ein Kreis- oder Linearsatz eingefügt wird.

0: Einfügen eines Kreissatzes (Standard)

1: Einfügen eines Linearsatz

2.4.4 TRC Option PERPENDICULAR_RADIUS_CHANGE

Mit dieser Option wird eine programmierte Änderung des Werkzeugradius unmittelbar durch Einfügen einer Bewegung orthogonal zur programmierten Kontur ausgefahren.

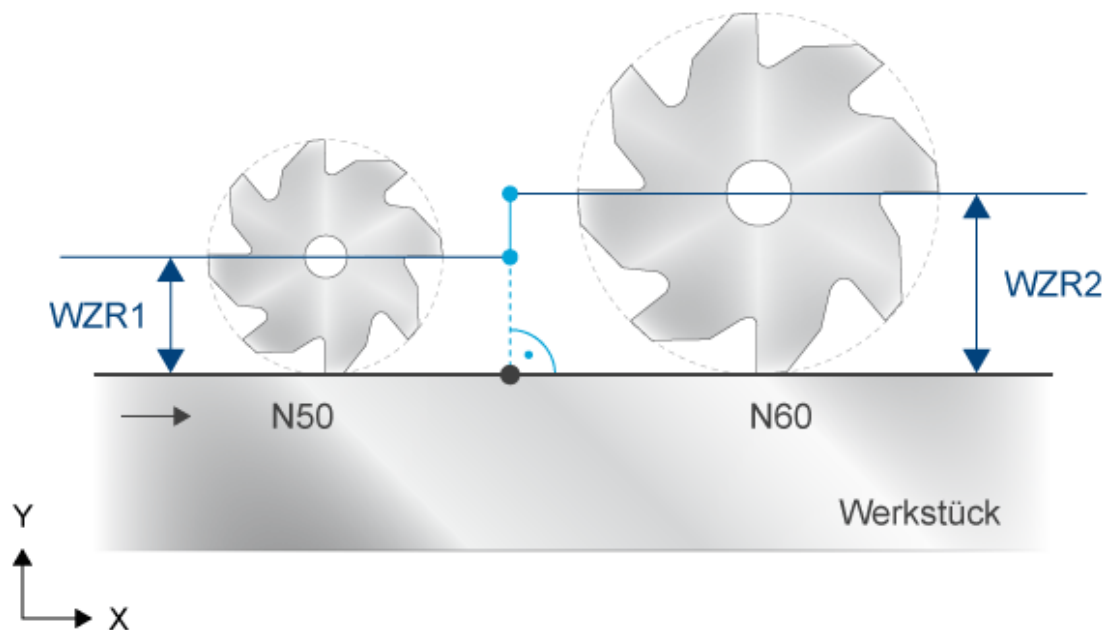


Abb. 11: Orthogonales Ausfahren der Änderung des Werkzeugradius

Syntax:

#TRC [[PERPENDICULAR_RADIUS_CHANGE=..]]

PERPENDICULAR_RADIUS_CHANGE=.. Mit diesem Parameter kann die Änderung des Werkzeugradius senkrecht zur Kontur ausgefahren werden.

0: Kein senkrecht ausgefahren der Änderung des neuen Werkzeugradius (Standard)

1: Senkrecht ausgefahren der Änderung des neuen Werkzeugradius

In den Programmbeispielen soll zwischen den programmierten Bewegungssätzen N50 und N60 der Werkzeugradius geändert werden.

Um das Verhalten besser zu veranschaulichen, ist die Änderung des Werkzeugradius sehr groß gewählt. Üblicherweise handelt es sich bei der Änderung des Werkzeugradius um sehr kleine Korrekturen zum Beispiel durch Verschleiß.

In den nachfolgenden Beispielen werden einige Satzübergänge exemplarisch dargestellt. Es sind alle Kombinationen aus Linear- und Zirkularsätzen zulässig.

Änderung des Werkzeugradius im Inneneck

	<pre>%Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 5 N20 G42 N40 G01 X20 Y0 N50 G01 X50 Y50 N55 V.G.WZ_AKT.R = 6 N60 G01 X100 N70 G40 X120 Y0 N99 M30</pre>
	<pre>%Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 6 N20 G42 N40 G01 X20 Y0 N50 G01 X50 Y50 N55 V.G.WZ_AKT.R = 5 N60 G01 X100 N70 G40 X120 Y0 N99 M30</pre>
	<pre>%Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 5 N20 G42 N30 G01 X10 Y20 N40 G01 X20 N50 G03 X50 Y25 R30 N55 V.G.WZ_AKT.R = 6 N60 G03 X90 Y18 R40 N70 G01 X100 N80 G40 X120 Y0 N99 M30</pre>

	<pre> %Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 6 N20 G42 N30 G01 X10 Y20 N40 G01 X20 N50 G03 X50 Y25 R30 N55 V.G.WZ_AKT.R = 5 N60 G03 X90 Y18 R40 N70 G01 X100 N80 G40 X120 Y0 N99 M30 </pre>
--	--

Änderung des Werkzeugradius am Außeneck

	<pre> %Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 6 N20 G42 G25 N40 G01 X20 Y20 N50 G01 X50 N55 V.G.WZ_AKT.R = 5 N60 G01 X80 Y50 N70 G01 X120 N80 G40 X140 Y0 N99 M30 </pre>
--	---

	<pre> %Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 6 N20 G42 G26 N40 G01 X20 Y20 N50 G01 X50 N55 V.G.WZ_AKT.R = 5 N60 G01 X80 Y50 N70 G01 X120 N80 G40 X140 Y0 N99 M30 </pre>
--	---

	<pre>%Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 5 N20 G42 G25 N40 G01 X20 Y20 N50 G01 X50 N55 V.G.WZ_AKT.R = 6 N60 G01 X80 Y50 N70 G01 X120 N80 G40 X140 Y0 N99 M30</pre>
	<pre>%Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 5 N20 G25 N30 G42 X10 Y10 N40 G01 X20 N50 G02 X60 Y0 R30 N55 V.G.WZ_AKT.R = 6 N60 G02 X90 Y12 R50 N70 G01 X100 N80 G40 X120 Y0 N99 M30</pre>
	<pre>%Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 5 N20 G26 N30 G42 X10 Y10 N40 G01 X20 N50 G02 X60 Y0 R30 N55 V.G.WZ_AKT.R = 6 N60 G02 X90 Y12 R50 N70 G01 X100 N80 G40 X120 Y0 N99 M30</pre>

	<pre> %Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 5 N20 G25 N30 G42 X10 Y10 N40 G01 X20 N50 G02 X60 Y0 R30 N55 V.G.WZ_AKT.R = 6 N60 G01X80 Y40 N70 G01 X100 N80 G40 X120 Y0 N99 M30 </pre>
	<pre> %Test.nc N05 G0 X0 Y0 F1000 #TRC [PERPENDICULAR_RADIUS_CHANGE=1] N10 V.G.WZ_AKT.R= 6 N20 G26 N30 G42 X10 Y10 N40 G01 X20 N50 G02 X60 Y0 R30 N55 V.G.WZ_AKT.R = 6 N60 G01X80 Y40 N70 G01 X100 N80 G40 X120 Y0 N99 M30 </pre>

2.5 Parameter

P-CHAN-00550	Festlegung der Funktionalitäten für die Werkzeugradiuskorrektur
Beschreibung	Der Parameter legt einzelne Funktionalitäten für die Werkzeugradiuskorrektur fest.
Parameter	configuration.tool_radius_comp.function
Datentyp	STRING
Datenbereich	MULTI_PATH: 2-Pfadkonfiguration und 2-Pfadprogrammierung aktiv -: Keine Funktionalitäten festgelegt.
Dimension	----
Standardwert	*
Anmerkungen	<p>Parameter ist ab folgenden Versionen verfügbar V2.11.2040.04 ; V2.11.2810.02 ; V3.1.3079.17 ; V3.1.3107.10</p> <p>* Hinweis: Der Standardwert der Variablen ist ein Leerstring.</p> <p>Über P-CHAN-00555 und P-CHAN-00556 besteht die Möglichkeit abhängig vom Bearbeitungsmodus Funktionen festzulegen.</p>

2.6 Fehlermeldungen

Anwenderspezifische Kompensationsfehler

Neben den standardmäßigen Transformationsfehler kann der Anwender bei einigen Methoden (Bsp. Fehler ID 123) individuelle Fehler über den Rückgabewert der Funktion absetzen (0 = OK).

```
HRESULT CToolRadiusComp::CalculationTrcPath(PTcCncToolRadiusCompParam trc){
    if (...)
        return 123; // raise error
    ...
    return S_OK;
}
```

Fehlertexte in TcCncUsersEvents.xml

Im Fehlerfall wird der individuelle Rückgabewert der Methode an die Fehlermeldungsauswertung über die SPS oder den TwinCAT-Eventlogger weitergereicht, siehe auch FCT-M7, Kapitel: Fehlerausgabe TwinCAT3.

Die Fehlertexte müssen in der XML-Fehlertextdatei in der entsprechenden Sprache ergänzt werden. (C:\TwinCAT\3.1\Target\Resource):

```
<Event>
    <Id>123</Id>
    <Message Lcid="1033">Online tool radius compensation: error 123</Message>
    <Message Lcid="1031">Online-Werkzeugradiuskorrektur: Fehler 123</Message>
</Event>
```

Die Ausgabe des Fehlers erfolgt über den Eventlogger.

3 Geometrische Vorschubanpassung

3.1 Übersicht

Aufgabe

Die Funktionalität „Geometrische Vorschubanpassung“ bietet beim Drahterodieren die Möglichkeit, über einen zyklisch auf den Sollvorschub wirkenden Faktor einen konstanten Flächenabtrag zu gewährleisten.



Versionshinweis

Diese Funktionalität ist ab CNC-Version V3.1.3108 verfügbar.



Hinweis

Diese Funktionalität ist Bestandteil einer lizenzpflichtigen Zusatzoption.

Eigenschaften

Die Funktionalität kann durch ein anwenderspezifisches TcCOM-Objekt verwendet werden.

Mit der geometrischen Vorschubanpassung wird ein Faktor ermittelt, der mit der aktuellen Sollgeschwindigkeit und dem Overridefaktor multipliziert wird.

Programmierung/Parametrierung

Die Funktionalität kann über den NC-Befehl #GEO FEED ADAPT [► 40] und über die entsprechenden Kanalparameter [► 35] konfiguriert, aktiviert und deaktiviert werden.

Verweise 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.

3.2 Beschreibung

Mit der geometrischen Vorschubanpassung kann bei der Drahterosion ein Faktor berechnet werden um einen konstanten Flächenabtrag zu gewährleisten.

Die Berechnungen erfolgen auf Basis einer zyklisch korrigierten Radiusmittelpunktbahn. Der Werkzeugradius setzt sich dabei aus dem effektiven Werkzeugradius und einem zusätzlichen Parameter, dem Spaltwert, zusammen.

Die korrigierten Bahnpositionen werden dem TcCom-Objekt zyklisch bereitgestellt.

Die zusätzlich benötigten Parameter können über die SPS übergeben werden. Eine Definition der Parameter im NC-Programm ist ebenfalls möglich.

Das Ergebnis der Berechnungen ist ein Faktor, der multiplikativ in den kommandierten Vorschub eingeht.

Nachfolgend ist die Anordnung der Vorschubanpassung im NC-Kanal dargestellt:

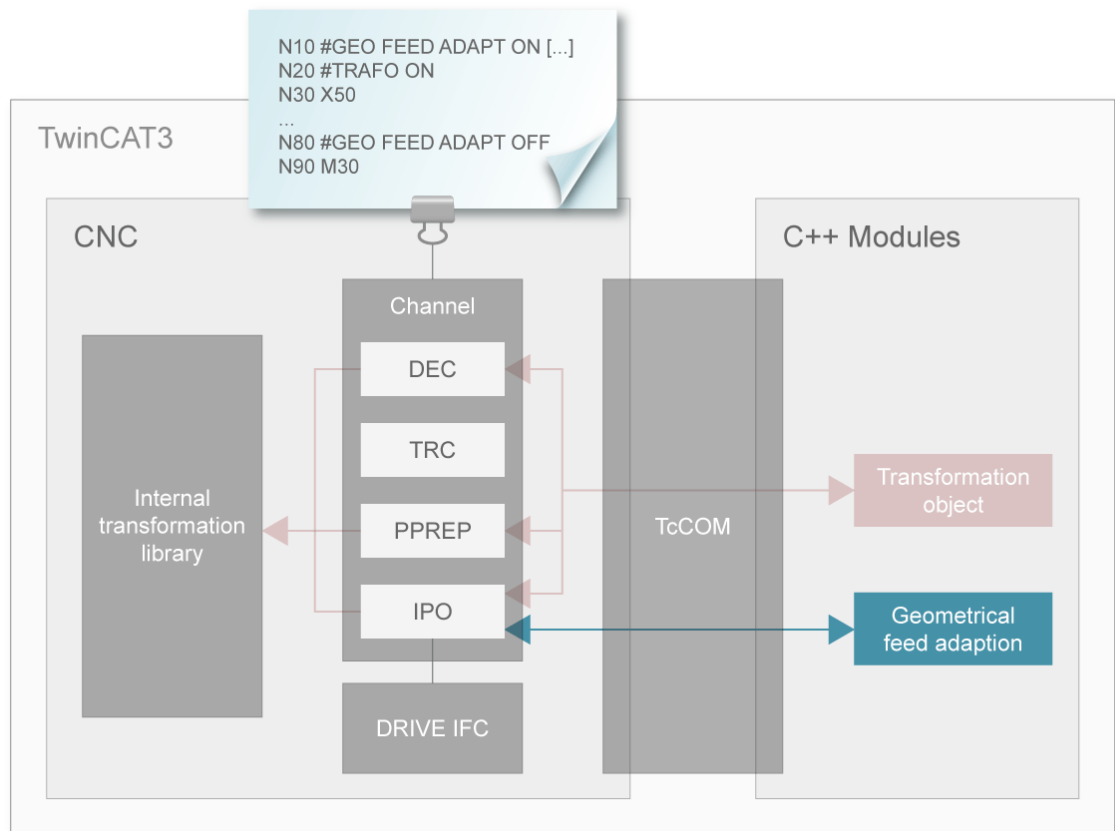


Abb. 12: Geometrische Vorschubanpassung im NC-Kanal



Hinweis

Bei einem TcCOM-Objekt mit instanzspezifischen Variablen muss das Objekt im jeweiligen CNC-Kanal konfiguriert werden.

3.2.1 Integration in den NC-Kanal

Die geometrische Vorschubanpassung besteht aus mehreren Elementen:

1. Aufruf der Online TRC mit Caller ID "geometric feed adaption". Der Spaltwert aus dem NC-Programm (Vorbelegung in der Kanalliste) wird hierbei noch zum Drahradius addiert.
2. Transformieren der Positionen von PCS- in MCS-Koordinaten. Die Positionen der zweiten Ebene werden relativ zum Referenzkoordinatensystem berechnet.
3. Aufruf des TcCOM-Objekts „Geometrische Vorschubanpassung“. Berechnung des zusätzlichen Overridefaktors. Dieser wird noch mit dem Overridewert von der SPS multipliziert.

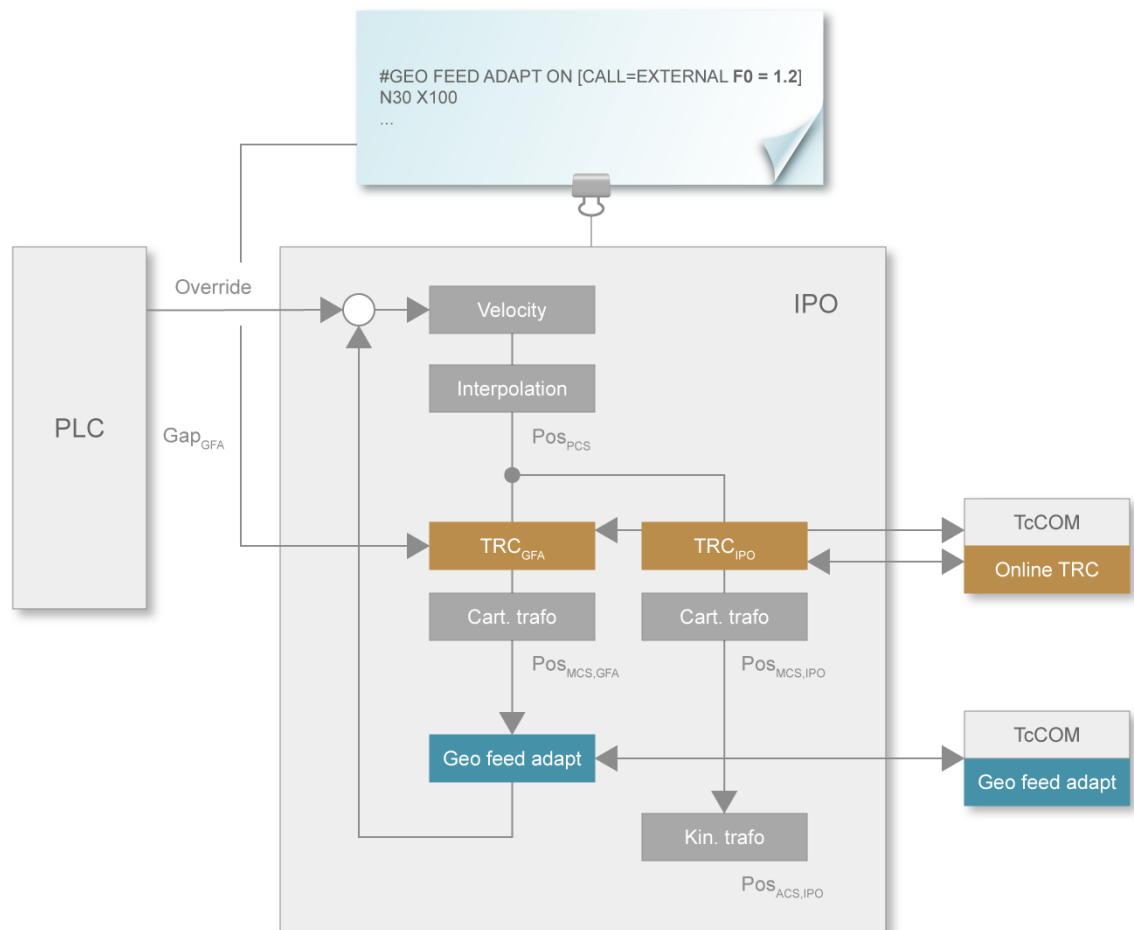


Abb. 13: Integration des Objekts im NC-Kanal

3.3 Anbindung einer geometrischen Vorschubanpassung via TcCOM

3.3.1 Methoden der Schnittstelle

Zu implementierende Methode

Folgende Methode sind für eine geometrische Vorschubanpassung zu implementieren (TcCnc-GeoFeedAdaptInterfaces.h):

- virtual HRESULT TCOMAPI CalculateFeedAdaption(PTcGeoFeedAdaptParam gfa)=0;

CalculateFeedAdaption	<p>In aktivem Zustand der Funktionalität erfolgt die Berechnung in jedem Interpolationsstakt.</p> <p>Abhängig vom Flächenverhältnis berechnet die Methode einen Übersteuerungsfaktor (Overridefaktor).</p> <p>Input: Aktuellen Parameter der geometrische Vorschubanpassung</p>
------------------------------	---

3.3.2 Instanzdaten

Arbeitsdaten - PTcGeoFeedAdaptParam

Parameter der Methoden:

Die Parameter für die einzelnen Methoden werden in gekapselter Form über die Struktur PTcGeoFeedAdaptParam (TcCncGeoFeedAdaptInterfaces.h) übergeben.

```
struct PTcGeoFeedAdaptParam
{
    // config: EcGeoFeedAdapt_ParamStandard
    EcGeoFeedAdapt    type;
    // config: dimension of path : 1 or 2
    int                dim_path;
    // config: dimension of tool parameters
    int                dim_tool_param;
    // tool parameters
    const double       * i_tool_param;
    // feed adaption factor
    double              * o_feed_adaption;

    struct TcGeoFeedAdaptPath path[2]; //
}
```

Pfadspezifische Parameter:

```
struct TcGeoFeedAdaptPath
{
    // input: 3-dim GFA position X,Y,Z
    const double       * i_position_gfa;
    // Input: 3-dim display position X,Y,Z
    const double       * i_position_display
}
```

3.3.3 Konfigurieren und Anmelden des TcCOM-Objekts

Registrieren in TwinCAT

Zum Anmelden eines TcCOM-Objekts werden folgende Daten verwendet (TcCncServices.h):

- Type 5 (TCCNC_REGISTEROBJECT_TYPE_GEO_FEED_ADAPT) fest vorgegeben.
- Verwendet das TcCOM-Objekt keine instanzspezifischen Variablen, so wird der Eintrag Group auf 0 gesetzt.
- Verwendet das TcCOM-Objekt instanzspezifische Variablen, so wird der Eintrag Group auf die jeweilige Kanalnummer [1;12] gesetzt, dem das Objekt zugeordnet ist. Maximal ein Objekt pro Kanal.
- Index wird nicht verwendet.

Die Anmeldung des Objekts der geometrischen Vorsteuerung erfolgt über nachfolgendes TcCOM-Interface, welches in der Datei TcCncInterfaces.h definiert ist:

- virtual HRESULT TCOMAPI RegisterObject(TcCncRegisterObject& id, ITcUnknown* ipUnk)=0;
- virtual HRESULT TCOMAPI UnregisterObject(TcCncRegisterObject& id)=0;

Bereitstellung des TcCOM-Objekts

Nach der Generierung der geometrischen Vorschubanpassung müssen 2 Dateien vorhanden sein:

1. TMC-Datei
2. Treiberdatei

Die Beschreibung der geometrischen Vorschubanpassung ist in der TMC-Datei enthalten, z.B. TcCncMyGeoFeedAdapt.tmc.

Diese ist im Arbeitsverzeichnis der Solution.

Das Verzeichnis der Treiberdatei ist abhängig von Release oder Debug:

- <TwinCAT> \3.1\sdk_products\TwinCAT RT (x64)\Release bzw.
- <TwinCAT> \3.1\sdk_products\TwinCAT RT (x64)\Debug

Mit der Aktivierung der Konfiguration wird die jeweilige Treiberdatei automatisch in das Verzeichnis <TwinCAT>\3.1\Driver\AutoInstall kopiert.

Anhand des obigen Beispielnamens: TcCncMyGeoFeedAdapt.sys



Hinweis

Der Anwender muss nur die Generierung (Debug/Release) auslösen und die entsprechende Konfiguration aktivieren.

Die Vorgehensweise zum Debuggen der erstellten geometrischen Vorschubanpassung ist analog zum Debuggen einer McCOM-Transformation. Diese Vorgehensweise wird in [McCOM-Trafo, Kapitel: Transformation debuggen] geschildert.

Laden des Objekts

Das Laden des Objekts für die geometrische Vorschubanpassung ist in „Objekt der geometrischen Vorschubanpassung einbinden [▶ 61]“ veranschaulicht.

3.4 Programmierung

Syntax:

#GEO FEED ADAPT [ON/OFF] [CALL=.. [I<i>=>=..] [F<j>=>=..] { I<i>=>=..}{ F<j>=>=..}]

ON Geometrische Vorschubanpassung aktivieren, analog zu P-CHAN-00386 [▶ 41].

OFF Geometrische Vorschubanpassung deaktivieren, analog zu P-CHAN-00386 [▶ 41].

CALL=.. Typ der verwendeten Methode der geometrischen Vorschubanpassung.

Zulässige Kennungen:

- DEFAULT - es wird die Methode aus P-CHAN-00387 [▶ 41] verwendet (Standard)
- EXTERNAL - Verwenden der kundenspezifischen Methode

I<i>=>=.. SGN32 Wert, analog zum Parameter P-CHAN-00391 [▶ 42].

Mit i=0 - 3.

Das Gleichheitszeichen ist zwingend erforderlich.

z.B. I2=17

F<j>=>=.. REAL64 Wert, analog zum Parameter P-CHAN-00390 [▶ 41].

Mit j=0 - 3.

Achtung:

F0=<Spaltwert> [0.1 µm] ist fest vergeben.

Das Gleichheitszeichen ist zwingend erforderlich,

z.B. F2=3.45

Im NC-Programm kann auf die folgenden V.G.-Variablen zugegriffen werden:

Variable	Bedeutung	Datentyp	Einheit der Ein-/Ausgabe	Erlaubter Zugriff: Lesen/Schreiben
V.G.GEO_FEED_ADAPT.ACTIVE	Zeigt an, ob die geometrischen Vorschubanpassung aktiv ist.	Boolean	0 / 1	L
V.G.GEO_FEED_ADAPT.CALL	Zeigt an, welche Methode der geometrischen Vorschubanpassung verwendet wird. 0: EXTERN	SGN32	0 / 1	L



Programmierbeispiel

Externe geometrische Vorschubanpassung in 2-Pfad-Anwendung

```

N180 #GEO FEED ADAPT ON [CALL=EXTERNAL F0=12.3]
N260 G141
N270 G139 G26
N270 G01 G42 : X10 Y0 : U12 V2
; ...
N400 G40 : X0 Y0 : U0 V0
N410 #GEO FEED ADAPT OFF
N420 M30
    
```

3.5 Parameter

P-CHAN-00386	An-/Abwahl der Funktion konstanter Flächenvorschub
Beschreibung	Mit diesem Parameter wird der konstante Flächenvorschub angewählt / abgewählt.
Parameter	geo_feed_adapt.active
Datentyp	BOOLEAN
Datenbereich	0/1
Dimension	----
Standardwert	0
Anmerkungen	Verfügbar ab V3.1.3108

P-CHAN-00387	Auswahl der Berechnungsmethode (Flächenvorschub)
Beschreibung	Mit diesem Parameter wird ausgewählt, ob der konstante Flächenvorschub in einem COM-Objekt berechnet wird oder den in der CNC integrierten Algorithmus verwendet.
Parameter	geo_feed_adapt.call
Datentyp	STRING
Datenbereich	EXTERNAL, BUILTIN
Dimension	----
Standardwert	EXTERNAL
Anmerkungen	Verfügbar ab V3.1.3108 BUILTIN ist nur für interne Tests

P-CHAN-00390	REAL64 Input-Parameter für den Flächenvorschub
Beschreibung	Es können insgesamt vier REAL64 Input-Parameter für den Flächenvorschub definiert werden. Achtung: f[0]=<Spaltwert> in [0.1 µm] Dieser Eintrag ist fest vergeben.
Parameter	geo_feed_adapt.param.f[i] mit i = 0...3
Datentyp	REAL64
Datenbereich	MIN(REAL64) ≤ f[i] ≤ MAX(REAL64)
Dimension	----
Standardwert	0.0
Anmerkungen	Verfügbar ab V3.1.3108

P-CHAN-00391	SGN32 Input-Parameter für den Flächenvorschub
Beschreibung	Es können insgesamt vier SGN32 Input-Parameter für den Flächenvorschub definiert werden.
Parameter	geo_feed_adapt.param.i[i] mit $i = 0 \dots 3$
Datentyp	SGN32
Datenbereich	$\text{MIN}(\text{SGN32}) \leq i[i] \leq \text{MAX}(\text{SGN32})$
Dimension	----
Standardwert	0
Anmerkungen	Verfügbar ab V3.1.3108

3.6 Fehlermeldungen

Fehler-ID	Beschreibung
ID 50752	Bei Aktivierung der geometrischen Vorschubanpassung ist ein Fehler aufgetreten.
ID 293100	Angeforderte Werkzeugradius-Kompensation ist nicht konfiguriert.
ID 293101	Ungenügend Speicher für Verwaltung des COM-Interfaces.
ID 293102	Verzeichnis der COM-Schnittstellenobjekte ist nicht vorhanden.
ID 293103	ISGCtrl ist noch nicht initialisiert.
ID 293104	Angegebenes COM-Interface ID ist nicht konfiguriert.
ID 293105	Angegebenes COM-Interface kann intern nicht gespeichert werden.

4 Dynamische Konturvorsteuerung

4.1 Übersicht

Aufgabe

Durch physikalische Verformung des Werkzeugs können Unterschiede zur programmierten Kontur entstehen. Mithilfe der dynamischen Konturvorsteuerung können diese Differenzen zur Laufzeit kompensiert werden.



Versionshinweis

Diese Funktionalität ist ab CNC-Version V3.1.3108 verfügbar.



Hinweis

Diese Funktionalität ist Bestandteil einer lizenzpflichtigen Zusatzoption.

Eigenschaft

Die Funktionalität kann abhängig vom verwendeten Berechnungsmodus

- intern mit CNC-Funktionen oder
- extern durch ein anwenderspezifisches TcCOM-Objekt

verwendet werden.

Programmierung / Parametrierung

Die Funktionalität kann über den NC-Befehl #DCC [▶ 49] und über die entsprechenden Kanalparameter [▶ 53] konfiguriert, aktiviert und deaktiviert werden.

Verweise 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.

4.2 Beschreibung

Aufgrund einer physikalischen Verformung des Werkzeugs (z.B. dem Erodierdraht) kann die resultierende Kontur von der programmierten Kontur abweichen. Um diesen Fehler zu kompensieren, modifiziert die dynamische Konturvorsteuerung (engl. Dynamic Contour Control = DCC) die interpolierte Werkzeugmittelpunktbahn in Abhängigkeit vom aktuellen und vorherigen Konturelement, den Werkzeugparametern und einem Online-Einfluss der SPS (z.B. der Geschwindigkeitsabhängigkeit).

Nachfolgend ist die Wirkungsweise der dynamischen Konturvorsteuerung veranschaulicht: Die Aufgabe der Konturvorsteuerung ist es, durch entsprechende Modifikation die Sollkontur so zu beeinflussen, dass die reale Istkontur der programmierten Sollkontur entspricht.

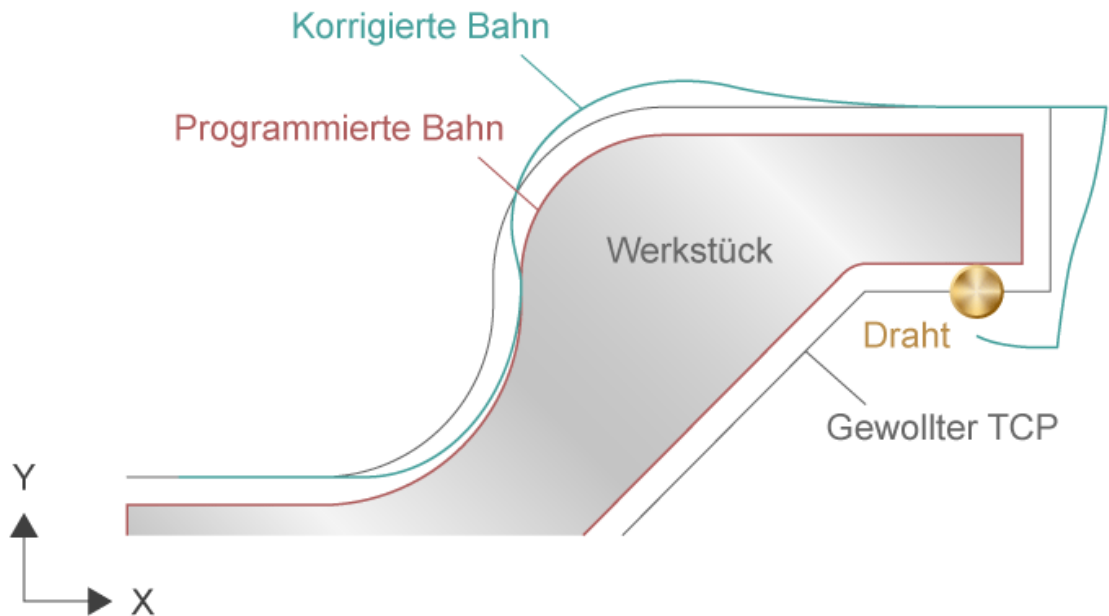


Abb. 14: Geometrische Wirkungsweise der programmierten und korrigierten Bahn

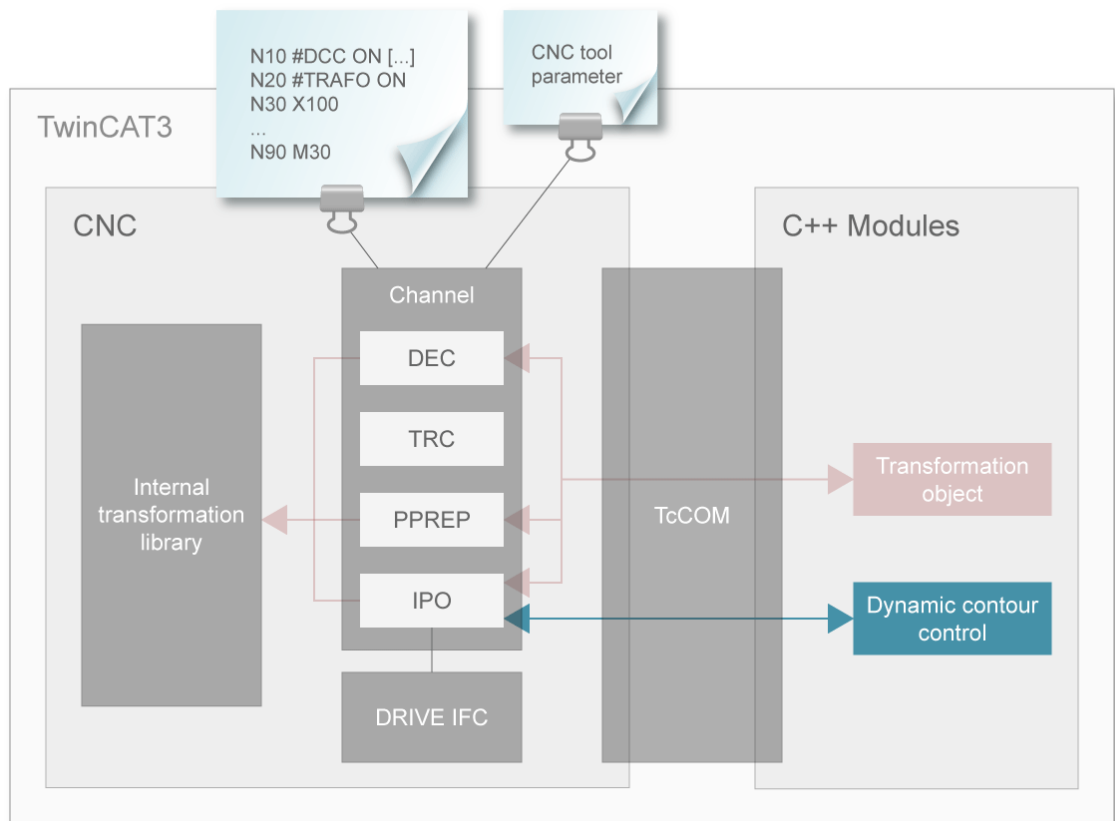


Abb. 15: Anordnung der dynamischen Konturvorsteuerung im System



Hinweis

Bei einem TcCOM-Objekt mit instanzspezifischen Variablen muss das Objekt im jeweiligen CNC-Kanal konfiguriert werden.

Die Festlegung des Berechnungsmodus erfolgt entweder über P-CHAN-00385 [▶ 53] oder im NC-Befehl #DCC [▶ 49][CALL=..]

- EXTERNAL, externe Vorgabe
- BULTIN, CNC-interner Algorithmus

4.2.1 Externe Berechnungsmodus

#DCC[CALL=EXTERNAL]

Die Berechnung der Versätze und die Überlagerung mit den Achspositionen können in den Methoden des TcCOM-Objekts durchgeführt werden.

Nachfolgend ist die Anordnung im System einer 2-Pfad-Konfiguration dargestellt. Die Überlagerung der Offsets erfolgt in der CNC:

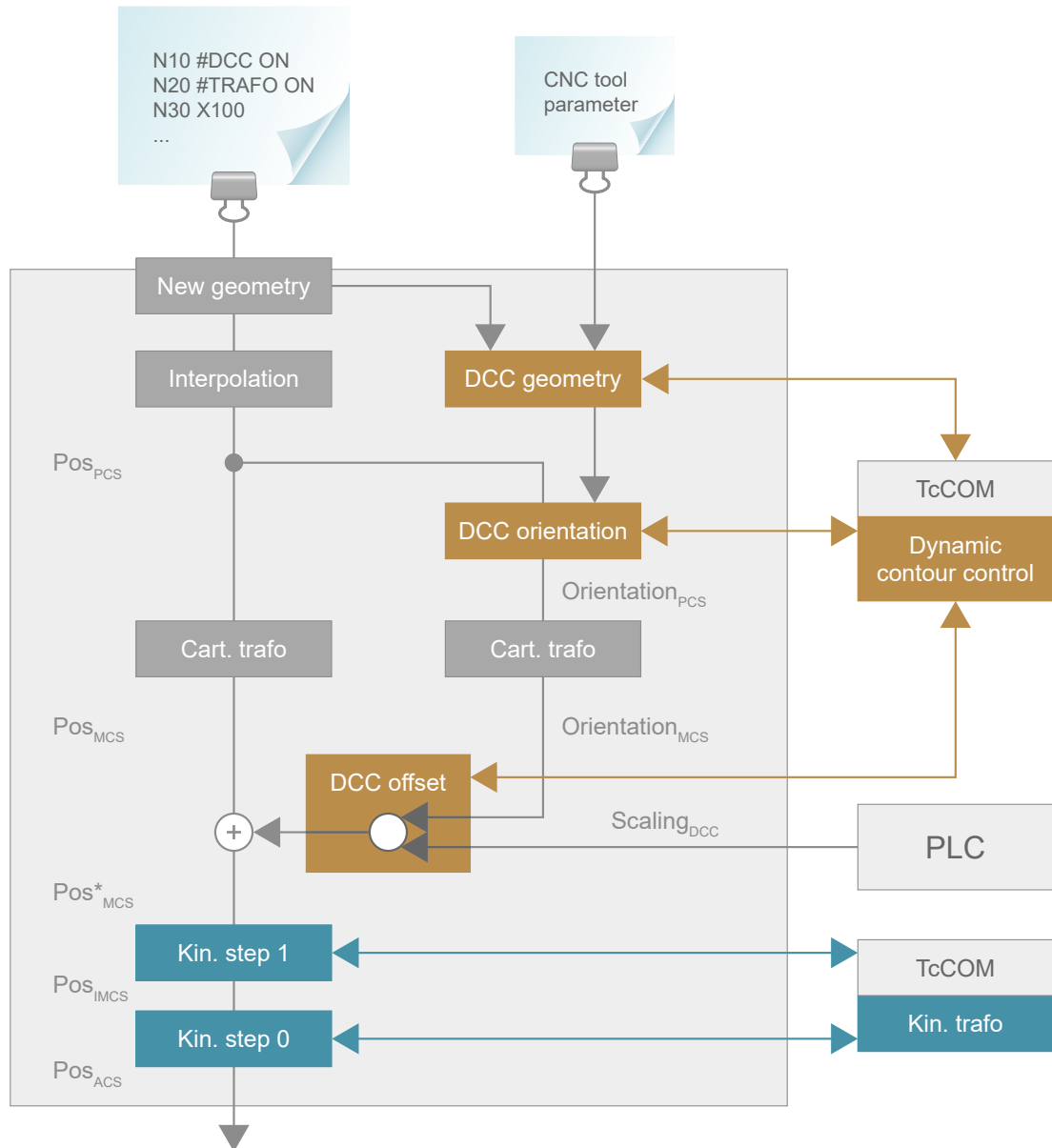


Abb. 16: Dynamische Konturvorsteuerung mit impliziter Überlagerung

Die Berechnung der Bewegung im ACS System erfolgt durch die kinematische Transformation. Zusätzlich zu den Standardparametern wird der kinematischen Rückwärtstransformation die Orientierung der korrigierten Geometrie im MCS mitgegeben.

Der Skalierungsfaktor kann dem TcCOM-Objekt durch die SPS über den zyklischen Input bereitgestellt werden.

Nachfolgend ist die Anordnung im System dargestellt mit der Überlagerung von Offsets im TcCOM-Objekt der kinematischen Transformation:

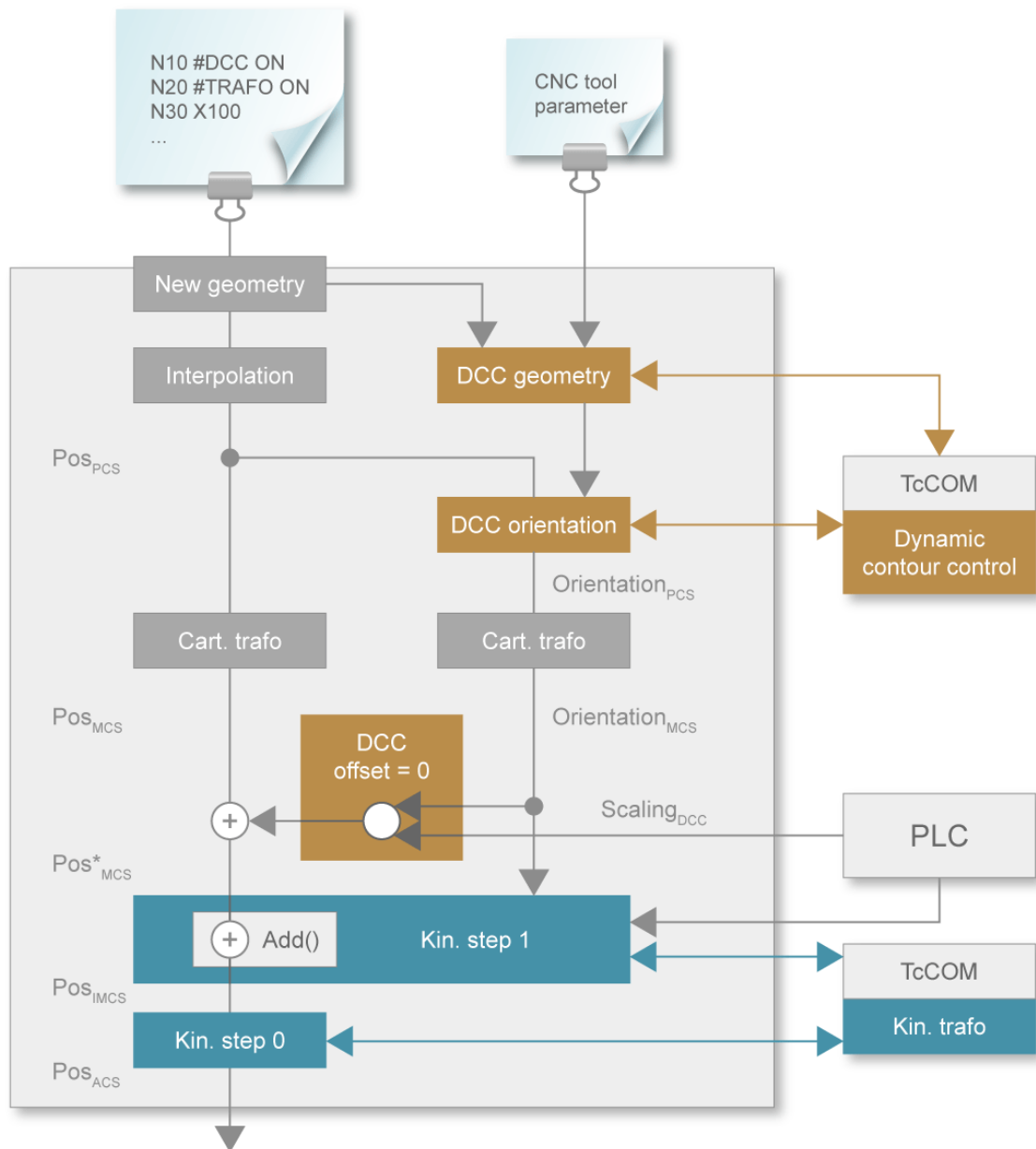


Abb. 17: Dynamische Konturvorsteuerung mit expliziter Offsetüberlagerung in erster Kinematikstufe



Hinweis

Sollen die Offsets der dynamische Konturvorsteuerung nicht implizit von der CNC überlagert werden, so muss die Funktion „CalculateOffset()“ Offsets mit dem Wert 0 zurückgeben.

Der Anwender kann die Berechnung und Überlagerung der Offsets in einem TcCOM-Objekt vom Typ „Kinematische Transformation“ durchführen.

4.3 Programmierung

Syntax:

#DCC [ON/OFF] [CALL=.. [I<i>=..] [F<j>=..] { I<i>=..}{ F<j>=..}]

- ON** Dynamische Konturvorsteuerung aktivieren, analog zu P-CHAN-00384 [▶ 53].
- OFF** Dynamische Konturvorsteuerung deaktivieren, analog zu P-CHAN-00384 [▶ 53].
- CALL=..** Typ der verwendeten Methode der dynamischen Konturvorsteuerung.
Zulässige Kennungen:
- DEFAULT - es wird die Methode aus P-CHAN-00385 [▶ 53] verwendet (Standard)
 - EXTERNAL - Verwenden der kundenspezifischen Methode
- I<i>=..** SGN32 Wert, analog zum Parameter P-CHAN-00389 [▶ 54].
Mit i=0 - 3.
Das Gleichheitszeichen ist zwingend erforderlich,
z.B. I2=17
- F<j>=..** REAL64 Wert, analog zum Parameter P-CHAN-00388 [▶ 53].
Mit j=0 - 3.
Das Gleichheitszeichen ist zwingend erforderlich,
z.B. F2=3.45

Im NC-Programm kann auf die folgenden V.G.-Variablen zugegriffen werden:

Variable	Bedeutung	Datentyp	Einheit der Ein-/ Ausgabe	Erlaubter Zugriff: Lesen/ Schreiben
V.G.DCC.ACTIVE	Zeigt an ob die dynamische Konturvorsteuerung aktiv ist.	Boolean	0 / 1	L
V.G.DCC.CALL	Zeigt an welche Methode der dynamischen Konturvorsteuerung verwendet wird. 0: EXTERN	SGN32	0 / 1	L



Programmierbeispiel

#DCC-Varianten der Konturvorsteuerung

```

; Verwenden der Standardeinstellungen
N10 #DCC ON ; Aktivierung
N15 #DCC OFF ; Abwahl DCC

; Verwenden der externen Methode
N20 #DCC ON [CALL=EXTERNAL] ; Aktivierung
N30 #DCC [CALL=DEFAULT] ; Rücksetzen auf Standard
N40 #DCC ON ; Aktivierung mit aktueller Methode

; Externe Methode mit Parametern
N50 #DCC ON [CALL=EXTERNAL I0=1 F0=2.2 I1=7]
    
```

4.4 Anbindung einer dynamischen Konturvorsteuerung via TcCOM

4.4.1 Methoden der Schnittstelle

Zu implementierende Methoden

Folgende Methoden sind für eine dynamische Konturüberwachung zu implementieren (TcCncDynContourControlInterface.h):

- virtual HRESULT TCOMAPI CalculateOnNewGeometry(PTcDynContourCtrlParam dcc)=0;
- virtual HRESULT TCOMAPI CalculateOrientation(PTcDynContourCtrlParam dcc)=0;
- virtual HRESULT TCOMAPI CalculateOffset(PTcDynContourCtrlParam dcc)=0;

CalculateOnNewGeometry	<p>Berechnung erfolgt mit jedem neuen Geometrieelement.</p> <p>Abhängig vom aktuellen Geometrieübergang (nachfolgende Tabelle) und den Werkzeugparametern berechnet die Methode einen Normalisierungsfaktor.</p> <p>Input: Aktuelle Parameter der dynamischen Konturvorsteuerung.</p>
CalculateOrientation	<p>Berechnung erfolgt mit jeder neuen Position.</p> <p>Abhängig von der aktuellen Position berechnet die Methode die neue Orientierung der Kompensation.</p> <p>Input: Aktuelle Parameter der dynamischen Konturvorsteuerung.</p>
CalculateOffset	<p>Berechnung erfolgt mit jeder neuen Position.</p> <p>Berechnung der Achsoffsets im MCS, abhängig von der zuvor ermittelten Orientierung der Kompensation. Die Achsoffsets werden den Achspositionen überlagert.</p> <p>Input: Aktuelle Parameter der dynamischen Konturvorsteuerung.</p> <p>Hinweis:</p> <p>Werden die Offsets in der kinematischen Transformation überlagert, so müssen die Offsets dieser Funktion den Wert 0 haben.</p>

Wert-Geometrieübergang	Bedeutung
0	Positionswechsel
1	Stillstand in Linearbewegung
2	Stillstand in Zirkularbewegung
3	Übergang von Linearbewegung zu Linearbewegung
4	Übergang von Zirkularbewegung zu Linearbewegung
5	Übergang von Linearbewegung zu Zirkularbewegung
6	Übergang von Zirkularbewegung zu Zirkularbewegung

4.4.2 Instanzdaten

Arbeitsdaten - TcCncDynContourCtrlParameter

Parameter der Methoden

Die Parameter für die einzelnen Methoden werden in gekapselter Form über die Struktur TcCncDynContourCtrlParam (TcCncDynContourCtrlInterfaces.h) übergeben:

```
struct TcDynContourCtrlParam
{
    // config: EcDynContourCtrl_ParamStandard
    EcDynContourCtrl    type;
    // config: dimension of path : 1 or 2
    int                 dim_path;
    // config: dimension of tool parameter
    int                 dim_tool_param;

    // input : <n>-dimensional actual parameter of tool
    const double        * i_tool_param;
    // input : path velocity [m/s]
    double               i_velocity;
    struct TcDynContourCtrlPath path[2];
}

```

Pfadspezifische Parameter

```
struct TcDynContourCtrlPath
{
    // calculation on new block: CalculateOnNewGeometry()
    // input: type of element, see enum EtDynContourCtrl
    int                 i_element_type;
    // input: radius of circle
    double              i_circle_radius;
    // input : actual tangent change
    //(== 0 for C2-continuous contour elements), in radiant
    double              i_tangent_variation;
    // output: normalization factor, dependent on geometry and technology
    (tool parameters): X0
    double              * o_norm_factor;

    // calculation orientation on new sample
    // position: CalculateOrientation()
    // input : normalization factor: X0
    double              i_norm_factor;
    // input : 3-dim actual position of path: X,Y,Z
    const double        * i_position;
    // in-/output : 3-dim normalized orientation
    double              * o_orientation_pcs;

    // calculation offset: CalculateOffset()
    // input : 3-dim normalized orientation
    double              * i_orientation_mcs;
    // output: 3-dim offset of path
    double              * o_offset;
}

```

4.4.3 Konfigurieren und Anmelden des TcCOM-Objekts

Registrieren in TwinCAT

Zum Anmelden eines TcCOM-Objekts werden folgende Daten verwendet (TcCncServices.h):

- Type 4 (TCCNC_REGISTEROBJECT_TYPE_DYN_CONTOUR_CTRL) fest vorgegeben.
- Verwendet das TcCOM-Objekt keine instanzspezifischen Variablen, so wird der Eintrag Group auf 0 gesetzt.
- Verwendet das TcCOM-Objekt instanzspezifischen Variablen, so wird der Eintrag Group auf die jeweilige Kanannummer [1;12] gesetzt, zu dem das Objekt zugeordnet ist. Maximal ein Objekt pro Kanal.
- Index wird nicht verwendet.

Die Anmeldung des Objekts der dynamischen Konturvorsteuerung erfolgt über nachfolgendes TcCOM-Interface, welches in der Datei TcCncInterfaces.h definiert ist:

- virtual HRESULT TCOMAPI RegisterObject(TcCncRegisterObject& id, ITcUnknown* ipUnk)=0;
- virtual HRESULT TCOMAPI UnregisterObject(TcCncRegisterObject& id)=0;

Bereitstellung des TcCOM-Objekts

Nach der Generierung der dynamischen Konturvorsteuerung müssen 2 Dateien vorhanden sein:

1. TMC-Datei
2. Treiberdatei

Die Beschreibung der dynamischen Konturvorsteuerung ist in der TMC-Datei enthalten, z.B. TcCncMyDynContCtrl.tmc.

Diese ist im Arbeitsverzeichnis der Solution.

Das Verzeichnis der Treiberdatei ist abhängig von Release oder Debug:

- <TwinCAT> \3.1\sdk_products\TwinCAT RT (x64)\Release bzw.
- <TwinCAT> \3.1\sdk_products\TwinCAT RT (x64)\Debug

Mit der Aktivierung der Konfiguration wird die jeweilige Treiberdatei automatisch in das Verzeichnis <TwinCAT>\3.1\Driver\AutoInstall kopiert.

Anhand des obigen Beispielnamens: TcCncMyDynContCtrl.sys



Hinweis

Der Anwender muss nur die Generierung (Debug/Release) auslösen und die entsprechende Konfiguration aktivieren.

Die Vorgehensweise zum Debuggen der erstellten dynamischen Konturvorsteuerung ist analog zum Debuggen einer McCOM-Transformation. Diese Vorgehensweise wird in [McCOM-Trafo// Transformation debuggen] geschildert.

Laden des Objekts

Das Laden des Objekts für die dynamische Konturvorsteuerung ist in „Objekt der dynamischen Konturvorsteuerung einbinden [▶ 65]“ veranschaulicht.

4.5 Parameter

P-CHAN-00384	An-/Abwahl der Funktion dynamische Konturvorsteuerung (Dynamic Contour Control)
Beschreibung	Mit diesem Parameter wird die Funktion 'Dynamische Konturvorsteuerung' an-/ abgewählt.
Parameter	dcc.active
Datentyp	BOOLEAN
Datenbereich	0/1
Dimension	----
Standardwert	0
Anmerkungen	Verfügbar ab V3.1.3108

P-CHAN-00385	Auswahl der Berechnungsmethode (Konturvorsteuerung)
Beschreibung	Dieser Parameter legt fest, dass die Funktion dynamische Konturvorsteuerung mit einem TcCOM-Objekt berechnet wird.
Parameter	dcc.call
Datentyp	STRING
Datenbereich	EXTERNAL, BUILTIN
Dimension	----
Standardwert	EXTERNAL
Anmerkungen	Verfügbar ab V3.1.3108 BUILTIN ist nur für interne Tests

P-CHAN-00388	REAL64 Input-Parameter der Konturvorsteuerung (Dynamic Contour Control)
Beschreibung	Es können insgesamt vier REAL64 Input-Parameter für die dynamische Konturvorsteuerung definiert werden.
Parameter	dcc.param.f[i] mit $i = 0 \dots 3$
Datentyp	REAL64
Datenbereich	$\text{MIN}(\text{REAL64}) \leq f[i] \leq \text{MAX}(\text{REAL64})$
Dimension	----
Standardwert	0.0
Anmerkungen	Verfügbar ab V3.1.3108

P-CHAN-00389	SGN32 Input-Parameter der Konturvorsteuerung (Dynamic Contour Control)
Beschreibung	Es können insgesamt vier SGN32 Input-Parameter für die dynamische Konturvorsteuerung definiert werden.
Parameter	dcc.param.i[i] mit $i = 0 \dots 3$
Datentyp	SGN32
Datenbereich	$\text{MIN}(\text{SGN32}) \leq i[i] \leq \text{MAX}(\text{SGN32})$
Dimension	----
Standardwert	0
Anmerkungen	Verfügbar ab V3.1.3108

4.5.1 Parametrierbeispiel

Alle Einstellungen der dynamischen Konturvorsteuerung können in den Kanalparametern durchgeführt werden. Diese sind bei jedem Programmstart gültig und können im NC-Programm über den #DCC-Befehl [► 49] modifiziert werden.

Standardeinstellungen:

```
dcc.active FALSE
dcc.call EXTERNAL
```

Parametrierbeispiel mit zusätzlichen Werten:

```
dcc.active TRUE
dcc.call EXTERNAL
dcc.i0 1
dcc.f0 2.3
dcc.i1 2
dcc.f1 17.3
```

4.6 Fehlermeldungen

Fehler-ID	Beschreibung
ID 50732	Reservierter Speicher für TwinCAT3 TcCOM-Interface der dynamischen Konturvorsteuerung zu klein.
ID 50733	Bei Aktivierung der dynamischen Konturvorsteuerung ist ein Fehler aufgetreten.
ID 50734	Die Option zur dynamischen Konturvorsteuerung ist in der aktuellen Softwareversion nicht enthalten.
ID 50735	Die Berechnung CalculateOnNewGeometry() der dynamischen Konturvorsteuerung hat einen Fehler gemeldet.
ID 50736	Die Berechnung CalculateOrientation() der dynamischen Konturvorsteuerung hat einen Fehler gemeldet.
ID 50737	Die Berechnung CalculateOffset() der dynamischen Konturvorsteuerung hat einen Fehler gemeldet.
ID 293100	Angeforderte Werkzeugradius-Kompensation ist nicht konfiguriert.
ID 293101	Ungenügend Speicher für Verwaltung des COM-Interfaces.
ID 293102	Verzeichnis der COM-Schnittstellenobjekte ist nicht vorhanden.
ID 293103	ISGCtrl ist noch nicht initialisiert.
ID 293104	Angegebenes COM-Interface ID ist nicht konfiguriert.
ID 293105	Angegebenes COM-Interface kann intern nicht gespeichert werden.

5 Erstellungsablauf eines TcCOM-Objekts

Mindestanforderung für die Nutzung der McCOM-Assistenten

- TwinCAT3 Version 4024
- Microsoft Visual Studio 2019 Professional/Enterprise, bei der Installation muss die Option "Desktop development with C++" zusätzlich selektiert sein.

5.1 Neues Projekt anlegen

Der nachfolgende exemplarische Ablauf zur Erstellung einer eigenen kinematischen Transformation mittels TcCOM-Objekt wurde mit Visual Studio 2019 durchgeführt.

TwinCAT3 XAE Projekt mit CNC-Konfiguration

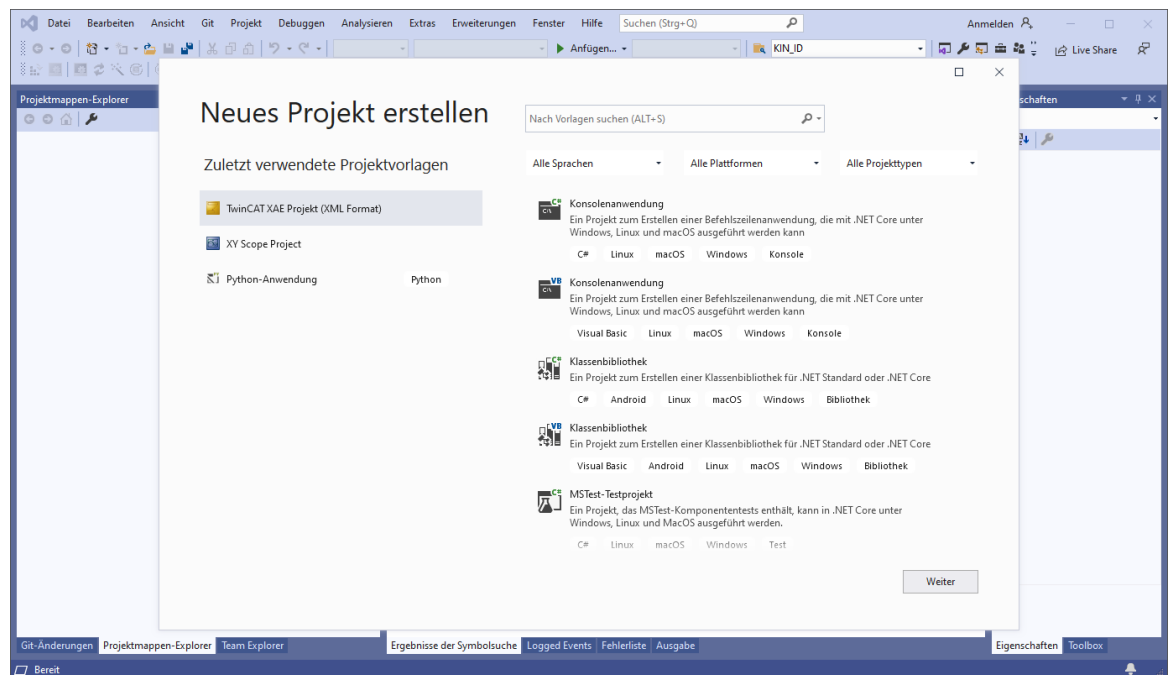


Abb. 18: Erstellung eines neuen Projekts

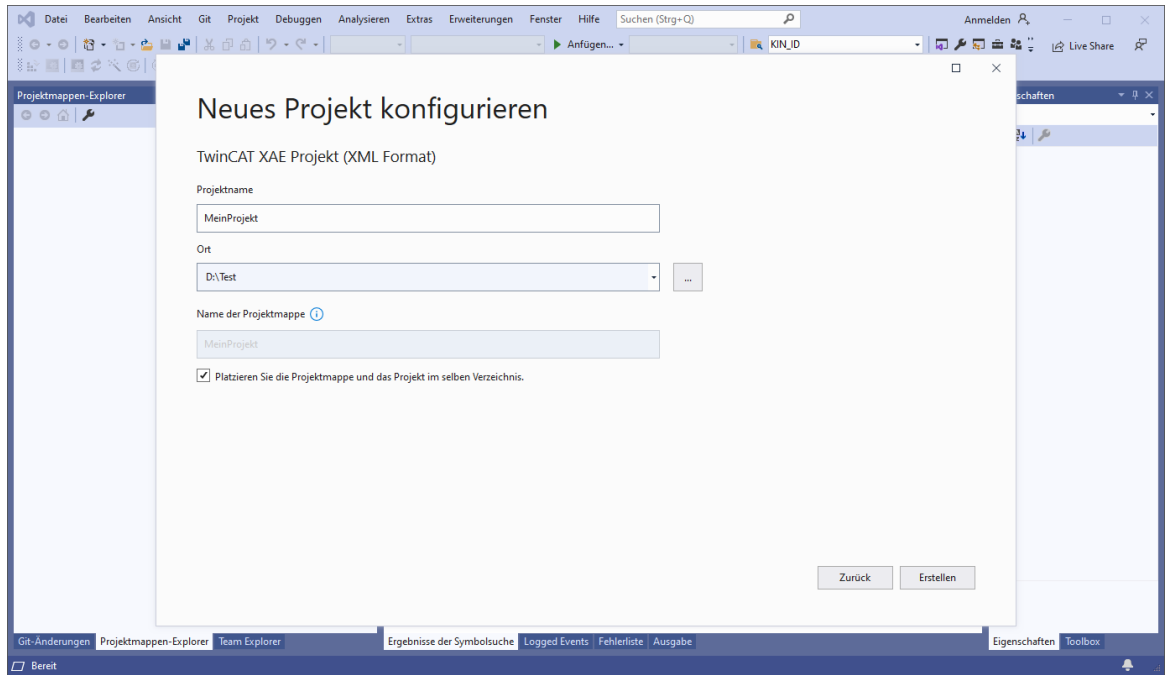


Abb. 19: Konfiguration des neuen Projekts

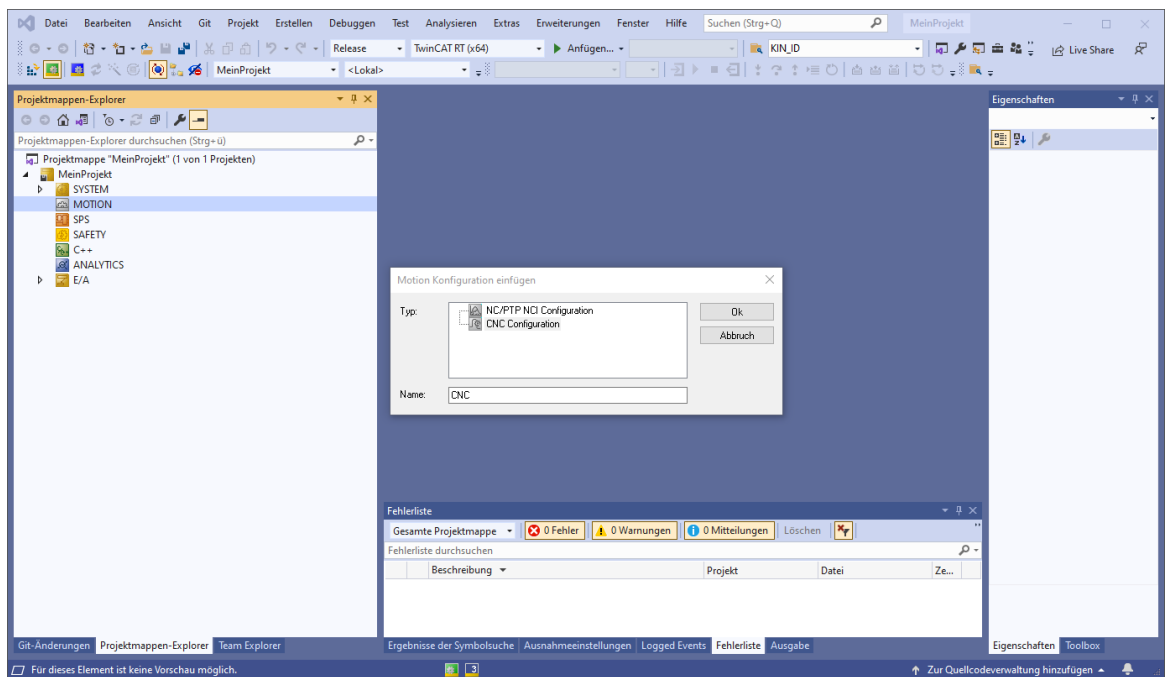


Abb. 20: Anlegen einer CNC-Konfiguration

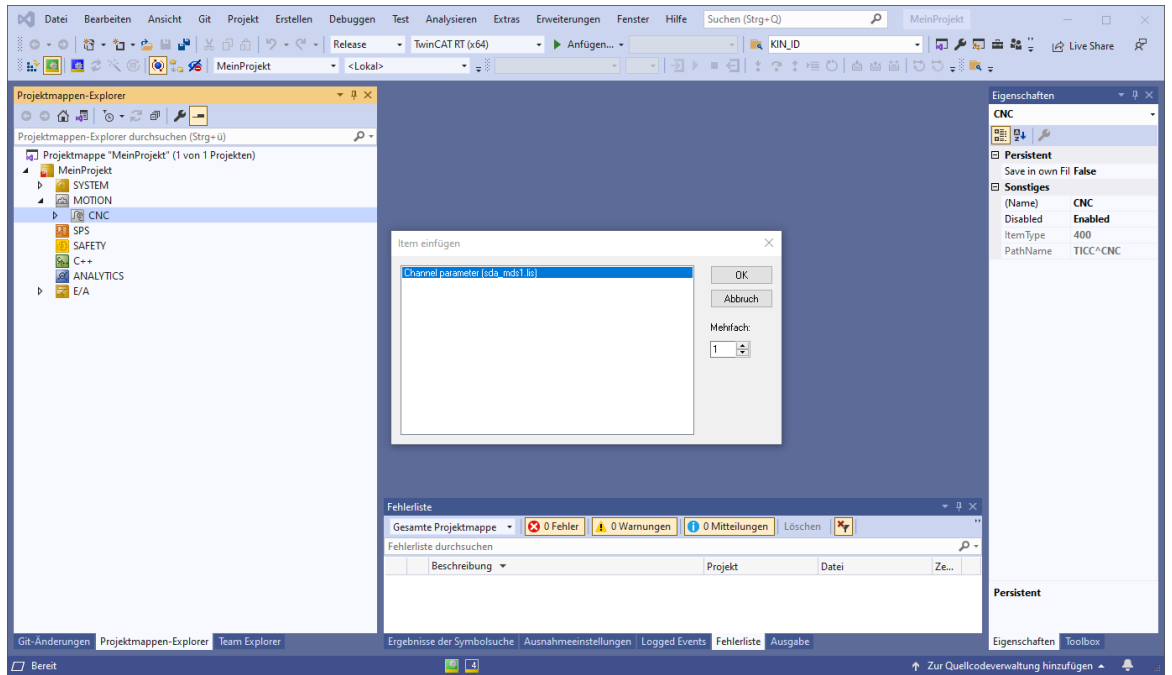


Abb. 21: Anlegen eines Kanals

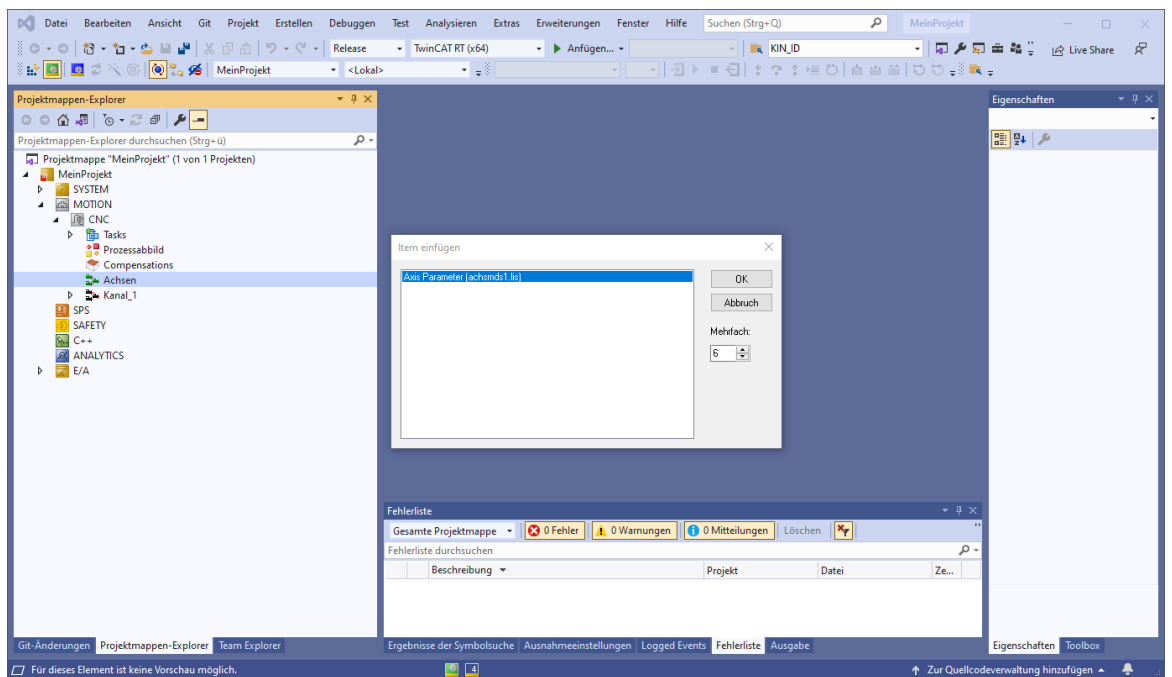


Abb. 22: Anlegen einer Achse

5.2 Geometrische Vorschubanpassung

5.2.1 Erstellen eines Objekts für geometrische Vorschubanpassung

Der nachfolgende exemplarische Ablauf zur Erstellung eines eigenen Objekts der geometrische Vorschubanpassung wurde mit Visual Studio 2019 durchgeführt.

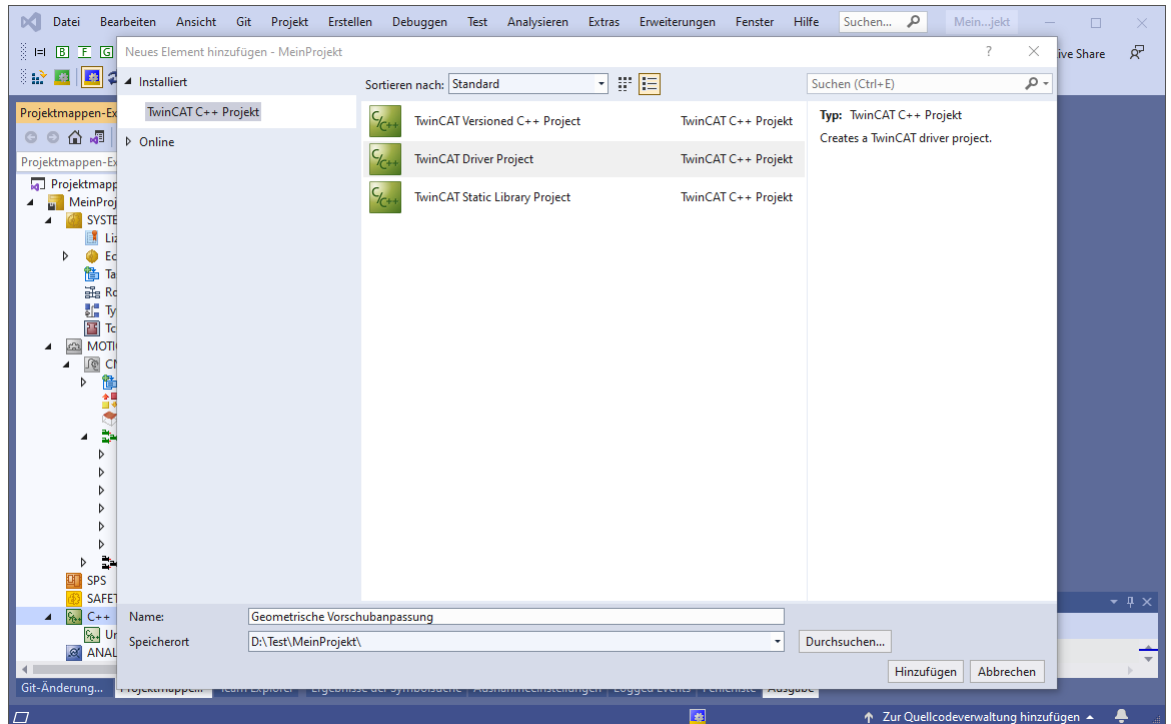


Abb. 23: Treiberprojekt für geometrische Vorschubanpassung

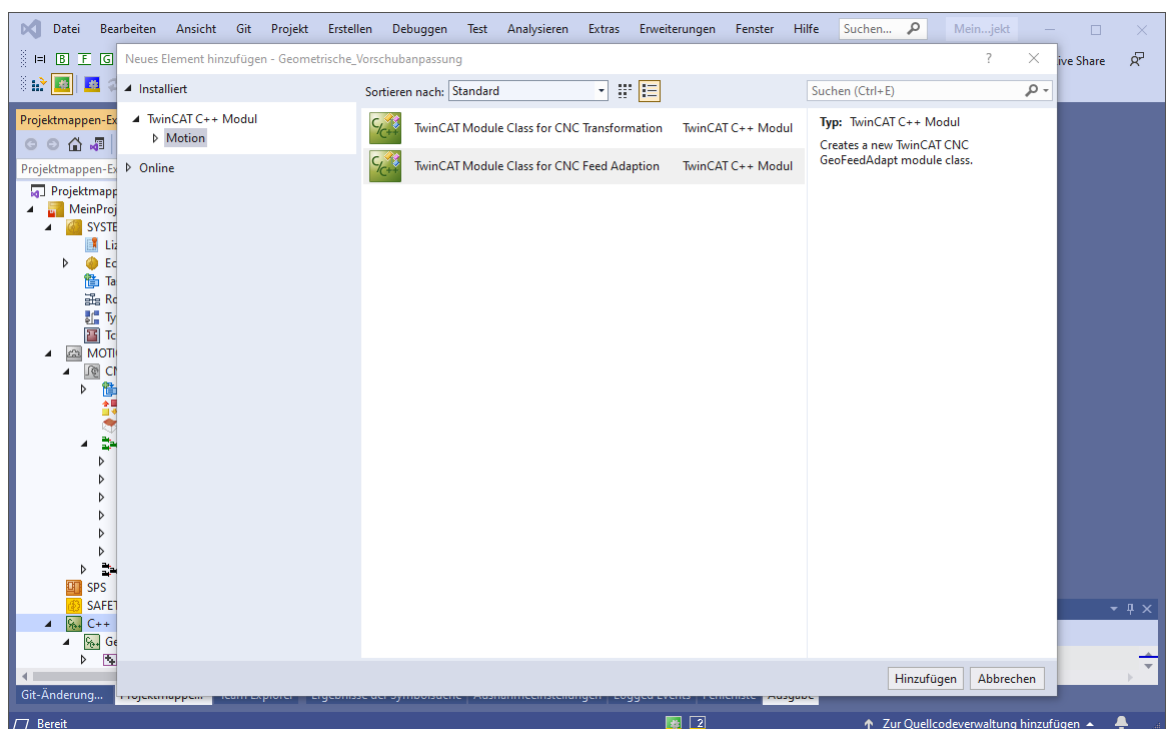


Abb. 24: Klasse festlegen

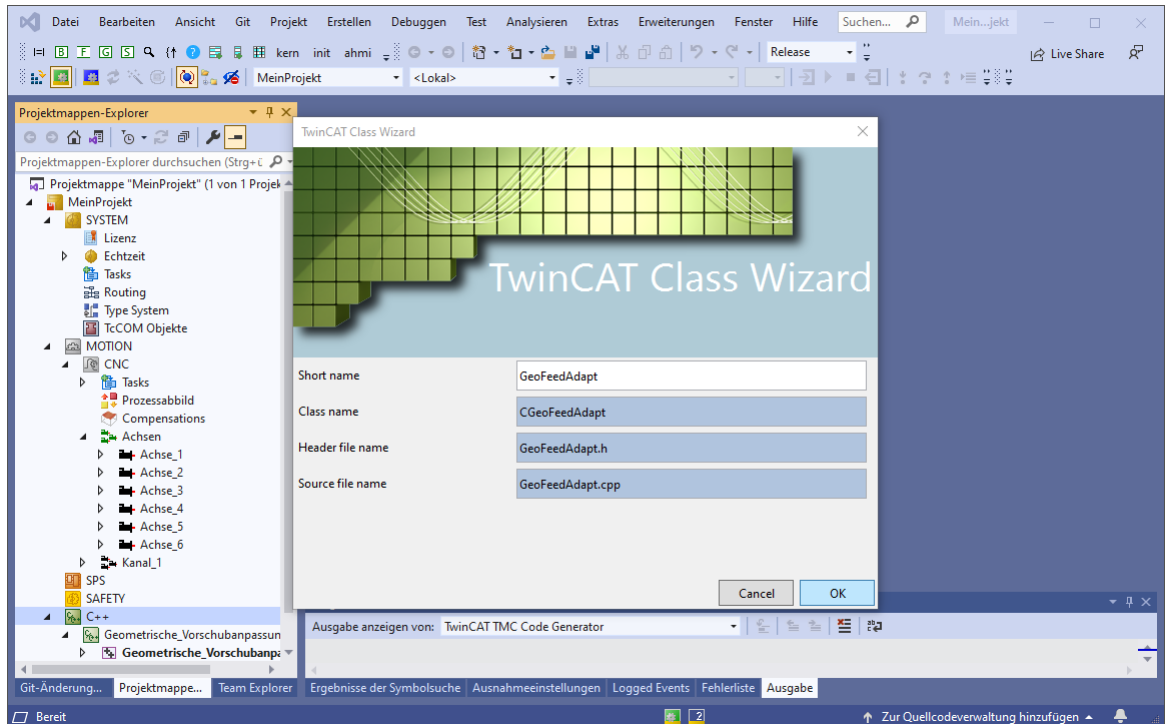


Abb. 25: Benennung der Klasse

Über einen Rechtsklick auf das Projekt kann mit „Erstellen“ der Treiber erstellt werden.

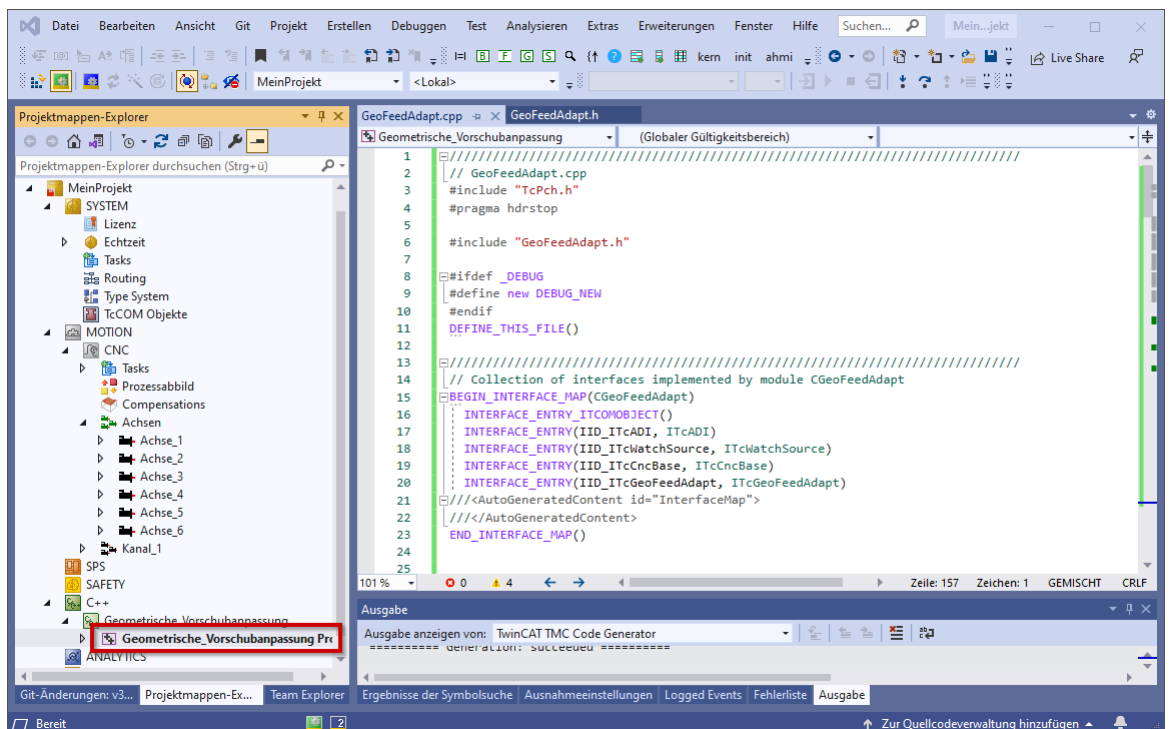


Abb. 26: Treiber erstellen

5.2.2 Objekt der geometrischen Vorschubanpassung einbinden

Das erstellte Objekt der geometrischen Vorschubanpassung muss in die vorhandene CNC-Konfiguration wie folgt integriert werden:

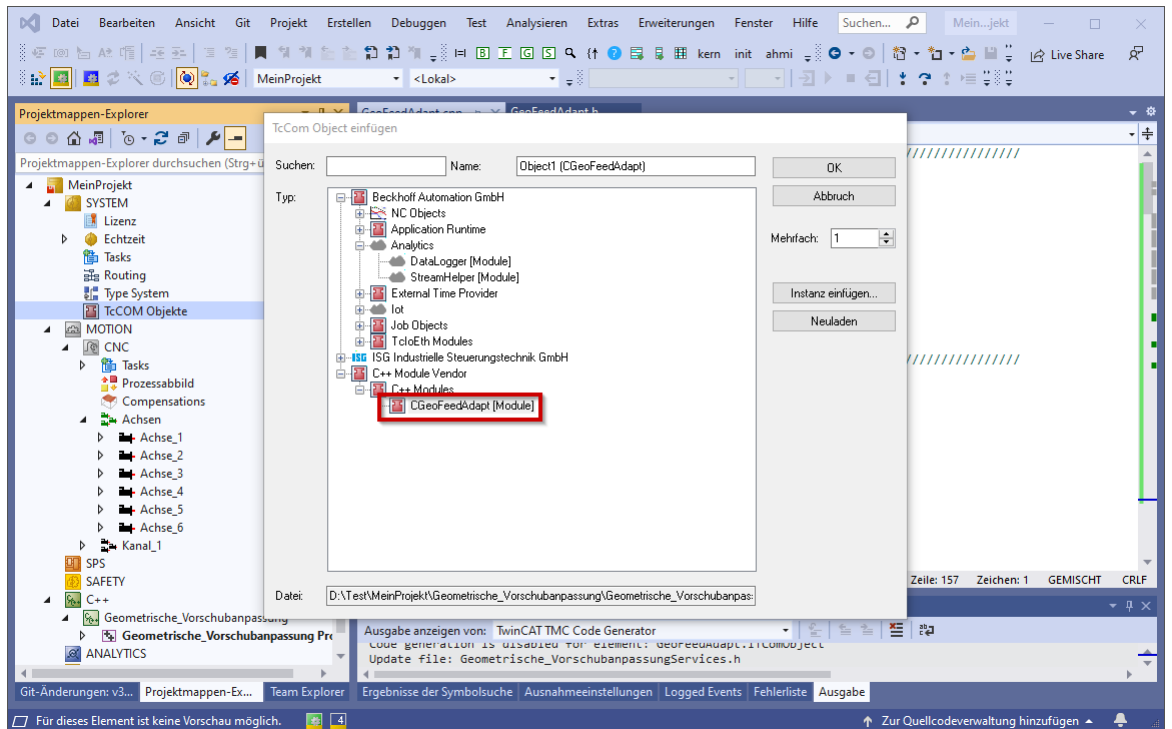


Abb. 27: Integration des TcCOM-Objekts

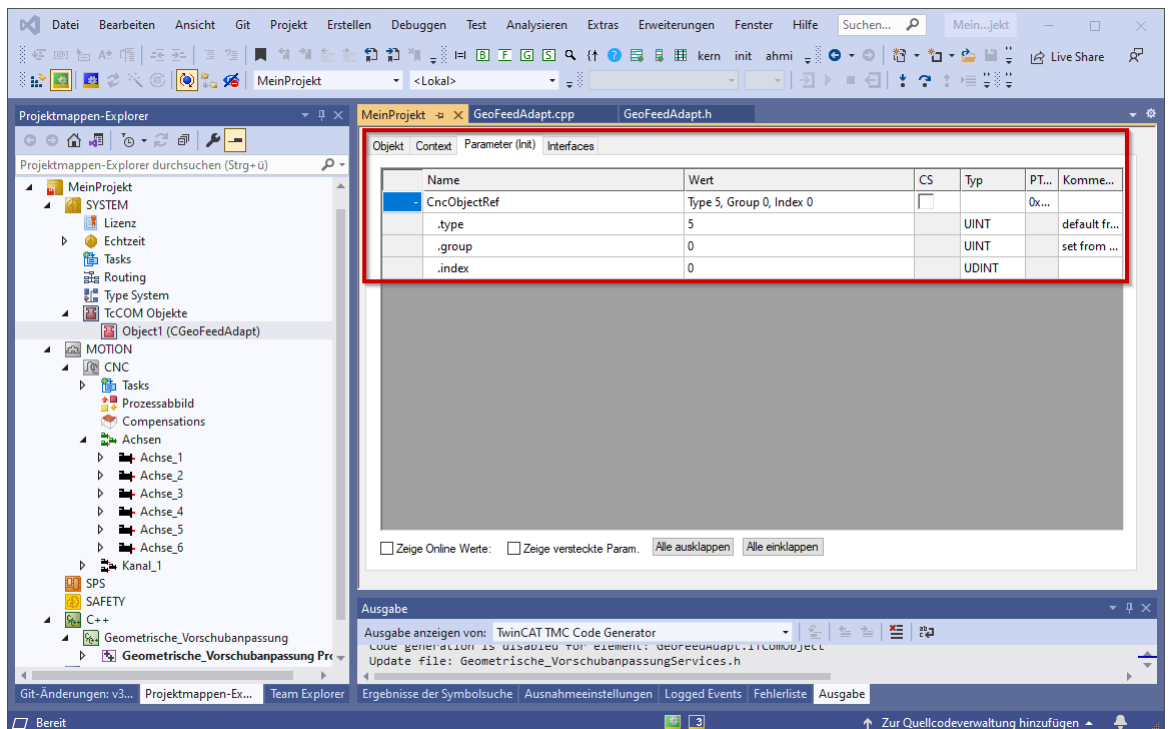


Abb. 28: Eigenschaften des TcCOM-Objekts

Der Typ für das Objekts der geometrischen Vorschubanpassung ist 5.

Die Angabe des Kanals ist abhängig davon, ob instanzspezifische Variablen im TcCOM-Objekt verwendet werden.

- Mit instanzspezifischen Daten entspricht die Gruppe der Kanalnummer [1;12].
- Ohne instanzspezifischen Daten ist die Gruppe mit 0 zu belegen.

Der Index des jeweiligen Objekts wird nicht verwendet.

5.3 Dynamische Konturvorsteuerung

5.3.1 Erstellen eines Objekts für dynamische Konturvorsteuerung

Der nachfolgende exemplarische Ablauf zur Erstellung eines eigenen Objekts der dynamischen Konturvorsteuerung wurde mit Visual Studio 2019 durchgeführt.

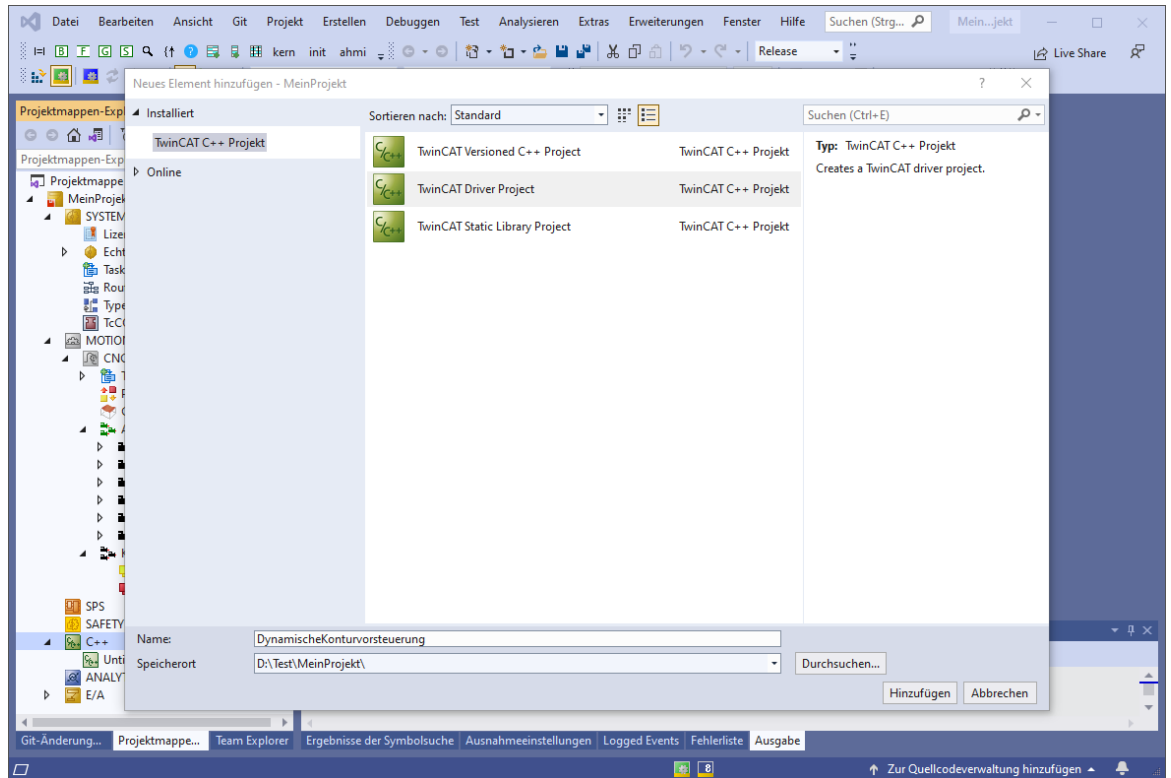


Abb. 29: Treiberprojekt für dynamische Kontursteuerung anlegen

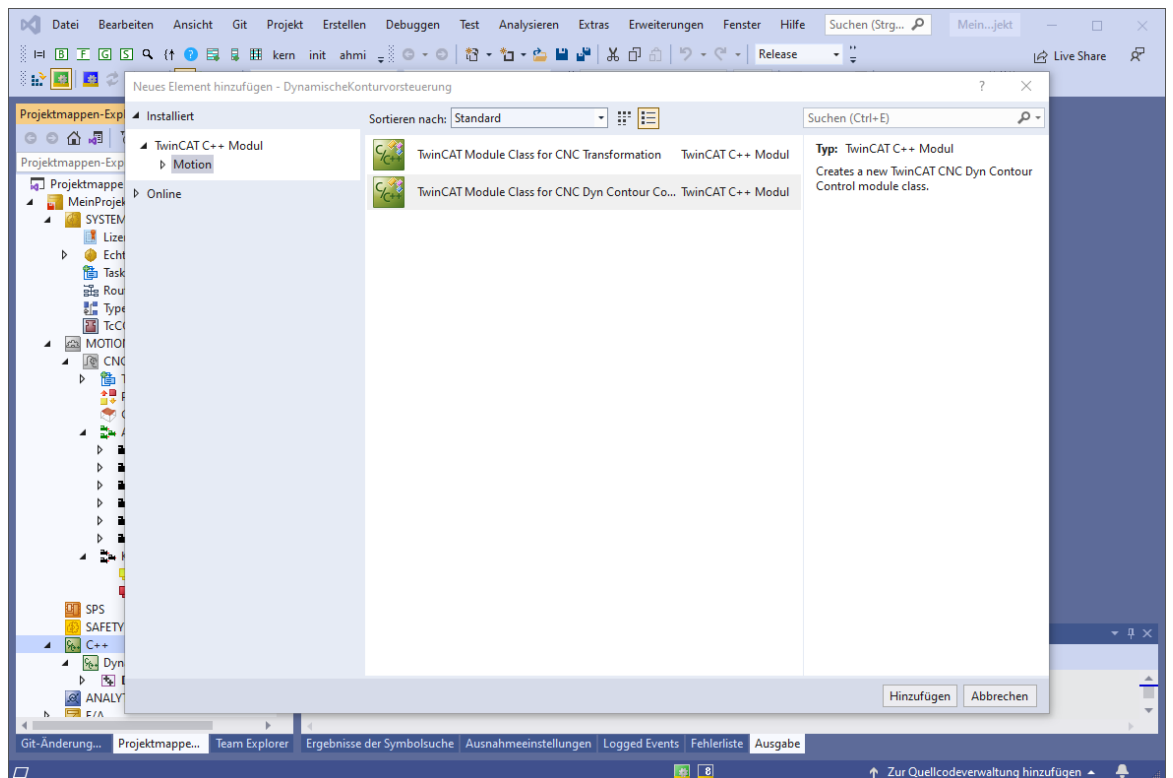


Abb. 30: Klasse festlegen

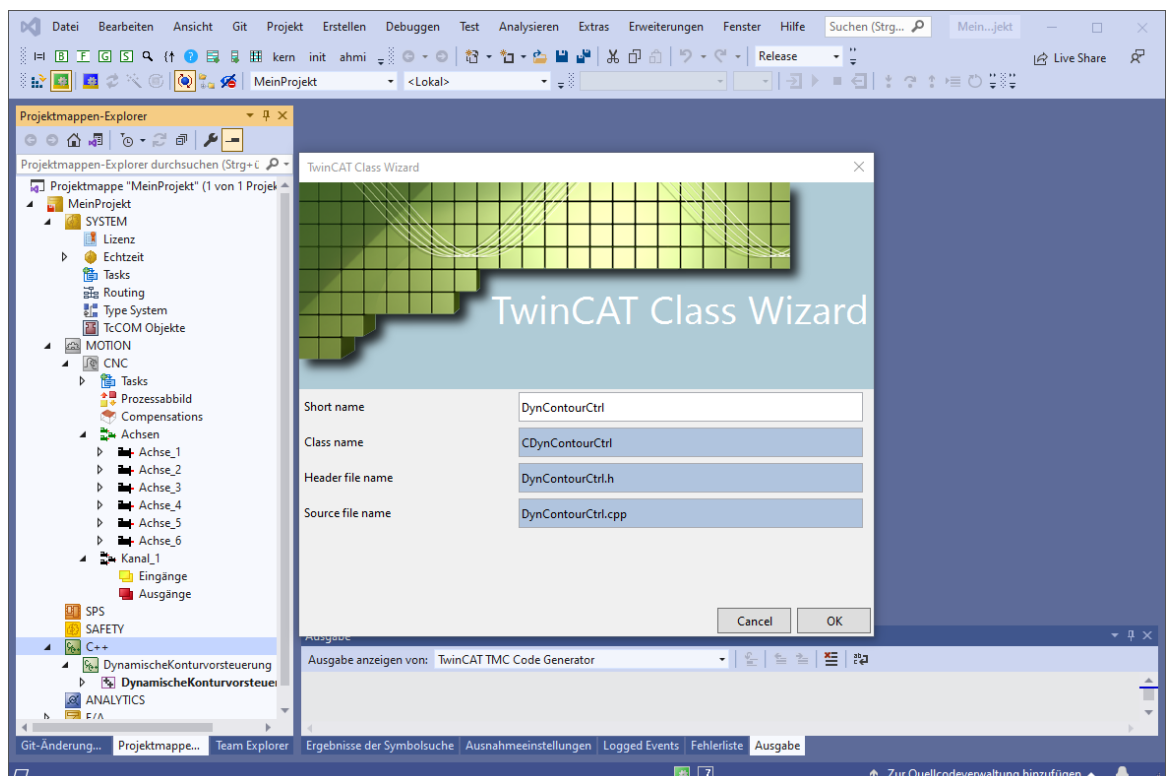


Abb. 31: Benennung der Klasse

Über einen Rechtsklick auf das Projekt kann mit „Erstellen“ der Treiber erstellt werden

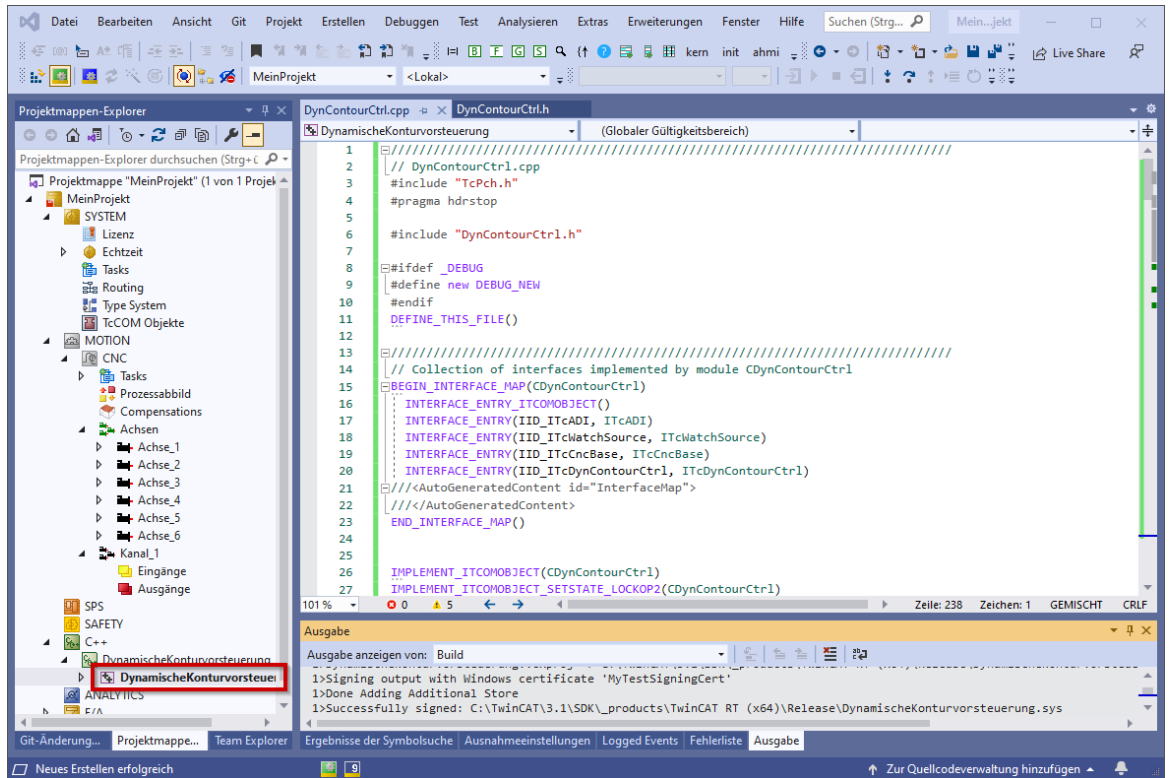


Abb. 32: Treiber erstellen

5.3.2

Objekt der dynamischen Konturvorsteuerung einbinden

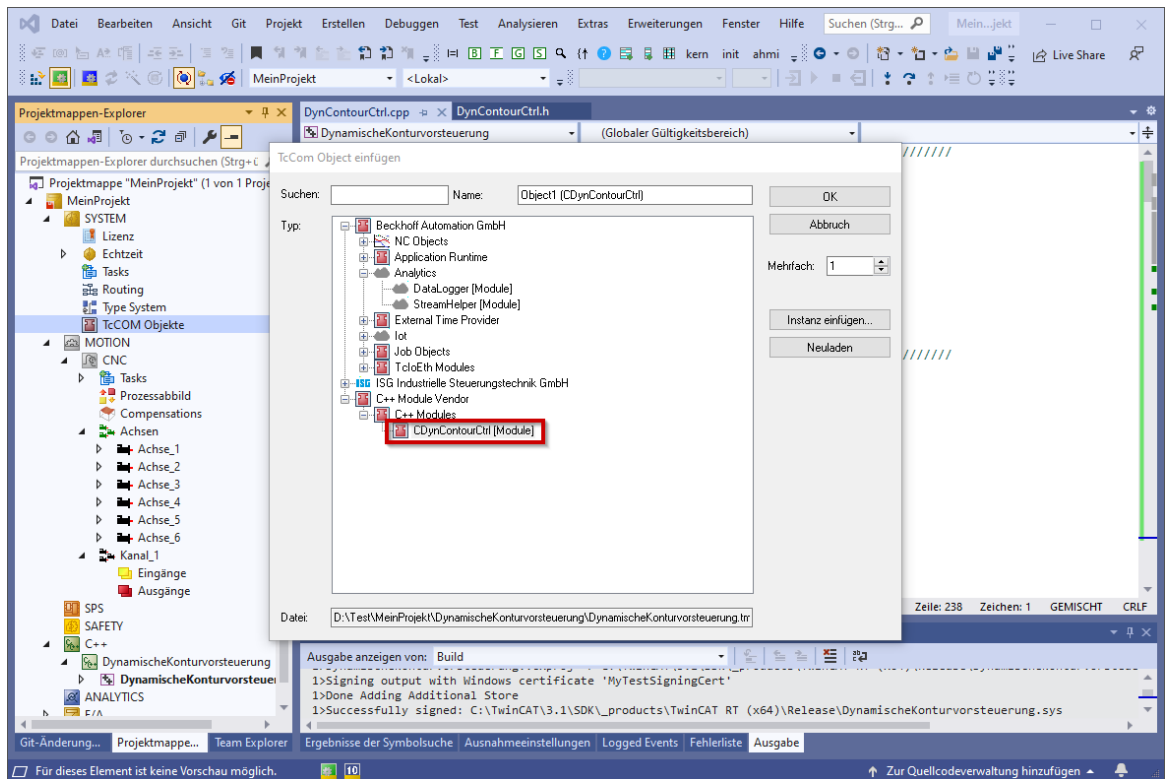


Abb. 33: Integration des TcCOM-Objekts

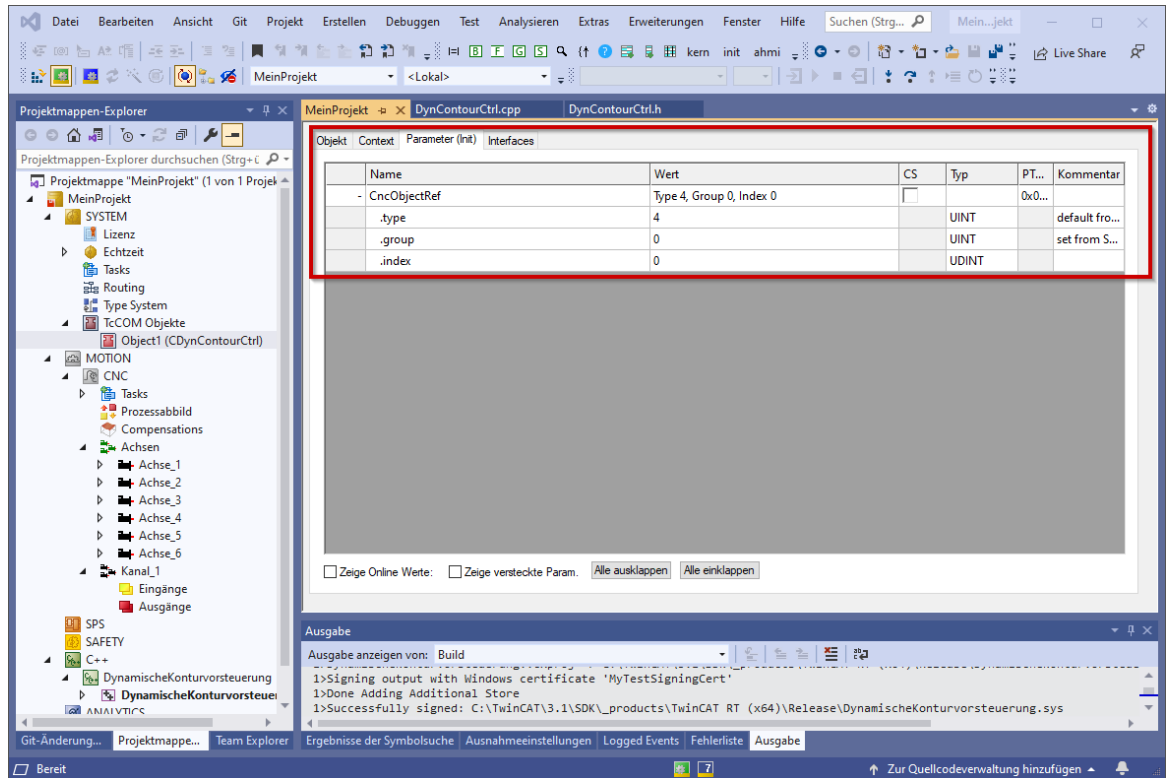


Abb. 34: Eigenschaften des TcCOM-Objekts

Der Typ für das Objekts der dynamischen Konturvorsteuerung ist 4.

Die Angabe des Kanals ist abhängig davon, ob instanzspezifische Variablen im TcCOM-Objekt verwendet werden.

- Mit instanzspezifischen Daten entspricht die Gruppe der Kanalnummer [1;12].
- Ohne instanzspezifischen Daten ist die Gruppe mit 0 zu belegen.

Der Index des jeweiligen Objekts wird nicht verwendet.



Hinweis

Das TcCOM-Objekt muss digital signiert sein, ansonsten kann dieses nicht geladen werden. Weitere Informationen zu TcCOM-Objekten und wie ein Objekt digital signiert werden kann, finden Sie im [Beckhoff Information System](#).

Es kann für jeden Kanal nur ein Dynamikmodell verwendet werden.

5.4 Online Werkzeugradiuskompensation

5.4.1 Erstellen eines Objekts für Online-Werkzeugradiuskompensation

Der nachfolgende exemplarische Ablauf zur Erstellung eines eigenen Objekts der Online-Werkzeugradiuskompensation wurde mit Visual Studio 2019 durchgeführt.

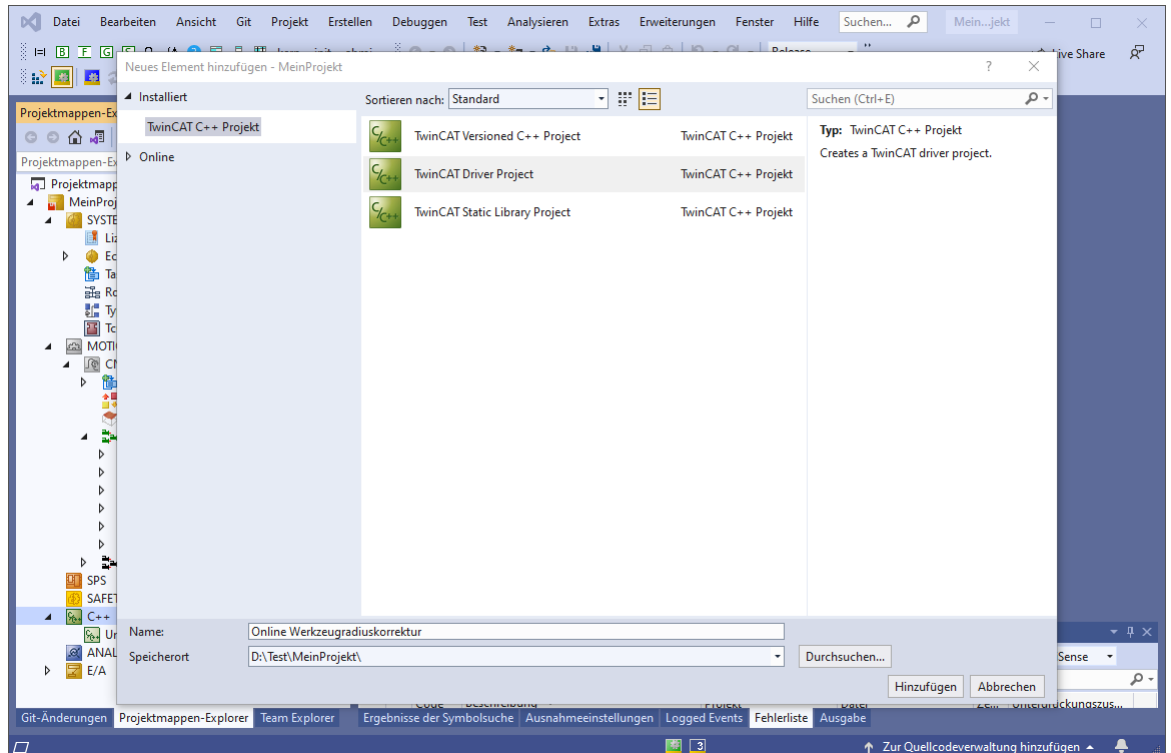


Abb. 35: Treiberprojekt für Online-Werkzeugradiuskompensation

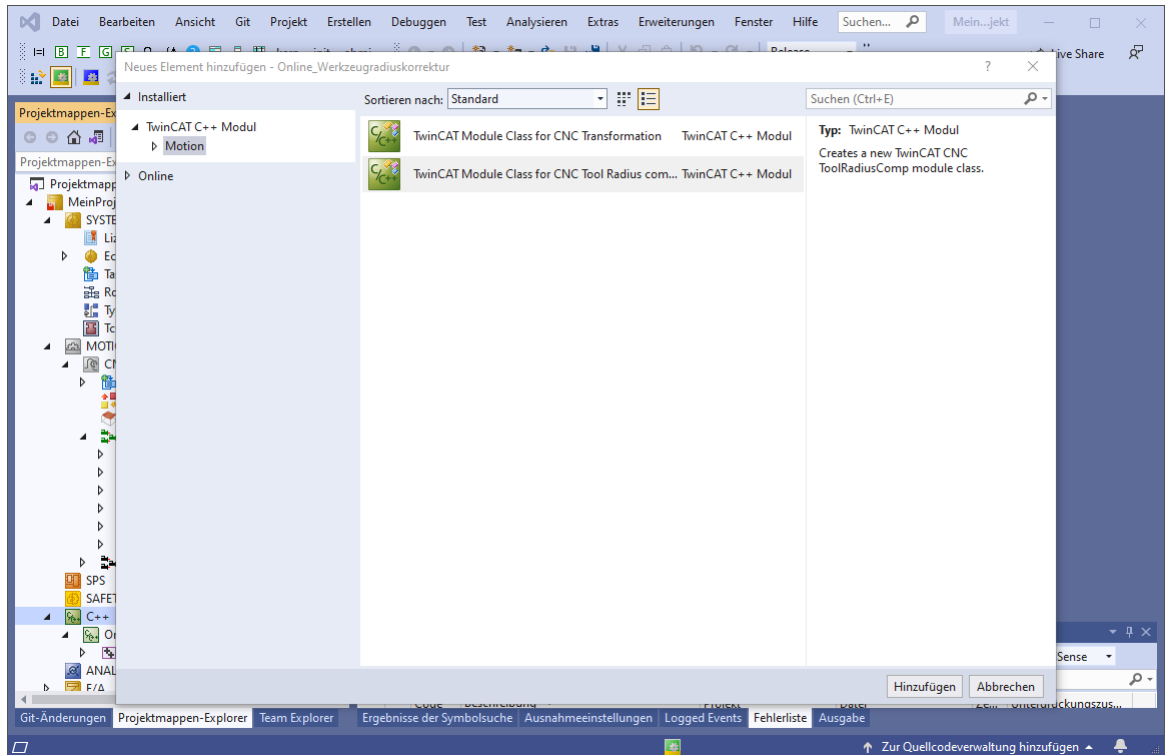


Abb. 36: Transformationsklasse festlegen

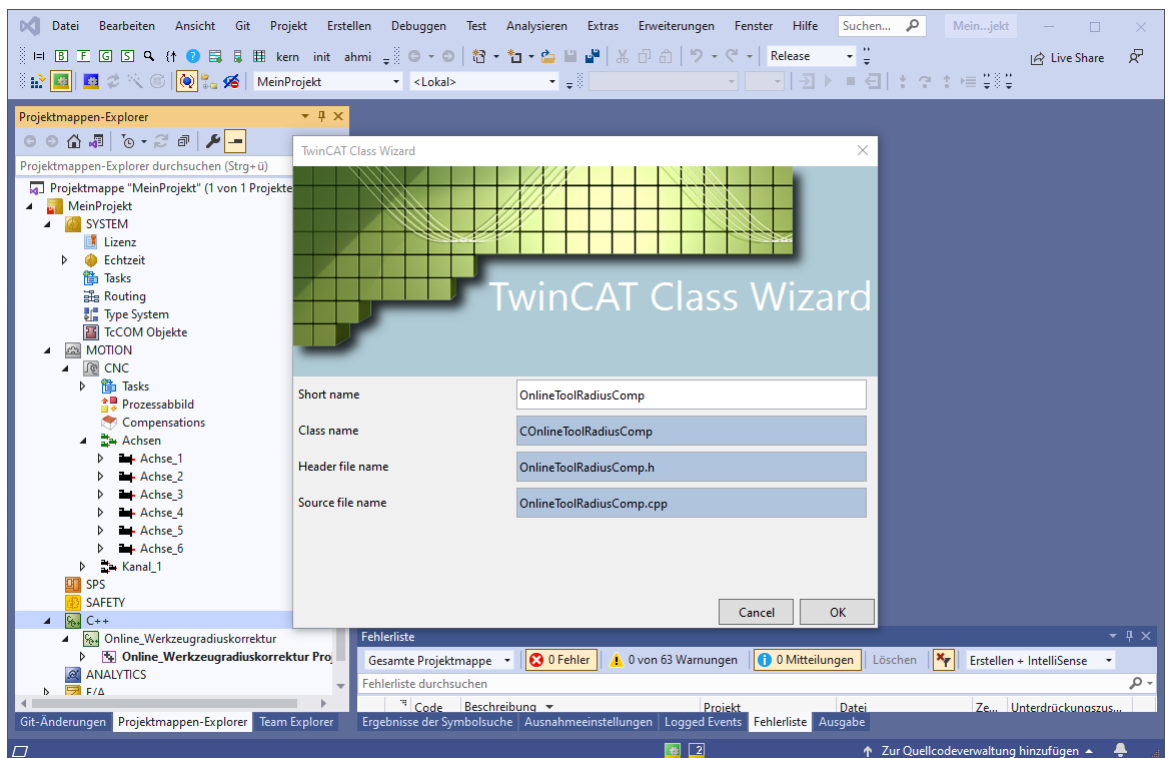


Abb. 37: Benennen der Transformationsklasse

Über einen Rechtsklick auf das Projekt kann mit „Erstellen“ der Treiber erstellt werden.

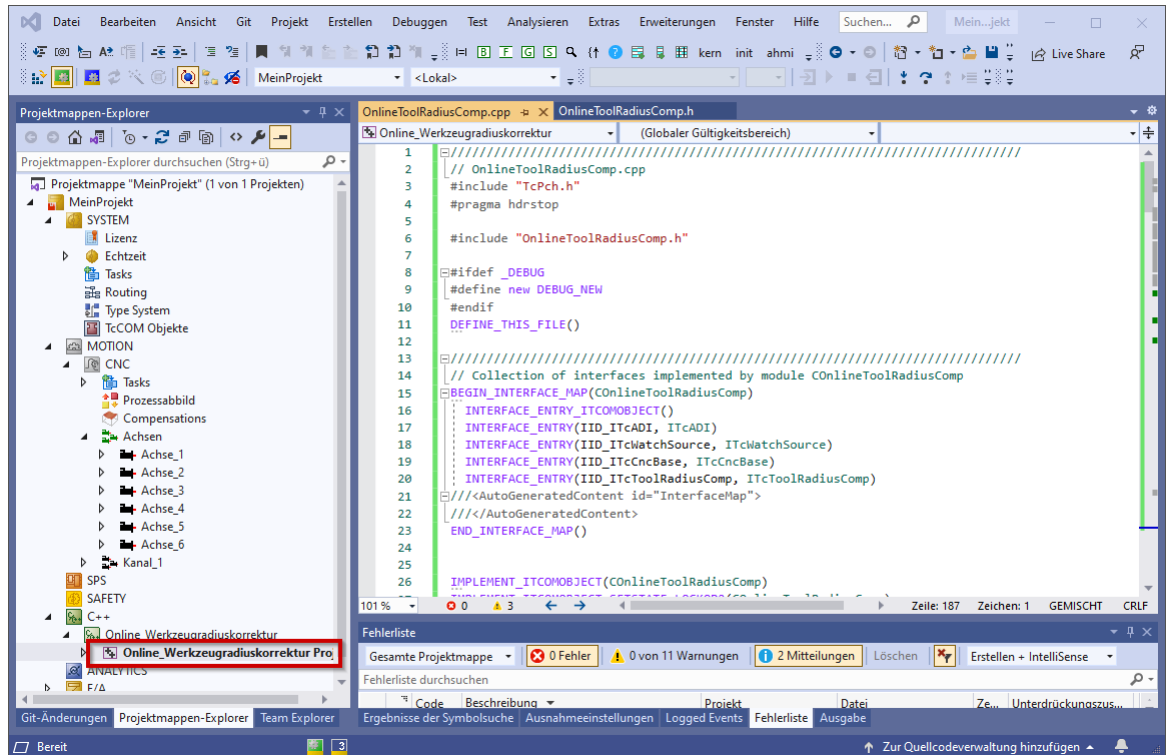


Abb. 38: Treiber erstellen

5.4.2

Objekt der Online-Werkzeugradiuskorrektur einbinden

Das erstellte Objekt der Online-Werkzeugradiuskorrektur muss in die vorhandene CNC-Konfiguration wie folgt integriert werden:

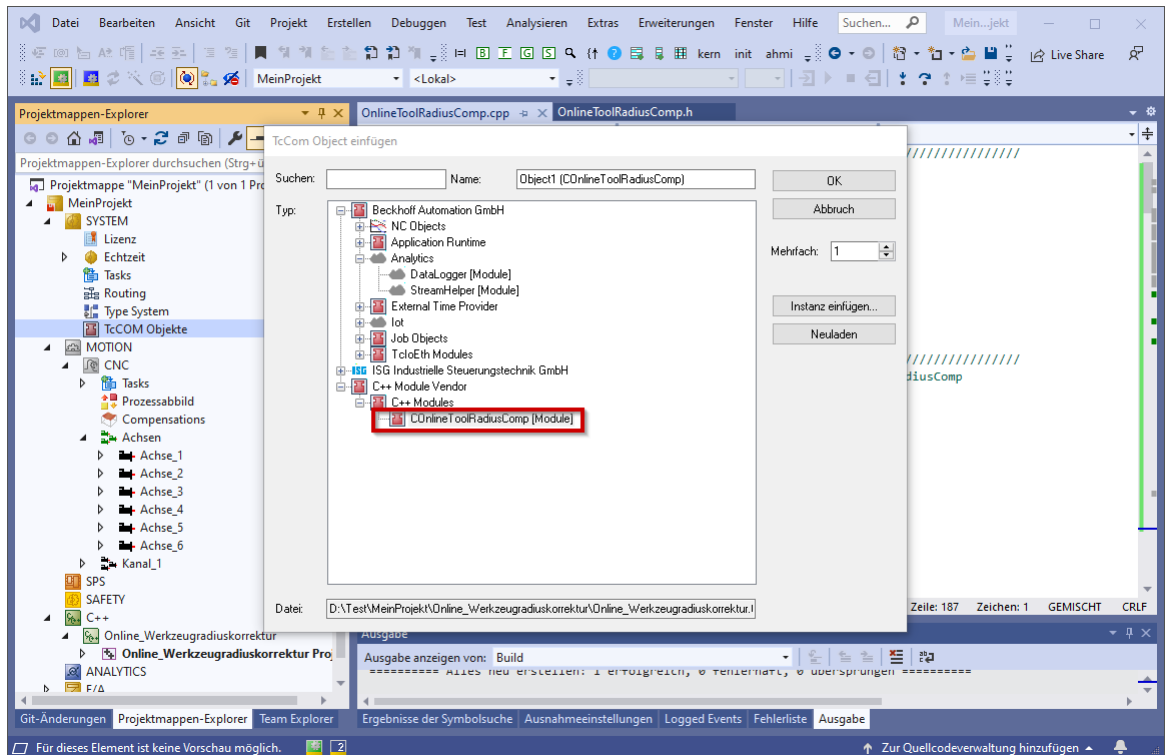


Abb. 39: Integration des TcCOM-Objekts

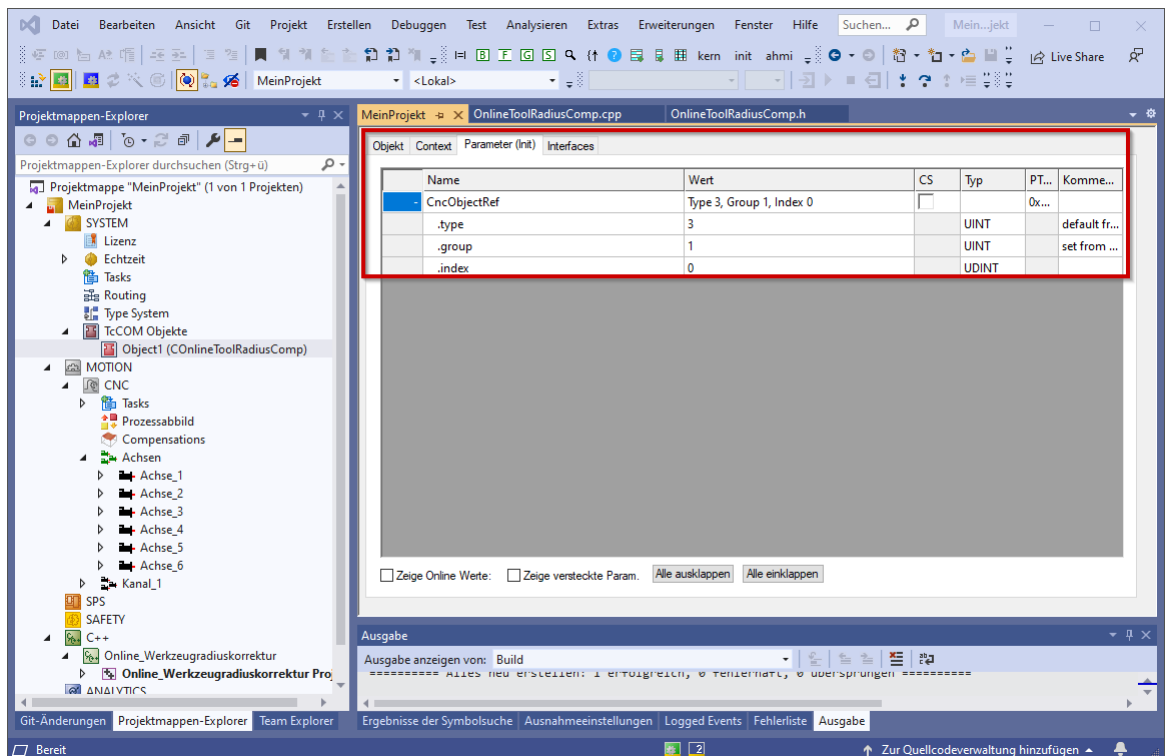


Abb. 40: Eigenschaften des TcCOM-Objekts

Der Typ für das Objekts der Online-Werkzeugradiuskompensation ist 3.

Die Angabe des Kanals ist abhängig davon, ob instanzspezifische Variablen im TcCOM-Objekt verwendet werden.

- Mit instanzspezifischen Daten entspricht die Gruppe der Kanalnummer [1;12].
- Ohne instanzspezifischen Daten ist die Gruppe mit 0 zu belegen.

Der Index des jeweiligen Objekts wird nicht verwendet.



Hinweis

Das TcCOM-Objekt muss digital signiert sein, ansonsten kann dieses nicht geladen werden. Weitere Informationen zu TcCOM-Objekten und wie ein Objekt digital signiert werden kann, finden Sie im [Beckhoff Information System](#).

Für jeden Kanal kann nur ein Objekt der Online-Werkzeugradiuskompensation verwendet werden.

6 Anhang

6.1 Anregungen, Korrekturen und neueste Dokumentation

Sie haben einen Fehler gefunden, 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

P

P-CHAN-00384	53
P-CHAN-00385	53
P-CHAN-00386	41
P-CHAN-00387	41
P-CHAN-00388	53
P-CHAN-00389	54
P-CHAN-00390	41
P-CHAN-00391	42
P-CHAN-00550	33



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

