SIEMENS

SINUMERIK

SINUMERIK MC
Werkzeuge

Funktionshandbuch

Vorwort

Grundlegende Sicherheitshinweise 1

W1: Werkzeugkorrektur 2

W4: Schleifspezifische Werkzeugkorrektur und -überwachung 3

Anhang A

Gültig für

Steuerung
SINUMERIK MC

CNC-Software Version 1.12

06/2019
A5E47438175A AA

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Digital Industries
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

Dokumentbestellnummer: A5E47438175A AA
℗ 05/2019 Änderungen vorbehalten

Copyright © Siemens AG 2019.
Alle Rechte vorbehalten

Vorwort

SINUMERIK-Dokumentation

Die SINUMERIK-Dokumentation ist in folgende Kategorien gegliedert:

- Allgemeine Dokumentation/Kataloge
- Anwender-Dokumentation
- Hersteller-/Service-Dokumentation

Weiterführende Informationen

Unter folgender Adresse (https://support.industry.siemens.com/cs/de/de/view/108464614) finden Sie Informationen zu den Themen:

- Dokumentation bestellen/Druckschriftenübersicht
- Weiterführende Links für den Download von Dokumenten
- Dokumentation online nutzen (Handbücher/Informationen finden und durchsuchen)

Bei Fragen zur technischen Dokumentation (z. B. Anregungen, Korrekturen) senden Sie eine E-Mail an folgende Adresse (mailto:docu.motioncontrol@siemens.com).

mySupport/Dokumentation

Unter folgender Adresse (https://support.industry.siemens.com/My/ww/de/documentation) finden Sie Informationen, wie Sie Ihre Dokumentation auf Basis der Siemensinhalte individuell zusammenstellen und für die eigene Maschinendokumentation anpassen.

Training

Unter folgender Adresse (http://www.siemens.de/sitrain) finden Sie Informationen zu SITRAIN - dem Training von Siemens für Produkte, Systeme und Lösungen der Antriebs- und Automatisierungstechnik.

FAQs

Frequently Asked Questions finden Sie in den Service&Support-Seiten unter Produkt Support (https://support.industry.siemens.com/cs/de/de/ps/faq).

SINUMERIK

Informationen zu SINUMERIK finden Sie unter folgender Adresse (http://www.siemens.de/sinumerik).

Zielgruppe

Die vorliegende Druckschrift wendet sich an:

- Projekteure
- Technologen (von Maschinenherstellern)
- Inbetriebnehmer (von Systemen/Maschinen)
- Programmierer

Nutzen

Das Funktionshandbuch beschreibt die Funktionen, so dass die Zielgruppe die Funktionen kennt und auswählen kann. Es befähigt die Zielgruppe, die Funktionen in Betrieb zu nehmen.

Standardumfang

In der vorliegenden Dokumentation ist die Funktionalität des Standardumfangs beschrieben. Ergänzungen oder Änderungen, die durch den Maschinenhersteller vorgenommen werden, werden vom Maschinenhersteller dokumentiert.

Es können in der Steuerung weitere, in dieser Dokumentation nicht erläuterte Funktionen ablauffähig sein. Es besteht jedoch kein Anspruch auf diese Funktionen bei der Neulieferung bzw. im Servicefall.

Ebenso enthält diese Dokumentation aus Gründen der Übersichtlichkeit nicht sämtliche Detailinformationen zu allen Typen des Produkts und kann auch nicht jeden denkbaren Fall der Aufstellung, des Betriebes und der Instandhaltung berücksichtigen.

Hinweis zur Datenschutzgrundverordnung

Siemens beachtet die Grundsätze des Datenschutzes, insbesondere die Gebote der Datenminimierung (privacy by design). Für dieses Produkt bedeutet dies:

Das Produkt verarbeitet/speichert keine personenbezogenen Daten, lediglich technische Funktionsdaten (z. B. Zeitstempel). Verknüpft der Anwender diese Daten mit anderen Daten (z. B. Schichtpläne) oder speichert er personenbezogene Daten auf dem gleichen Medium (z. B. Festplatte) und stellt so einen Personenbezug her, hat er die Einhaltung der datenschutzrechtlichen Vorgaben selbst sicherzustellen.

Technical Support

Landesspezifische Telefonnummern für technische Beratung finden Sie im Internet unter folgender Adresse (https://support.industry.siemens.com/sc/ww/de/sc/2090) im Bereich "Kontakt".

Informationen zu Struktur und Inhalt

Aufbau

Das vorliegende Funktionshandbuch ist wie folgt aufgebaut:

- Innentitel (Seite 3) mit dem Titel des Funktionshandbuchs, den SINUMERIK-Steuerungen sowie der Software und Version, für die diese Ausgabe des Funktionshandbuchs gültig ist, und der Übersicht der einzelnen Funktionsbeschreibungen.
- Funktionsbeschreibungen in der Reihenfolge ihrer alphanumerischen Kurzzeichen (z. B. A2, A3, B1 etc.)
- Anhang mit:
  - Abkürzungsverzeichnis
  - Dokumentationsübersicht
- Begriffsverzeichnis

Hinweis

Ausführliche Daten- und Alarm-Beschreibungen siehe:

- für Maschinen- und Settingdaten:
  Ausführliche Maschinendaten-Beschreibung
- für NC/PLC-Nahtstellensignale:
  Listenhandbuch NC-Variablen und Nahtstellensignale
- für Alarme:
  Diagnosehandbuch

Schreibweise von Systemdaten

Für Systemdaten gelten in dieser Dokumentation folgende Schreibweisen:

Signal/Datum	Schreibweise	Beispiel
NC/PLC-Nahtstellensignale	... NC/PLC-Nahtstellensignal: <Signaladresse> (<Signalname>)	Ist die neue Getriebestufe eingelegt, dann werden vom PLC-Programm die folgenden NC/PLC-Nahtstellensignale gesetzt: DB31, ... DBX16.0-2 (Istgetriebestufe A bis C) DB31, ... DBX16.3 (Getriebe ist umgeschaltet)
Maschinendatum	... Maschinendatum: <Typ><Nummer> <Vollständiger Bezeichner> (<Bedeutung>)	Masterspindel ist die abgelegte Spindel im Maschinendatum: MD20090 $MC_SPIND_DEF_MASTER_SPIND (Löschstellung Masterspindel im Kanal)
Settingdatum	... Settingdatum: <Typ><Nummer> <Vollständiger Bezeichner> (<Bedeutung>)	Die logische Masterspindel ist enthalten im Settingdatum: SD42800 $SC_SPIND_ASSIGN_TAB[0] (Spindelnummernumsetzer)

Mengengerüst

Erläuterungen bezüglich der NC/PLC-Nahtstelle gehen von der absoluten maximalen Anzahl folgender Komponenten aus:

- Betriebsartengruppen (DB11)
- Kanäle (DB21, ...)
- Achsen/Spindeln (DB31, ...)

Datentypen

In der Steuerung stehen zur Programmierung in Teileprogrammen folgenden Datentypen zur Verfügung:

Typ	Bedeutung	Wertebereich
INT	Ganzzahlige Werte mit Vorzeichen	-2.147.483.648 ... +2.147.483.647
REAL	Zahlen mit Dezimalpunkt	≈ ±5,0*10^{-324} … ≈ ±1,7*10^{+308}
BOOL	Wahrheitswerte	TRUE (≠0) , FALSE (0)
CHAR	Zeichen ASCII bzw. Byte	0 ... 255 bzw. -128 ... 127
STRING	Zeichenkette, Null-terminiert	maximal 400 Zeichen + /0 (keine Sonderzeichen)
AXIS	Achsnamen	alle in der Steuerung vorhandenen Achsnamen
FRAME	geometrische Angaben für Verschieben, Drehen, Skalieren, Spiegeln	---

Felder

Felder können nur aus gleichen elementaren Datentypen gebildet werden. Es sind maximal 3-dimensionale Arrays möglich.

Beispiel: `DEF INT FELD[2, 3, 4]`

Zahlensysteme

Es stehen folgende Zahlensystem zur Verfügung:

- Dezimal: `DEF INT Zahl = 1234` oder `DEF REAL Zahl = 1234.56`
- Hexadezimal: `DEF INT Zahl = 'H123ABC'`
- Binär: `DEF INT Zahl = 'B10001010010'`

Abfrage von REAL-Variablen

Es wird empfohlen, die Abfragen von REAL- bzw. DOUBLE-Variablen in NC-Programmen und Synchronaktionen als Grenzwertbetrachtung zu programmieren.

Beispiel: Abfrage des Istwerts einer Achse auf einen bestimmten Wert

```
Programmcode                                                                    Kommentar
DEF REAL AXPOS = 123.456
IF ($VA_IM[<Achse>] - 1ex-6) <= AXPOS <= ($VA_IM[<Achse>] + 1ex-6)              ; Istposition
   ...                                                                          == AXPOS
```

Programmcode	Kommentar
ELSE	
...	<> AXPOS
ENDIF	

Inhaltsverzeichnis

Vorwort 3

1 Grundlegende Sicherheitshinweise 15

1.1 Allgemeine Sicherheitshinweise 15

1.2 Gewährleistung und Haftung für Applikationsbeispiele 15

1.3 Industrial Security 16

2 W1: Werkzeugkorrektur 19

2.1 Einführung 19
2.1.1 Programmierte Kontur und Werkzeugbahn 19
2.1.2 Werkzeuglängenkorrektur 19
2.1.3 Werkzeugradiuskorrektur 21

2.2 Werkzeugwechsel und Aktivierung der Werkzeugkorrektur 21
2.2.1 Anwahl von Werkzeug und Schneide und Werkzeugwechsel 21
2.2.2 Aktivierung der Werkzeugkorrektur 24

2.3 Werkzeugkorrekturspeicher 24
2.3.1 Korrekturspeicherstruktur 24
2.3.2 Verrechnung der WZ-Korrektur 26
2.3.3 Adresserweiterung für die NC-Adressen T und M 26
2.3.4 Freie D-Nummernvergabe 28
2.3.5 Korrektursatz bei Fehler im Werkzeugwechsel 34
2.3.6 Definition der Wirkung der Werkzeugparameter 36

2.4 Werkzeugparameter 37
2.4.1 Übersicht 37
2.4.2 WZ-Parameter 1: Werkzeugtyp 39
2.4.3 WZ-Parameter 2: Schneidenlage 42
2.4.4 WZ-Parameter 3 ... 5: Geometrie - Werkzeuglängen 44
2.4.5 WZ-Parameter 6 ... 11: Geometrie - Werkzeugform 45
2.4.6 WZ-Parameter 12 ... 14: Verschleiß - Werkzeuglängen 46
2.4.7 WZ-Parameter 15 ... 20: Verschleiß - Werkzeugform 46
2.4.8 WZ-Parameter 21 ... 23: Basismaß / Adaptermaß 47
2.4.9 WZ-Parameter 24: Freiwinkel 50

2.5 2D-Werkzeugradiuskorrektur 51
2.5.1 Allgemeine Informationen 51
2.5.2 Anwahl der WRK (G41/G42) 52
2.5.3 An- und Abfahrverhalten (NORM/KONT/KONTC/KONTT) 53
2.5.4 Weiches An- und Abfahren 58
2.5.4.1 Funktion 58
2.5.4.2 Parameter 58
2.5.4.3 Geschwindigkeiten 65
2.5.4.4 Systemvariablen 67
2.5.4.5 Randbedingungen 67
2.5.4.6 Beispiele 68

2.5.5 Abwahl der WRK (G40) ... 71
2.5.6 Korrektur an Außenecken ... 71
2.5.7 Korrektur an Innenecken ... 75
2.5.8 Kollisionsüberwachung ("Flaschenhalserkennung") ... 77
2.5.8.1 Funktion ... 77
2.5.8.2 Parametrierung ... 78
2.5.8.3 Programmierung ... 78
2.5.8.4 Randbedingungen ... 79
2.5.8.5 Beispiel ... 80
2.5.9 Sätze mit veränderlichem Korrekturwert ... 81
2.5.10 Alarmverhalten ... 82
2.5.11 Schnittpunktverfahren für Polynome ... 83
2.5.12 G461/G462: Erweiterung An- / Abfahrstrategie ... 83
2.5.13 Werkzeuge mit relevanter Schneidenlage ... 87

2.6 2 1/2 D-Werkzeugradiuskorrektur ... 87

2.7 Werkzeugradiuskorrektur konstant halten ... 90

2.8 Orientierbare Werkzeugträger ... 94
2.8.1 Allgemeine Informationen ... 94
2.8.2 Kinematische Zusammenhänge und Maschinenaufbau ... 101
2.8.3 Werkzeugträger mit kinematischen Ketten ... 108
2.8.4 Maschinenvermessung für Werkzeugträger ... 117
2.8.4.1 Orientierbare Werkzeugträger nach Maschinenvermessung modifizieren (CORRTC) ... 117
2.8.5 Schrägbearbeitung mit 3 + 2 Achsen ... 121
2.8.6 Maschine mit drehbarem Werktisch ... 122
2.8.7 Vorgehen bei Benutzung der orientierbaren Werkzeugträger ... 125
2.8.8 Programmierung ... 130
2.8.9 Randbedingungen und Steuerungsverhalten für Orientierungen ... 131
2.8.10 Beispiele ... 134
2.8.10.1 Beispiel: Orientierbare Werkzeugträger ... 134
2.8.10.2 Beispiel Orientierbare Werkzeugträger mit drehbarem Tisch ... 134
2.8.10.3 Korrekturen einsatzort- und werkstückspezifisch einrechnen ... 137
2.8.10.4 Beispiel Orientierbare Werkzeugträger über kinematische Kette ... 140

2.9 Modifikation der Korrekturdaten bei drehbaren Werkzeugen ... 142
2.9.1 Einführung ... 142
2.9.2 Drehung von Drehwerkzeugen ... 144
2.9.2.1 Schneidenlage, Schnittrichtung und Winkel bei Drehwerkzeugen ... 144
2.9.2.2 Modifikationen bei der Drehung von Drehwerkzeugen ... 146
2.9.3 Drehung von Fräs- und Bohrwerkzeugen ... 150
2.9.3.1 Schneidenlage bei Fräs- und Bohrwerkzeugen ... 150
2.9.3.2 Modifikationen bei der Drehung von Fräs- und Bohrwerkzeugen ... 151
2.9.4 Inbetriebnahme ... 151
2.9.4.1 Parametrierung ... 151
2.9.5 Programmierung ... 154
2.9.5.1 Orientierungen berechnen (ORISOLH) ... 154
2.9.5.2 Modifikation der Korrekturdaten bei drehbaren Werkzeugen aktivieren (CUTMOD, CUTMODK) ... 165
2.9.6 Beispiel ... 171

2.10 Inkrementell programmierte Korrekturwerte ... 173
2.10.1 G91 Erweiterung ... 173
2.10.2 Verfahren in Richtung der Werkzeugorientierung (MOVT) ... 174

2.11 Zuordnung der Werkzeuglängenkomponenten zu den Geometrieachsen 176
2.11.1 Zuordnung gemäß Werkzeugtyp und Arbeitsebene 176
2.11.2 Zuordnung bei Ebenenwechsel 176
2.11.3 Zuordnung unabhängig vom Werkzeugtyp 177

2.12 Achsparallele Werkzeugorientierung 178
2.12.1 Werkzeuggrundorientierung 178
2.12.2 Werkzeugorientierung bei Ebenenwechsel 178

2.13 Parametrierbare Werkzeuggrundorientierung 180
2.13.1 Funktion 180
2.13.2 Inbetriebnahme 181
2.13.2.1 Aktivierung 181
2.13.2.2 Parametrierung 181
2.13.3 Programmierung 182
2.13.4 Beispiele 184

2.14 Werkzeugkorrektur-Sonderbehandlungen 185
2.14.1 Relevante Settingdaten 185
2.14.2 Werkzeuglängen spiegeln 186
2.14.3 Verschleißlängen spiegeln 187
2.14.4 Werkzeuglängen im WKS unter Berücksichtigung der Orientierung 189
2.14.5 Werkzeuglängenoffsets in Werkzeugrichtung 189
2.14.6 Besonderheiten bei orientierbaren Werkzeugträgern 194

2.15 Summen- und Einrichtekorrekturen 194
2.15.1 Allgemeine Informationen 194
2.15.2 Funktionale Beschreibung 195
2.15.3 Aktivierung 199
2.15.4 Beispiele 204
2.15.5 Erweiterungen der Werkzeuglängenbestimmung 205
2.15.5.1 Korrekturen einsatzort- und werkstückspezifisch einrechnen 205
2.15.5.2 Funktionalität der einzelnen Verschleißwerte 209

2.16 Mit Werkzeugumgebungen arbeiten 213
2.16.1 Werkzeugumgebung speichern (TOOLENV) 213
2.16.2 Werkzeugumgebung löschen (DELTOOLENV) 216
2.16.3 T-, D- und DL-Nummer lesen (GETTENV) 217
2.16.4 Informationen zu gespeicherten Werkzeugumgebungen lesen ($P_TOOLENVN, ($P_TOOLENV) 218
2.16.5 Werkzeuglängen bzw. -längenkomponenten lesen (GETTCOR) 218
2.16.6 Werkzeugkomponenten ändern (SETTCOR) 225

2.17 Zuordnung der Werkzeuglängen L1, L2, L3 zu den Koordinatenachsen lesen (LENTOAX) 237

2.18 Randbedingungen 240
2.18.1 SD42935 Erweiterungen 240
2.18.2 Ankratzen 240

2.19 Datenlisten 241
2.19.1 Maschinendaten 241
2.19.1.1 NC-spezifische Maschinendaten 241
2.19.1.2 Kanal-spezifische Maschinendaten 241
2.19.1.3 Achs-/Spindel-spezifische Maschinendaten 242
2.19.2 Settingdaten 243

2.19.2.1 Kanal-spezifische Settingdaten 243
2.19.2.2 Kanal-spezifische Settingdaten 243

3 W4: Schleifspezifische Werkzeugkorrektur und -überwachung 245

3.1 Schleifspezifische Werkzeugdaten 245
3.1.1 Struktur der Werkzeugdaten 245
3.1.2 Schneidenspezifische Parameter 246
3.1.2.1 Liste der schneidenspezifischen Parameter 246
3.1.2.2 $TC_DP1 248
3.1.2.3 Definition zusätzlicher Parameter $TC_DPC1 ... 10 248
3.1.3 Werkzeugspezifische Parameter 249
3.1.3.1 Liste der werkzeugspezifischen Parameter 249
3.1.3.2 $TC_TPG1 250
3.1.3.3 $TC_TPG2 250
3.1.3.4 $TC_TPG3, $TC_TPG4 252
3.1.3.5 $TC_TPG5 252
3.1.3.6 $TC_TPG6 und $TC_TPG7 252
3.1.3.7 $TC_TPG8 252
3.1.3.8 $TC_TPG9 253
3.1.3.9 $TC_TPG_DRSPATH und $TC_TPG_DRSPROG 254
3.1.3.10 Definition zusätzlicher Parameter $TC_TPC1 ... 10 254
3.1.3.11 Zugriff auf werkzeugspezifische Parameter 254
3.1.4 Ebenen und Achszuordnungen 255
3.1.5 Beispiele 256

3.2 Online-Werkzeugkorrektur 258
3.2.1 Funktion 258
3.2.2 Inbetriebnahme 260
3.2.3 Programmierung 261
3.2.3.1 Polynom-Funktion definieren (FCTDEF) 261
3.2.3.2 Online-Werkzeugkorrektur schreiben, kontinuierlich (PUTFTOCF) 262
3.2.3.3 Online-Werkzeugkorrektur schreiben, diskret (PUTFTOC) 263
3.2.3.4 Online-Werkzeugkorrektur ein-/ausschalten (FTOCON/FTOCOF) 264
3.2.4 Randbedingungen 264
3.2.5 Beispiele 266
3.2.5.1 Beispiel: Online-Werkzeugkorrektur schreiben, kontinuierlich 266

3.3 Online-Werkzeugradiuskorrektur 268

3.4 Schleifspezifische Werkzeugüberwachung 269
3.4.1 Funktion 269
3.4.2 Inbetriebnahme 271
3.4.3 Programmierung 271
3.4.3.1 Schleifspezifische Werkzeugüberwachung ein-/ausschalten (TMON, TMOF) 271

3.5 Konstante Scheibenumfangsgeschwindigkeit (SUG) 272
3.5.1 Funktion 272
3.5.2 Inbetriebnahme 273
3.5.3 Programmierung 274
3.5.3.1 Konstante Scheibenumfangsgeschwindigkeit ein-/ausschalten (GWPSON, GWPSOF) 274
3.5.4 Beispiel 275

3.6 Datenlisten 276
3.6.1 Maschinendaten 276
3.6.1.1 Allgemeine Maschinendaten 276

3.6.1.2 Kanal-spezifische Maschinendaten....276
3.6.1.3 Achs-/Spindel-spezifische Maschinendaten....277
A Anhang....279
A.1 Liste der Abkürzungen....279
Index....289

Grundlegende Sicherheitshinweise 1

1.1 Allgemeine Sicherheitshinweise

⚠ WARNUNG

Lebensgefahr bei Nichtbeachtung von Sicherheitshinweisen und Restrisiken

Bei Nichtbeachtung der Sicherheitshinweise und Restrisiken in der zugehörigen Hardware-Dokumentation können Unfälle mit schweren Verletzungen oder Tod auftreten.

- Halten Sie die Sicherheitshinweise der Hardware-Dokumentation ein.
- Berücksichtigen Sie bei der Risikobeurteilung die Restrisiken.

⚠ WARNUNG

Fehlfunktionen der Maschine infolge fehlerhafter oder veränderter Parametrierung

Durch fehlerhafte oder veränderte Parametrierung können Fehlfunktionen an Maschinen auftreten, die zu Körperverletzungen oder Tod führen können.

- Schützen Sie die Parametrierung vor unbefugtem Zugriff.
- Beherrschen Sie mögliche Fehlfunktionen durch geeignete Maßnahmen, z. B. NOT-HALT oder NOT-AUS.

1.2 Gewährleistung und Haftung für Applikationsbeispiele

Applikationsbeispiele sind unverbindlich und erheben keinen Anspruch auf Vollständigkeit hinsichtlich Konfiguration und Ausstattung sowie jeglicher Eventualitäten. Applikationsbeispiele stellen keine kundenspezifischen Lösungen dar, sondern sollen lediglich Hilfestellung bieten bei typischen Aufgabenstellungen.

Als Anwender sind Sie für den sachgemäßen Betrieb der beschriebenen Produkte selbst verantwortlich. Applikationsbeispiele entheben Sie nicht der Verpflichtung zu sicherem Umgang bei Anwendung, Installation, Betrieb und Wartung.

1.3 Industrial Security

Hinweis

Industrial Security

Siemens bietet Produkte und Lösungen mit Industrial Security-Funktionen an, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen.

Um Anlagen, Systeme, Maschinen und Netzwerke gegen Cyber-Bedrohungen zu sichern, ist es erforderlich, ein ganzheitliches Industrial Security-Konzept zu implementieren (und kontinuierlich aufrechtzuerhalten), das dem aktuellen Stand der Technik entspricht. Die Produkte und Lösungen von Siemens formen einen Bestandteil eines solchen Konzepts.

Die Kunden sind dafür verantwortlich, unbefugten Zugriff auf ihre Anlagen, Systeme, Maschinen und Netzwerke zu verhindern. Diese Systeme, Maschinen und Komponenten sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn und soweit dies notwendig ist und nur wenn entsprechende Schutzmaßnahmen (z. B. Nutzung von Firewalls und/oder Netzwerksegmentierung) ergriffen wurden.

Weiterführende Informationen zu möglichen Schutzmaßnahmen im Bereich Industrial Security finden Sie unter:

Industrial Security (https://www.siemens.com/industrialsecurity)

Die Produkte und Lösungen von Siemens werden ständig weiterentwickelt, um sie noch sicherer zu machen. Siemens empfiehlt ausdrücklich, Produkt-Updates anzuwenden, sobald sie zur Verfügung stehen und immer nur die aktuellen Produktversionen zu verwenden. Die Verwendung veralteter oder nicht mehr unterstützter Versionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Produkt-Updates informiert zu sein, abonnieren Sie den Siemens Industrial Security RSS Feed unter:

Industrial Security (https://www.siemens.com/industrialsecurity)

Weitere Informationen finden Sie im Internet:

Projektierungshandbuch Industrial Security (https://support.industry.siemens.com/cs/ww/de/view/108862708)

⚠ WARNUNG

Unsichere Betriebszustände durch Manipulation der Software

Manipulationen der Software, z. B. Viren, Trojaner oder Würmer, können unsichere Betriebszustände in Ihrer Anlage verursachen, die zu Tod, schwerer Körperverletzung und zu Sachschäden führen können.

- Halten Sie die Software aktuell.
- Integrieren Sie die Automatisierungs- und Antriebskomponenten in ein ganzheitliches Industrial Security-Konzept der Anlage oder Maschine nach dem aktuellen Stand der Technik.
- Berücksichtigen Sie bei Ihrem ganzheitlichen Industrial Security-Konzept alle eingesetzten Produkte.
- Schützen Sie die Dateien in Wechselspeichermedien vor Schadsoftware durch entsprechende Schutzmaßnahmen, z. B. Virenscanner.
- Prüfen Sie beim Abschluss der Inbetriebnahme alle security-relevanten Einstellungen.
- Schützen Sie den Antrieb vor unberechtigten Änderungen, indem Sie die Umrichterfunktion "Know-How-Schutz" aktivieren.

W1: Werkzeugkorrektur

2

2.1 Einführung

2.1.1 Programmierte Kontur und Werkzeugbahn

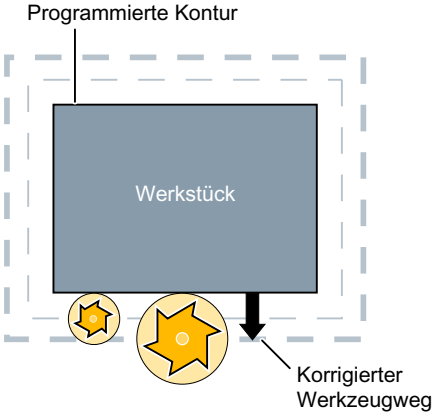
Werkstückmaße werden direkt programmiert (z. B. nach Fertigungszeichnung). Werkzeugdaten wie Fräserdurchmesser, Schneidenlage der Drehmeißel (linker / rechter Drehmeißel) und Werkzeuglängen müssen daher bei der Programmerstellung nicht berücksichtigt werden.

Die Steuerung korrigiert den Verfahrweg

Bei der Fertigung eines Werkstücks werden die Werkzeugwege abhängig von der jeweiligen Werkzeuggeometrie so gesteuert, dass mit jedem eingesetzten Werkzeug die programmierte Kontur hergestellt werden kann.

Damit die Steuerung die Werkzeugwege berechnen kann, müssen die Werkzeugdaten im Werkzeugkorrekturspeicher der Steuerung eingetragen sein. Über das NC-Programm werden lediglich das benötigte Werkzeug (T...) und der benötigte Korrekturdatensatz (D...) aufgerufen.

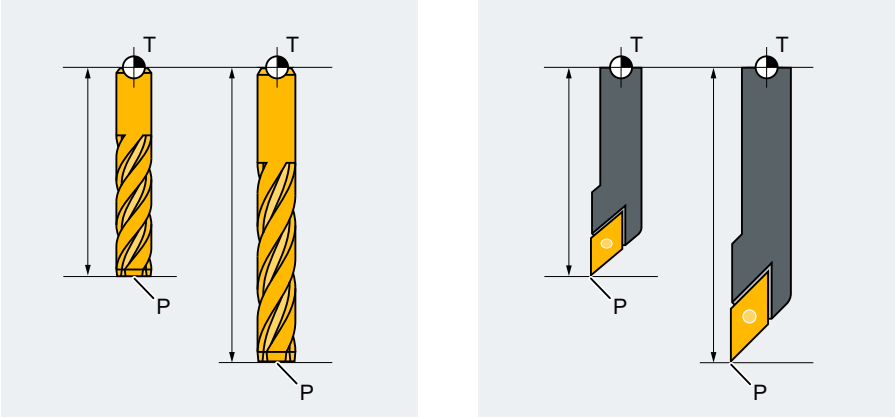
Die Steuerung holt sich während der Programmverarbeitung die benötigten Korrekturdaten aus dem Werkzeugkorrekturspeicher und korrigiert für unterschiedliche Werkzeuge individuell die Werkzeugbahn:



2.1.2 Werkzeuglängenkorrektur

Mit der Werkzeuglängenkorrektur werden die Längenunterschiede zwischen den eingesetzten Werkzeugen ausgeglichen.

Als Werkzeuglänge gilt der Abstand zwischen Werkzeugträgerbezugspunkt und Werkzeugspitze:



T	Werkzeugträgerbezugspunkt
P	Werkzeugspitze

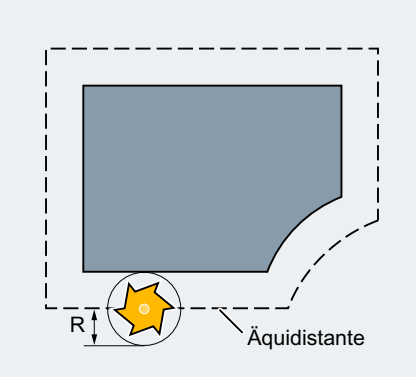
Diese Länge wird vermessen und zusammen mit vorgebbaren Verschleißwerten in den Werkzeug-Korrekturspeicher der Steuerung eingegeben. Hieraus errechnet die Steuerung die Verfahrbewegungen in Zustellrichtung.

Hinweis

Der Korrekturwert der Werkzeuglänge ist abhängig von der räumlichen Orientierung des Werkzeugs.

2.1.3 Werkzeugradiuskorrektur

Kontur und Werkzeugbahn sind nicht identisch. Der Fräser- bzw. Schneidenmittelpunkt muss entsprechend des Werkzeugradius auf einer Äquidistanten zur Kontur (Werkzeugmittelpunktsbahn) verfahren werden. Dazu wird von der Steuerung während der Abarbeitung des Programms die programmierte Werkzeugmittelpunktsbahn anhand des Werkzeugradius des aktiven Werkzeugs (Werkzeug-Korrekturspeicher) so verschoben, dass die Werkzeugschneide exakt an der programmierten Kontur verfahren wird.





R Werkzeugradius
S Schneidenmittelpunkt

Die Werkzeugradiuskorrektur ist ausführlich beschrieben in Kapitel "2D-Werkzeugradiuskorrektur (Seite 51)".

2.2 Werkzeugwechsel und Aktivierung der Werkzeugkorrektur

2.2.1 Anwahl von Werkzeug und Schneide und Werkzeugwechsel

Werkzeuganwahl

Ein Werkzeug wird im NC-Programm mit der T-Funktion angewählt.

Die T-Funktion bzw. die Werkzeugnummer kann folgende ganzzahlige Werte annehmen:

- Minimalwert: T0 (kein Werkzeug)
- Maximalwert: T32000 (Werkzeug mit der Nummer 32000)

Ob mit der T-Funktion sofort das neue Werkzeug eingewechselt wird, hängt ab von der Einstellung im Maschinendatum:

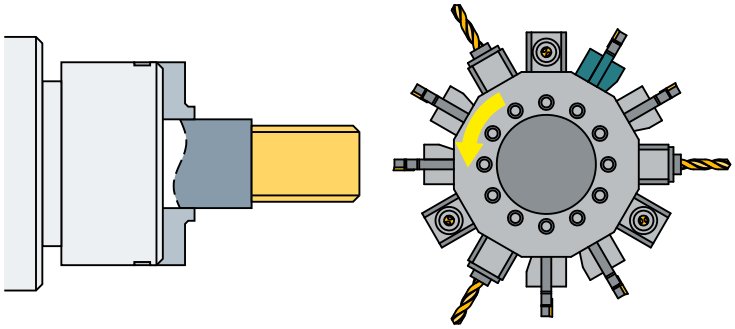
MD22550 $MC_TOOL_CHANGE_MODE (Neue Werkzeugkorrektur bei M-Funktion)

Werkzeugwechsel sofort

MD22550 $MC_TOOL_CHANGE_MODE = 0

Das neue Werkzeug wird mit der T-Funktion sofort eingewechselt.

Diese Einstellung wird hauptsächlich bei Drehmaschinen mit Revolvermagazinen verwendet.



Werkzeugwechsel mit M-Funktion

MD22550 $MC_TOOL_CHANGE_MODE = 1

Das neue Werkzeug wird mit der T-Funktion zum Wechsel vorbereitet.

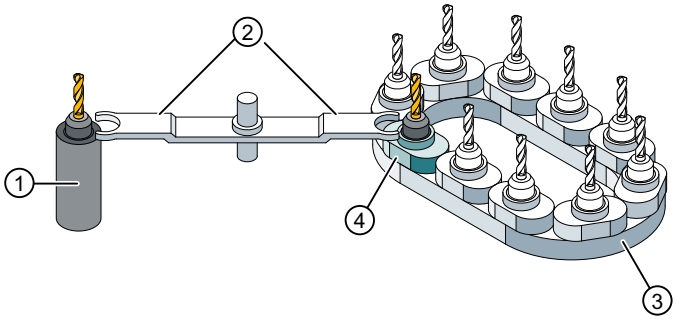
Diese Einstellung wird hauptsächlich bei Fräsmaschinen mit Werkzeugmagazin verwendet, um das neue Werkzeug hauptzeitparallel (die Bearbeitung wird nicht unterbrochen) auf die Werkzeugwechselposition zu bringen.

Mit der M-Funktion für den Werkzeugwechsel wird das alte Werkzeug aus der Spindel entfernt und das neue Werkzeug in die Spindel eingewechselt.

Die M-Funktion für den Werkzeugwechsel wird festgelegt im Maschinendatum:

MD22560 $MC_TOOL_CHANGE_M_CODE (M-Funktion für Werkzeugwechsel)

Standardeinstellung: M06 (gemäß DIN 66025)



① Spindel
② Greifer
③ Magazin (hier: Kettenmagazin)
④ Werkzeugwechselposition

Vorgewähltes Werkzeug

Das Werkzeug, dessen Werkzeuglängenkorrekturwerte bei Steuerungshochlauf und bei Reset bzw. Teileprogrammende in Abhängigkeit vom MD20110 $MC_RESET_MODE_MASK und bei Teileprogrammstart in Abhängigkeit vom MD20112 $MC_START_MODE_MASK berücksichtigt werden sollen, wird festgelegt mit dem Maschinendatum:

MD20121 $MC_TOOL_PRESEL_RESET_VALUE

Werkzeugschneide D

Einem Werkzeug T<x> können mehrere Werkzeugschneiden D1 ... D<n> zugeordnet werden.

Maximale Anzahl Werkzeugschneiden pro Werkzeug

Die maximale Anzahl von Werkzeugschneiden (D-Nummern), die pro Werkzeug definiert werden können, wird über folgendes Maschinendatum festgelegt:

MD18106 $MN_MM_MAX_CUTTING_EDGE_PERTOOL

D-Funktion

Die Anwahl der zur Bearbeitung gewünschten Werkzeugschneide 1 ... <n> erfolgt im NC-Programm durch Programmierung der entsprechenden D-Funktion D1 ... D<n>.

Die angewählte Werkzeugschneide bezieht sich immer auf das aktive Werkzeug. Eine Werkzeugschneide ohne aktives Werkzeug (aktuelle T-Nummer ist T0 ⇒ kein Werkzeug angewählt) ist unwirksam.

Mit D0 werden alle Werkzeugkorrekturen des aktiven Werkzeugs T<x> abgewählt.

Anwahl der Schneide bei Werkzeugwechsel

Nach Anwahl eines neuen Werkzeugs und dem Einwechseln dieses Werkzeugs gibt es folgende Möglichkeiten zur Anwahl der Schneide:

- Die Schneidennummer D<x> wird programmiert.
- Die Schneidennummer wird vorgegeben durch das Maschinendatum: MD20270 $MC_CUTTING_EDGE_DEFAULT (Grundstellung der Werkzeugschneide ohne Programmierung)

Wert	Bedeutung
0	Nach M06 erfolgt keine automatische Schneidenanwahl.
≠ 0	Nummer der Schneide, die nach M06 angewählt ist.
-1	Die Schneidennummer des alten Werkzeugs bleibt erhalten und wird nach M06 auch für das neue Werkzeug angewählt.
-2	Die Schneidenkorrektur des alten Werkzeugs bleibt weiterhin aktiv, bis mit D eine neue Schneide programmiert wird.

2.2.2 Aktivierung der Werkzeugkorrektur

Mit D1 bis D<n> wird die Werkzeugkorrektur einer Werkzeugschneide für das aktive Werkzeug aktiviert. Die Werkzeuglängenkorrektur und die Werkzeugradiuskorrektur werden jedoch zu unterschiedlichen Zeitpunkten wirksam:

- Die **Werkzeuglängenkorrektur** (WLK) wird mit der ersten Verfahrbewegung der Achse, in der die WLK wirken soll, herausgefahren.
  Diese Verfahrbewegung muss eine Linearinterpolation (G0, G1, POS, POSA) oder Polynominterpolation (POLY) sein. Ist die POS/POSA-Achse eine der aktiven Geometrieachsen, wird die Werkzeuglängenkorrektur mit der ersten Verfahrbewegung der Achse, in der die WLK wirken soll, herausgefahren.
- Die **Werkzeugradiuskorrektur** (WRK) wird durch Programmierung von G41/G42 in der aktiven Ebene (G17, G18 oder G19) wirksam.
  Im NC-Satz, in dem die Werkzeugradiuskorrektur mit G41/G42 aktiviert wird, muss G0 (Eilgang) oder G1 (Linearinterpolation) aktiv sein und mindestens eine Geometrieachse der aktiven Arbeitsebene programmiert werden.
  Wird beim Einschalten nur eine Geometrieachse angegeben, wird die letzte Position der zweiten Geometrieachse automatisch ergänzt und anschließend in beiden Achsen verfahren.
  Sind die Achsen aktuell keine Geometrieachsen, müssen sie durch GEOAX-Programmierung zu Geometrieachsen gemacht werden.
  Die Anwahl der WRK mit G41/G42 darf nur in einem Programmsatz mit G0 (Eilgang) oder G1 (Linearinterpolation) erfolgen.

2.3 Werkzeugkorrekturspeicher

2.3.1 Korrekturspeicherstruktur

WZK-Speichergröße

Jeder Kanal kann einen eigenen WZK-Speicher (TO-Einheit) haben.

Welcher WZK-Speicher für den entsprechenden Kanal existiert, wird eingestellt mit dem Maschinendatum:

MD28085 $MC_MM_LINK_TOA_UNIT (Zuordnung der TO-Einheit zu einem Kanal)

Die maximale Anzahl von WZ-Schneiden für alle von der NC verwalteten Werkzeuge wird eingestellt mit dem Maschinendatum:

MD18100 $MN_MM_NUM_CUTTING_EDGES_IN_TOA (Anzahl der Werkzeugschneiden in der NC)

Werkzeuge

Der WZK-Speicher besteht aus Werkzeugen mit den Nummern `T1` bis `T32000`.

Jedes Werkzeug kann über TOA-Dateien oder einzeln mit dem Softkey "Neues Werkzeug" eingerichtet werden. Nicht benötigte Korrekturen sind mit dem Wert Null zu belegen. Dies ist

auch die Voreinstellung beim Anlegen der Korrekturspeicher. Die einzelnen Werte der Korrekturspeicher (WZ-Parameter) sind über Systemvariablen vom Programm les- und schreibbar.

Hinweis

Die Werkzeuge (`T1` bis `T32000`) müssen nicht in aufsteigender Reihenfolge, nicht lückenlos und nicht mit `T1` beginnend in den WZK-Speicher eingebracht werden.

WZ-Schneiden

Jedes Werkzeug kann bis zu 12 WZ-Schneiden (`D1` bis `D12`) besitzen. Wird ein neues Werkzeug in den WZK-Speicher eingebracht, wird automatisch die erste WZ-Schneide (`D1`) eingerichtet. Weitere WZ-Schneiden (max. 11) werden mit dem Softkey "Neue Schneide" in aufsteigender Reihenfolge lückenlos eingerichtet. Auf diese Weise kann jedem Werkzeug eine unterschiedliche Anzahl von WZ-Schneiden zugeordnet werden.

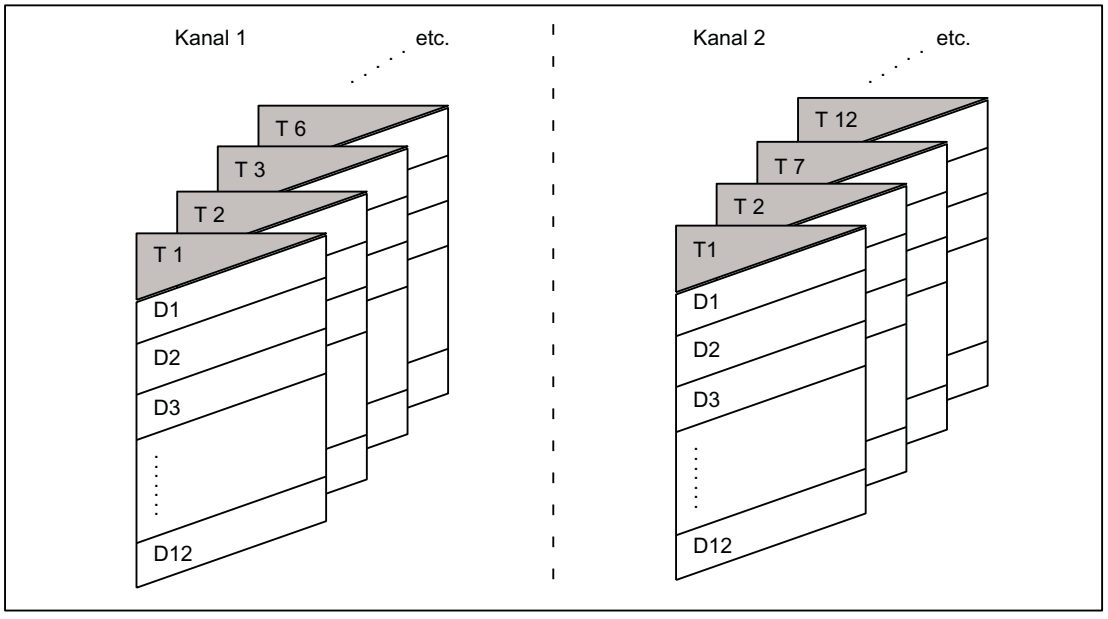
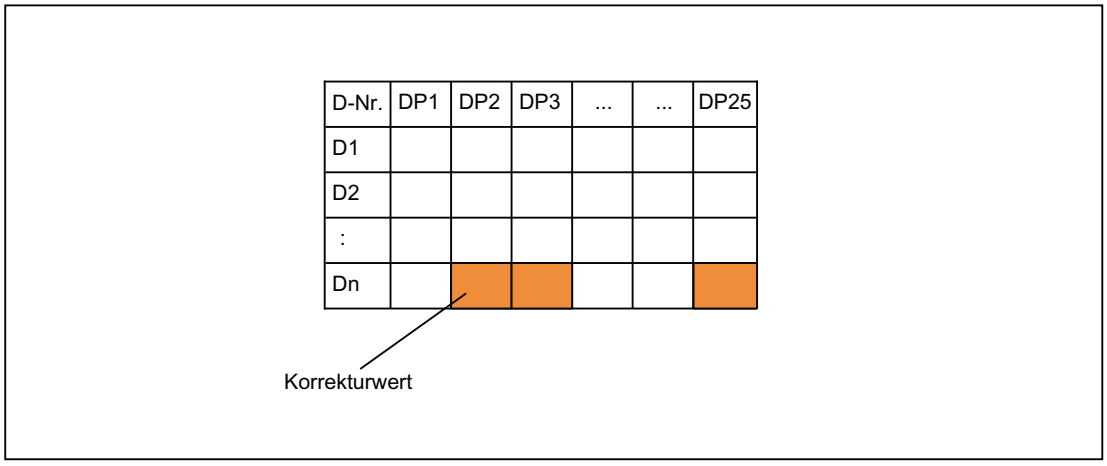


Bild 2-1 Beispiel einer WZK-Speicherstruktur für 2 Kanäle

2.3.2 Verrechnung der WZ-Korrektur

D-Nr.

Die **D-Nr.** reicht zur Ermittlung der WZ-Korrekturen aus (über MD einstellbar).



Programmierung

Obiger Korrekturblock soll in der NC verrechnet werden.

Aufruf im Teileprogramm:

```
...
Dn
```

2.3.3 Adresserweiterung für die NC-Adressen T und M

MD20096

Ob auch bei **nicht** aktivierter Werkzeugverwaltung die Adresserweiterung von T und M als Spindelnummer interpretiert werden soll, kann festgelegt werden über das Maschinendatum:

MD20096 $MC_T_M_ADDRESS_EXT_IS_SPINO (Spindelnummer als Adresserweiterung)

Für den Bezug der D-Nummer zur T-Nummer gelten dann dieselben Regeln wie bei aktiver Funktion "Werkzeugverwaltung".

Auswirkung auf die D-Nummer

Über die D-Nummer wird ein Korrekturdatensatz bestimmt.

Die D-Adresse kann nicht mit Adresserweiterung programmiert werden.

Die Auswertung der D-Adresse bezieht sich immer auf das momentan aktive Werkzeug.

Die programmierte D-Adresse bezieht sich auf das bzgl. der Masterspindel aktive Werkzeug (analog zur Funktion Werkzeugverwaltung) bei gesetztem Maschinendatum:

MD20096 $MC_T_M_ADDRESS_EXT_IS_SPINO = TRUE (Spindelnummer als Adresserweiterung)

Auswirkung auf die T-Nummer

Bei aktiver Funktion "Werkzeugverwaltung" werden die bzgl. der Masterspindel (bzw. Master-WZ-Halter) programmierten Werte als programmierte/aktive T-Nummern angezeigt.

Ohne aktive Werkzeugverwaltung werden **alle** programmierten T-Werte als programmiert/aktiv angezeigt (unabhängig davon, welche Adresserweiterung dazu programmiert wurde).

Nur noch der bzgl. der Masterspindel programmierte T-Wert wird als programmiert/aktiv angezeigt, wenn:

MD20096 $MC_T_M_ADDRESS_EXT_IS_SPINO = TRUE (Spindelnummer als Adresserweiterung)

Beispiel

Im folgenden Beispiel wird die Auswirkung von MD20096 gezeigt.

Es werden zwei Spindeln betrachtet. Die Spindel 1 ist Masterspindel. `M6` wurde als WZ-Wechselsignal festgelegt.

T1 = 5	
M1= 6	
T2 = 50	
M2 = 6	
D4	

- Bei aktiver Werkzeugverwaltung bezieht sich `D4` auf das Werkzeug "5".
  T2=50 bestimmt das Werkzeug für die Nebenspindel, deren Werkzeug nicht die Korrektur der Bahn beeinflusst. Die Bahn wird ausschließlich durch das für die Masterspindel programmierte Werkzeug bestimmt.
- `D4` bezieht sich auf das Werkzeug "50" ohne aktive Werkzeugverwaltung und mit dem Maschinendatum:
  MD20096 $MC_T_M_ADDRESS_EXT_IS_SPINO = **FALSE** (Bedeutung der Adresserweiterung bei T, M Werkzeugwechsel)
  Weder die Adresserweiterung von T noch von M werden im NC ausgewertet.
  Jeder WZ-Wechselbefehl bestimmt eine neue Korrektur der Bahn.
- `D4` bezieht sich auf das Werkzeug "5" (wie bei aktiver WZV) ohne aktive Werkzeugverwaltung und mit dem Maschinendatum:
  MD20096 $MC_T_M_ADDRESS_EXT_IS_SPINO = **TRUE**.
  Die Adresserweiterung 1 (`T1=...`, `M1=...`) bezeichnet die Masterspindel.

Hinweis

Bisher führt bei nicht aktivierter Werkzeugverwaltung jede Programmierung eines Werkzeugwechsels (mit T bzw. M) zu einer Neueinrechnung der WZ-Korrektur in die Bahn. Die Adresserweiterung wird dabei nicht näher definiert. Die Bedeutung der Erweiterung wird vom Anwender festgelegt (über PLC-Anwenderprogramm).

2.3.4 Freie D-Nummernvergabe

"Relative" D-Nummern

IN der NC besteht die Möglichkeit, die D-Nummern für die WZ-Korrekturdatensätze als "relative" D-Nummern zu führen. Dabei werden jeder T-Nummer die entsprechenden D-Nummern zugeordnet.

Funktionen

Erweiterungen der Funktionen bei der Vergabe der D-Nummern:

- Die maximal erlaubte D-Nummer wird festgelegt über das Maschinendatum:
  MD18105 $MN_MM_MAX_CUTTING_EDGE_NO (Max. Wert der D-Nummern (DRAM))
  Es ist der Wert 9 voreingestellt, so dass bestehende Anwendungen weiterhin lauffähig bleiben.
- Die Anzahl der Schneiden (bzw. Korrekturdatensätze) **je Werkzeug** kann festgelegt werden über das Maschinendatum:
  MD18106 $MN_MM_MAX_CUTTING_EDGE_PERTOOL (Max. Anzahl der D-Nummern pro Werkzeug (DRAM))
  Damit kann die Anzahl der zu parametrierenden Schneiden pro Werkzeug an die Anzahl der real möglichen Schneiden angepasst und so überwacht werden.
- Es gibt außerdem die Möglichkeit, D-Nummern im NC umzubenennen und damit beliebige D-Nummern für die Schneiden zu vergeben.

Schneidennummer CE

Beim Umbenennen der D-Nummern geht die Information über die im Werkzeugkatalog für diese Schneiden festgelegte Nummer verloren. Es kann somit nach dem Umbenennen nicht mehr erkannt werden, um welche Schneide des Kataloges es sich handelt.

Da diese Information bei Umrüstvorgängen benötigt wird, wurde für jede Schneide eine **Schneidennummer CE** eingeführt, die beim Umbenennen der D-Nummer erhalten bleibt.

Die D-Nummer bezeichnet im Teileprogramm die Schneidenkorrektur. Diese **Korrekturnummer D** wird getrennt von der **Schneidennummer CE** (die Nummer im WZ-Katalog) verwaltet und ist eine beliebige Zahl, mit der eine Korrektur im Teileprogramm und auf der Anzeige benannt wird.

Die CE-Nummer bezeichnet beim Umrüsten die konkrete physikalische Schneide. Die Schneidennummer CE wird vom NC bei der Korrekturanwahl beim Werkzeugwechsel nicht ausgewertet (nur über BTSS verfügbar).

Die Schneidennummer CE wird über die Systemvariable **$TC_DPCE[t,d]** beschrieben:

- **t** steht für die interne T-Nummer
- **d** steht für die D-Nummer.

Schreibvorgänge werden auf Kollision überwacht; d.h. die Schneidennummern eines Werkzeugs müssen alle verschieden sein. Die Variable $TC_DPCE ist Bestandteil des Schneidenparameterdatensatzes $TC_DP1,..., $TC_DP25.

Die Parametrierung von $TC_DPCE ist nur sinnvoll, wenn die maximale Schneidennummer (MD18105) größer ist als die maximale Anzahl der Schneiden je Werkzeug (MD18106).

Dann ist die Vorbelegung der Schneidennummer gleich der Ordnungsnummer der Schneide. Die angelegten Korrekturen eines Werkzeugs werden mit 1 beginnend bis zur maximalen Anzahl der Schneiden je Werkzeug hochgezählt (MD18106).

Die Schneidennummer CE ist gleich der D-Nummer (in Kompatibilität zum bisherigen Verhalten) falls:

MD18105 ≤ MD18106.

Eine Leseoperation liefert CE=D. Eine Schreiboperation wird ohne Alarmmeldung ignoriert.

Hinweis

Die Korrekturwerte $TC_DP1,..., $TC_DP25 der aktiven WZ-Korrektur können mit der Systemvariablen $P_AD[n]; mit n=1,...,25 gelesen werden. Die Schneidennummer CE der aktiven Korrektur erhält man für n=26.

Befehle

Ist die maximale Schneidennummer **größer** als die maximale Anzahl der Schneiden pro Werkzeug, stehen die in der folgenden Tabelle aufgeführten Befehle zur Verfügung

- (MD18105 $MN_MM_MAX_CUTTING_EDGE_NO) > (MD18106 $MN_MM_MAX_CUTTING_EDGE_PERTOOL)

Befehl	Bedeutung
CHKDNO	Prüft die vorhandenen D-Nummern auf Eindeutigkeit; Die D-Nummern aller innerhalb einer TO-Einheit definierten Werkzeuge dürfen nur einmal auftreten. Ersatzwerkzeuge werden dabei nicht berücksichtigt.
GETDNO	Ermittelt zu einer Schneide eines Werkzeuges die D-Nummer. Existiert keine D-Nummer zu den eingegebenen Parametern, wird d=0 gesetzt. Ist die D-Nummer ungültig wird ein Wert größer 32000 zurückgegeben.
SETDNO	Setzt oder ändert die D-Nummer der Schneide CE eines Werkzeuges T. Existiert kein Datensatz zu den eingegebenen Parametern wird FALSE zurückgegeben. Syntaxfehler erzeugen einen Alarm. Die D-Nummer kann nicht explizit auf 0 gesetzt werden.
GETACTTD	Ermittelt zu einer absoluten D-Nummer die dazugehörige T-Nummer. Es erfolgt keine Prüfung auf Eindeutigkeit. Gibt es mehrere gleiche D-Nummern innerhalb einer TO-Einheit, wird die T-Nummer des ersten gefundenen Werkzeuges zurückgegeben.
DZERO	Kennzeichnet alle D-Nummern der TO-Einheit als ungültig; Der Befehl dient zur Unterstützung während dem Umrüsten. So gekennzeichnete Korrekturdatensätze werden nicht mehr vom Sprachbefehl CHKDNO geprüft. Um sie wieder zugänglich zu machen, muss die D-Nummer wieder mit SETDNO gesetzt werden.

Hinweis

Ist die maximale Schneidennummer **kleiner** als die maximale Anzahl der Schneiden pro Werkzeug, sind die in der Tabelle aufgeführten Befehle ohne Wirkung. Aus Kompatibilitätsgründen, ist dies die Voreinstellung im System.

Die einzelnen Befehle sind ausführlich beschrieben in:

Weitere Informationen
Programmierhandbuch Grundlagen

Aktivierung

Um mit eindeutigen D-Nummern und damit mit den beschriebenen Sprachbefehlen arbeiten zu können, müssen die D-Nummern für die Werkzeuge frei benannt werden können.

Dazu muss folgende Voraussetzung erfüllt sein:

- MD18105 **>** MD18106

Beispiele

MD18105 $MN_MM_MAX_CUTTING_EDGE_NO = 1 (Max. Wert der D-Nummern)

Damit kann pro Werkzeug maximal eine Korrektur definiert werden (mit D-Nummer = 1).

MD18105 $MN_MM_MAX_CUTTING_EDGE_NO = 9999

Damit können Werkzeuge mit eindeutigen D-Nummern versehen werden.

Beispiel: Zuordnung von D- (Schneiden) zu T-Nummern (Werkzeugen) und Aufzeigen von Prüfungsmöglichkeiten auf Eindeutigkeit:

- Zuordnung:
  - T-Nr. 1 ⇒ Zuordnung der D-Nummern 1, 2, 3
  - T-Nr. 2 ⇒ Zuordnung der D-Nummern 10, 20, 30, 40, 50
  - T-Nr. 3 ⇒ Zuordnung der D-Nummern 100, 200, 30 (**Fehler**: Statt 300 wurde 30 angegeben)
- Prüfung auf Eindeutigkeit:
  Die D-Nummern aller innerhalb einer TO-Einheit definierten Werkzeuge müssen eindeutig sein.
  Die Prüfung erfolgt mit dem Befehl `CHKDNO`. Ohne Angabe von Parametern werden alle D-Nummern aller Werkzeuge gegeneinander geprüft. Im vorliegenden Fall ist der Rückgabewert == `FALSE`, da D30 sowohl in Wertzeug T2 und T3 vorhanden ist.

Befehl	Prüfung	Prüfergebnis
`CHKDNO`	alle D-Nummern aller Werkzeuge gegen alle anderen Werkzeuge	Rückgabewert == `FALSE`, da D=30 doppelt vorhanden ist.
`CHKDNO(2, 3, 30)`	D-Nummer 30 in Werkzeug 2 und 3	Rückgabewert == `FALSE`, da D=30 doppelt vorhanden ist.
`CHKDNO(1, 3)`	alle D-Nummern der Werkzeuge 1 und 3	Rückgabewert == `TRUE`, obwohl D=30 vom dritten Werkzeug mit D=30 vom zweiten Werkzeug kollidiert.

Hinweis

Schneiden pro Werkzeug

Werden Werkzeuge mit jeweils **n** Schneiden eingesetzt, sollte der Wert des Maschinendatums auf **n** gesetzt werden. Dies verhindert die Definition unerlaubten Schneiden für ein Werkzeug.

MD18106 $MN_MM_MAX_CUTTING_EDGE_PERTOOL = **n**

Programmierbeispiele

Umbenennen einer D-Nummer

Es soll die D-Nummer der Schneide mit CE = 3 von 2 auf den Wert 17 umbenannt werden. Dazu sollen folgende Vorgaben gelten:

- `$TC_DPx[ <Werkzeug Tn>, <Schneide Dm> ]`
- Interne T-Nummer Tn = 1

- D-Nummer Dm = 2
- Werkzeug mit einer Schneide mit:

Programmcode	Kommentar
$TC_DP2[1, 2] = 120	Werkzeuglänge = 120
$TC_DP3[1, 2] = 5.5	Werkzeugradius = 5.5
$TC_DPCE[1, 2] = 3	Schneidennummer CE = 3

- MD18105 $MN_MM_MAX_CUTTING_EDGE_NO = 20 (Max. Wert der D-Nummern)

Innerhalb des Teileprogramms wird diese Korrektur standardmäßig mit `T1,....D2` programmiert.

Sie weisen die momentane D-Nummer der Schneide 3 einer Variablen (DNrAlt) zu und definieren für die neue D-Nummer die Variable DNrNeu:

```
Programmcode
def int DNrAlt, DNrNeu = 17
DNrAlt = GETDNO( 1, 3 )
SETDNO( 1, 3, DNrNeu )
```

Damit wird der Schneide CE=3 der neue D-Wert 17 zugewiesen.

Jetzt werden die Daten dieser Schneide über die D-Nummer 17 angesprochen; sowohl über die Systemvariable, als auch in der Programmierung mit der NC-Adresse D.

Innerhalb des Teileprogramms wird diese Korrektur nun standardmäßig mit `T1,....D17` programmiert und die Daten werden folgendermaßen angesprochen:

```
Programmcode                Kommentar
$TC_DP2[ 1, 17 ] = 120
$TC_DP3[ 1, 17 ] = 5.5
$TC_DPCE[ 1, 17 ] = 3       ; Schneidennummer CE
```

Hinweis

Sofern das Werkzeug noch eine weitere Schneide definiert hat, z.B.: $TC_DPCE[1, 2] = 1 ; = CE, kann die D-Nummer 2 der Schneide 1 nicht gleich benannt werden wie die D-Nummer der Schneide 3; d.h.: `SETDNO( 1, 1, 17)` liefert den Status = `FALSE` als Rückgabewert.

DZERO - D-Nummern ungültig setzen

Mit Aktivieren dieses Befehls werden alle D-Nummern der Werkzeuge der TO-Einheit ungültig. Es kann keine Korrektur mehr aktiviert werden, bis wieder gültige D-Nummern im NC verfügbar sind. Die D-Nummern werden über den Befehl `SETDNO` neu vergeben.

Es sind folgende Werkzeuge vorgegeben (alle mit der Schneidennummer1):

T1, D1	D-Nr. der Schneide CE=1
T2, D10	D-Nr. der Schneide CE=1
T3, D100	D-Nr. der Schneide CE=1

Nun wird programmiert:

Programmcode
DZERO

Wird jetzt eine der Korrekturen aktiviert (z.B. mit `T3 D100`), wird ein Alarm erzeugt, da `D100` momentan nicht definiert ist.

Die D-Nummern werden neu definiert über:

Programmcode	Kommentar
SETDNO(1, 1, 100)	; T=1, Schneide 1 erhält die (neue) D-Nummer 100
SETDNO(2, 1, 10)	; T=2, Schneide 1 erhält die (alte) D-Nummer 10
SETDNO(3, 1, 1)	; T=3, Schneide 1 erhält die (neue) D-Nummer 1

Hinweis

Der Befehl `DZERO` kann bei Spannungsausfall den NC bzgl. der D-Nummern in einem undefinierten Zustand hinterlassen. Um dies zu beheben, wiederholen Sie nach dem Wiedereinschalten den Befehl `DZERO`.

Prinzipieller Ablauf eines Umrüstprogramms

Es soll sichergestellt werden, dass die benötigten Werkzeuge und Schneiden vorhanden sind. Das Magazin, der Werkzeugspeicher der NC, befindet sich in einem beliebigen Zustand. Die D-Nummern in den Teileprogrammen der neuen Bearbeitung stimmen meist nicht mit den D-Nummern der konkreten Schneiden überein. Das Umrüstprogramm kann wie folgt aussehen:

Programmcode	Kommentar
DZERO	; alle D-Nummern der TO-Einheit werden als ungültig gekennzeichnet
....	; Schleife oder mehrere Schleifen über die Plätze des Magazins bzw. der Magazine mit Prüfung der Werkzeuge und ihrer Schneidennummern.
	; Wird ein WZ gefunden, das noch nicht gesperrt ist ($TC_TP8) und das die gewünschte Schneidennummer CE hat (GETDNO), dann wird die Schneide mit der neuen D-Nummer versehen (SETDNO).
....	; Entlade- und Beladevorgänge werden durchgeführt.
	; Es kann mit dem Status 'zu entladen' bzw. 'zu beladen' des Werkzeuges gearbeitet werden.
CHKDNO	; Das Ent-/ Beladen und der Vorgang zum Umbenennen der D-Nummern sind abgeschlossen.
	; Es können einzelne Werkzeuge und / oder D-Nummern geprüft und anhand des Rückgabewertes auftretende Kollisionen automatisch behandelt werden.

2.3.5 Korrektursatz bei Fehler im Werkzeugwechsel

MD22550

Wenn im Teileprogramm eine Werkzeugvorbereitung programmiert ist und der NC stellt dabei einen Fehler fest (z.B. der Datensatz zur programmierten T-Nummer ist im NC nicht vorhanden), hat der Bediener die Möglichkeit, die Fehlersituation zu beurteilen und entsprechende Handlungen vorzunehmen, um anschließend die Bearbeitung fortzusetzen.

Der Werkzeugwechsel kann unabhängig programmiert werden, in Abhängigkeit des Maschinendatums:

MD22550 $MC_TOOL_CHANGE_MODE (Neue Werkzeugkorrektur bei M-Funktion)

MD22550 $MC_TOOL_CHANGE_MODE = 0

```
T= 'T-Nr.'        ; WZ-Vorbereitung + Werkzeugwechsel in einem NC-Satz,
                  ; d.h. mit Programmierung von T erfolgt im NC das
                  ; Aktivwerden einer neuen D-Korrektur. Siehe dazu das Maschinenda-
                  tum:
                  ; MD20270 $MC_CUTTING_EDGE_DEFAULT (Grundstellung der Werkzeug-
                  schneide ohne Programmierung)
```

MD22550 $MC_TOOL_CHANGE_MODE = 1

```
T= 'T-Nr.'        ; Werkzeugvorbereitung
M06               ; Werkzeugwechsel
                  ; (die Nummer des WZ-Wechsel-M-Codes ist einstellbar)
                  ; d.h. mit Programmierung von M06 erfolgt in NC das
                  ; Aktivwerden einer neuen D-Korrektur; (siehe dazu das
                  ; Maschinendatum MD20270 $MC_CUTTING_EDGE_DEFAULT)
```

Bei nicht aktiver Werkzeugverwaltung können folgende Probleme erkannt werden:

- D-Korrekturdatensatz fehlt
- Fehler im Teileprogramm

Hinweis

Das Problem "Werkzeug ist nicht im Magazin" kann nicht erkannt werden, da bei der Werkzeugkorrektur dem NC keine Magazininformationen vorliegen.

D-Korrekturdatensatz fehlt

Die Programmabarbeitung bleibt auf dem Satz mit dem fehlerhaften D-Wert stehen (unabhängig vom Wert des Maschinendatums MD22550). Der Bediener muss entweder das Programm korrigieren, oder er muss den fehlenden Datensatz nachladen. Dazu benötigt er D- und T-Nummer. Diese Parameter werden durch den ausgelösten Alarm mit übergeben.

Fehler im Teileprogramm

Die Bearbeitungsmöglichkeiten im Fehlerfall hängen davon ab, wie der Werkzeugwechsel programmiert wurde, festgelegt über das Maschinendatum:

MD22550 $MC_TOOL_CHANGE_MODE (Neue Werkzeugkorrektur bei M-Funktion)

Werkzeugwechsel mit T-Programmierung (MD22550 = 0)

Für diesen Fall erfolgt die Bearbeitung über die im NC verfügbare Funktion "Korrektursatz". Das NC-Programm stoppt auf dem NC-Satz, in dem ein Fehler beim programmierten T-Wert erkannt wurde. Beim Fortsetzen des Programms wird der "Korrektursatz" erneut abgearbeitet.

Durch den Bediener können folgende Bearbeitungen vorgenommen werden:

- Das fehlerhafte Teileprogramm kann korrigiert werden.
- Fehlende Schneidenkorrektur-Daten können von HMI nachgeliefert werden.
- Fehlende Schneidenkorrektur-Daten können in "Überspeichern" im NC eingebracht werden.

Nach dem Bedienereingriff wird die `START`-Taste gedrückt und der beanstandete Satz wird wieder abgearbeitet. Wurde der Fehler richtig korrigiert, wird das Programm fortgesetzt. Ansonsten wird erneut ein Alarm erzeugt.

Werkzeugwechsel mit T und M06-Programmierung (MD22550 = 1)

In diesem Fall wird im NC-Satz der WZ-Vorbereitung (Programmierung von T) ein Fehler bemerkt, der aber zunächst ignoriert werden soll. Die Verarbeitung wird fortgesetzt, bis die WZ-Wechselanforderung (meist `M06`) im NC-Programm abgearbeitet wird. An dieser Stelle soll das Programm "halten".

Die programmierte T-Adresse kann beliebig viele Programmzeilen vor dem M06-Befehl stehen, oder die beiden Anweisungen können in unterschiedlichen (Unter-)Programmen stehen. Deshalb ist es im Allgemeinen nicht möglich, einen abgearbeiteten Satz über Korrektursatz zu ändern.

Der Bediener hat dieselben Bearbeitungsmöglichkeiten wie bei = 0.

Das Nachladen fehlender Daten ist möglich. Dann muss aber beim "Überspeichern" T programmiert werden.

Sofern ein Programmfehler vorliegt, kann nicht die Zeile mit dem Fehler korrigiert werden (Txx), sondern nur die Zeile, auf der die Programmabarbeitung gestoppt und den Alarm erzeugt hat. Nur wenn das Maschinendatum:

MD22562 $MC_TOOL_CHANGE_ERROR_MODE Bit0 = 1 (Verhalten bei Fehlern im Werkzeugwechsel)

Es ergibt sich folgender Ablauf:

```
Txx              ; Fehler! Datensatz mit xx gibt es nicht
                 ; Zustand merken; xx merken;
                 ; weitermachen im Programm
....
```

```
M06              ; Merker 'xx fehlt' erkennen → Alarm ausgegeben,
                 ; Programm stoppen
                 ; Satz korrigieren mit z.B. Tyy M06, starten,
                 ; der Satz Tyy M06 wird interpretiert und ist o.k.
                 ; Die Bearbeitung geht weiter.
```

Bei erneuter Abarbeitung der Programmstelle ergibt sich folgendes:

```
Txx              ; Fehler! Datensatz mit xx gibt es nicht,
                 ; Zustand merken; xx merken;
                 ; weitermachen im Programm
....
Tyy M06          ; Merker "xx fehlt" erkennen → ohne weitere Reaktion verwerfen,
                 ; denn Tyy M06 ist korrekt → Programm stoppt nicht (richtig).
```

Falls erforderlich kann nach Programmende die ursprüngliche Stelle des T-Aufrufs korrigiert werden. Falls die WZ-Wechsellogik an der Maschine dies nicht verarbeiten kann, muss das Programm abgebrochen und die fehlerhafte Stelle korrigiert werden.

Sofern nur der Datensatz fehlt, wird dieser zum NC übertragen, in "Überspeichern" Txx programmiert und anschließend das Programm fortgesetzt.

Wie bei "fehlender D-Nummer" wird auch bei "fehlender T-Nummer" über den entsprechenden Alarm (17191) der erforderliche Parameter (T-Nummer) dem Bediener zur Verfügung gestellt.

Hinweis

Um die Programmkorrektur zu ermöglichen, wird sofort bei dem fehlerhaften Txx-Satz angehalten.

Bei Programmtestbetrieb wird auch angehalten beim Maschinendatum:

MD22562 $MC_TOOL_CHANGE_ERROR_MODE Bit0=1 (Verhalten bei Fehlern im Werkzeugwechsel)

2.3.6 Definition der Wirkung der Werkzeugparameter

MD20360

Die Wirkung der Werkzeugparameter auf die Planachse kann im Zusammenhang mit Durchmesserprogrammierung gezielt gesteuert werden mit dem Maschinendatum:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK (Definition der WZ-Parameter)

Details sind beim genannten MD beschrieben.

DRF manuelles Verfahren per Handrad mit halbem Weg

Beim DRF-Hanradverfahren kann eine Planachse nur mit dem halben Weg des vorgegebenen Inkrements wie folgt verfahren werden:

Festlegung der Wegvorgabe mit Handrad über das Maschinendatum:

MD11346 $MN_HANDWHEEL_TRUE_DISTANCE = 1 (Handrad Weg- oder Geschwindigkeitsvorgabe)

Definition der DRF-Verschiebung in der Planachse als Durchmesserkorrekturwert einrechnen mit dem Maschinendatum:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK Bit 9 = 1 (Definition der WZ-Parameter)

Bei der axialen Abwahl `DRFOF` einer DRF-Verschiebung wird eine existierende Werkzeugkorrektur (Handradüberlagerung in Werkzugrichtung) auch gelöscht.

Hinweis

Weitere Informationen zu überlagerten Bewegungen mit Handrad entnehmen Sie bitte:

Weitere Informationen
Funktionshandbuch Achsen und Spindeln; Manuelles Verfahren

Programmierhandbuch Grundlagen

(Die Programmieranleitung beschreibt die vollständigen programmtechnischen Möglichkeiten, um die DRF-Verschiebung achsspezifisch abzuwählen.)

2.4 Werkzeugparameter

2.4.1 Übersicht

Werkzeugparameter

Eine Werkzeugschneide ist eindeutig durch folgende Angaben beschrieben:

- Werkzeugtyp (Schaftfräser, Bohrer, etc.)
- Geometrische Beschreibung
- Technologische Beschreibung

Die geometrische Beschreibung, die technologische Beschreibung und der WZ-Typ werden für jede WZ-Schneide auf WZ-Parameter abgebildet.

Folgende WZ-Parameter sind vorhanden:

WZ-Parameter	Bedeutung	Bemerkung
1	Werkzeugtyp	
2	WZ-Schneidenlage	Für Drehwerkzeuge oder für Fräs-/Schleifwerkzeuge bei 2D-WRK mit Konturwerkzeugen.

WZ-Parameter	Bedeutung	Bemerkung
Geometrie - Werkzeuglängen		
3	Länge 1	
4	Länge 2	
5	Länge 3	
Geometrie - Werkzeugform		
6	Radius 1	Bei 2D-WRK. Drehen: Verrundungsradius der Schneide 3D-Stirnfräsen: Werkzeugradius Nutsäge: Durchmesser
7	Radius 2	3D-Stirnfräsen: Eckenradius (bei Werkzeugen mit Eckenverrundung) Nutsäge: Nutbreite
8	Länge 4	Nutsäge: Überstand
9	Länge 5	3D-Stirnfräsen: Oberer Kegelstumpfdurchmesser (bei kegelförmigen Werkzeugen)
10	Winkel 1	Drehen: Halterwinkel 2D-WRK mit Konturwerkzeugen: Minimaler Grenzwinkel
11	Winkel 2	Drehen: Schnittrichtung 2D-WRK mit Konturwerkzeugen: Maximaler Grenzwinkel 3D-Stirnfräsen: Winkel zwischen Mantellinie und Werkzeuglängsachse (bei kegelförmigen Werkzeugen)
Verschleiß - Werkzeuglängen		
12	Verschleiß Länge 1	
13	Verschleiß Länge 2	
14	Verschleiß Länge 3	
Verschleiß - Werkzeugform		
15	Verschleiß Radius 1	
16	Verschleiß Radius 2	
17	Verschleiß Länge 4	
18	Verschleiß Länge 5	
19	Verschleiß Winkel 1	
20	Verschleiß Winkel 2	
Basismaß / Adaptermaß		
21	Basis-Länge 1	
22	Basis-Länge 2	
23	Basis-Länge 3	
Technologie		

WZ-Parameter	Bedeutung	Bemerkung
24	Freiwinkel	Nur für Drehwerkzeuge.
25	Reserviert*	

* "Reserviert" bedeutet, dass dieser WZ-Parameter nicht benutzt wird und für Erweiterungen reserviert ist.

2.4.2 WZ-Parameter 1: Werkzeugtyp

Der Werkzeugtyp legt fest, um welches Werkzeug es sich handelt. Mit der Auswahl des Werkzeugtyps sind die anderen Komponenten wie Geometrie, Verschleiß oder Basismaß vorbestimmt.

Jedem Werkzeugtyp ist eine eindeutige 3-stellige Nummer zugeordnet. Über die erste Ziffer (Hunderterstelle) erfolgt die Zuordnung des Werkzeugs zu einer der folgenden Technologien bzw. Werkzeuggruppen:

Werkzeugtyp	Werkzeuggruppe
1xy	Fräswerkzeuge
2xy	Bohrwerkzeuge
3xy	reserviert
4xy	Schleifwerkzeuge
5xy	Drehwerkzeuge
6xy	reserviert
7xy	Sonderwerkzeuge

Für den Parameter "Werkzeugtyp" gilt:

- Der WZ-Typ muss für jede WZ-Schneide angegeben werden.
- Der WZ-Typ darf nur die festgelegten Werte annehmen.
- Der WZ-Typ "0" bedeutet, dass kein gültiges Werkzeug definiert ist.

Relevante Werkzeugparameter

Die abhängig vom Werkzeugtyp einzutragenden Parameterwerte (Werkzeugkorrekturdaten; TOA-Daten) werden in Systemvariablen ($TC_DPx und $TC_TPGx) abgelegt.

Eine Übersicht, welche Parameterwerte für welchen Werkzeugtyp erforderlich sind, zeigt folgende Tabelle:

Schneiden-Parameter $TC_DPx																								WZ-Parameter $TC_TPGx					
x =	1		2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	3	4	5	6	7	8
	Werkzeugtyp		Schneidenlage	Geometrie - Werkzeuglängen: Länge 1	Geometrie - Werkzeuglängen: Länge 2	Geometrie - Werkzeuglängen: Länge 3	Geometrie - Werkzeugform: Radius 1	Geometrie - Werkzeugform: Radius 2	Geometrie - Werkzeugform: Länge 4	Geometrie - Werkzeugform: Länge 5	Geometrie - Werkzeugform: Winkel 1	Geometrie - Werkzeugform: Winkel 2	Verschleiß - Werkzeuglängen: Länge 1	Verschleiß - Werkzeuglängen: Länge 2	Verschleiß - Werkzeuglängen: Länge 3	Verschleiß - Werkzeugform: Radius 1	Verschleiß - Werkzeugform: Radius 2	Verschleiß - Werkzeugform: Länge 4	Verschleiß - Werkzeugform: Länge 5	Verschleiß - Werkzeugform: Winkel 1	Verschleiß - Werkzeugform: Winkel 2	Basismaß / Adaptermaß: Länge 1	Basismaß / Adaptermaß: Länge 2	Basismaß / Adaptermaß: Länge 3	Freiwinkel	Min. Scheibenradius	Min. Scheibenbreite	Akt. Scheibenbreite	Max. Drehzahl	Max. Umfangsgeschw.	Winkel schräge Scheibe
Fräswerkzeuge	100	Fräswerkzeug nach CLDATA [1]		✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓							
	110	Kugelkopffräser		✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓							
	111	Zylindrischer Gesenkfräser		✓	✓	✓	✓	✓					✓	✓	✓	✓	✓					✓	✓	✓							
	120	Schaftfräser		✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓							
	121	Schaftfräser mit Eckenverrundung		✓	✓	✓	✓	✓					✓	✓	✓	✓	✓					✓	✓	✓							
	130	Winkelkopffräser		✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓							
	131	Winkelkopffräser mit Eckenverrundung		✓	✓	✓	✓	✓					✓	✓	✓	✓	✓					✓	✓	✓							
	140	Planfräser		✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓							
	145	Gewindefräser		✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓							
	150	Scheibenfräser		✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓							
	155	Kegelstumpffräser		✓	✓	✓	✓			✓		✓	✓	✓	✓	✓			✓		✓	✓	✓	✓							
	156	Kegelstumpffräser mit Eckenverrundung		✓	✓	✓	✓	✓		✓		✓	✓	✓	✓	✓	✓		✓		✓	✓	✓	✓	✓						
	157	Kegeliger Gesenkfräser		✓	✓	✓	✓					✓	✓	✓	✓	✓					✓	✓	✓	✓							
	160	Bohrgewindefräser		✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓							
Bohrwerkzeuge	200	Spiralbohrer		✓	✓	✓	✓						✓	✓	✓							✓	✓	✓							
	205	Vollbohrer		✓	✓	✓	✓						✓	✓	✓							✓	✓	✓							
	210	Bohrstange		✓	✓	✓	✓						✓	✓	✓							✓	✓	✓							
	220	Zentrierbohrer		✓	✓	✓	✓						✓	✓	✓							✓	✓	✓							
	230	Spitzsenker		✓	✓	✓	✓						✓	✓	✓							✓	✓	✓							
	231	Flachsenker		✓	✓	✓	✓						✓	✓	✓							✓	✓	✓							
	240	Gewindebohrer Regelgewinde		✓	✓	✓	✓						✓	✓	✓							✓	✓	✓							
	241	Gewindebohrer Feingewinde		✓	✓	✓	✓						✓	✓	✓							✓	✓	✓							
	242	Gewindebohrer Whitworthgewinde		✓	✓	✓	✓						✓	✓	✓							✓	✓	✓							
	250	Reibahle		✓	✓	✓	✓						✓	✓	✓							✓	✓	✓							
Schleifwerkzeuge	400	Umfangsschleifscheibe	✓	✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓						✓	✓
	401	Umfangsschleifscheibe mit Überwachung	✓	✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓		✓	✓	✓	✓	✓	✓
	403	wie 401 aber ohne Basismaß für SUG [2]	✓	✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓		✓	✓	✓	✓		✓
	410	Planschleibe	✓	✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓						✓	
	411	Planschleibe mit Überwachung	✓	✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓		✓	✓	✓	✓	✓	
	412	Planschleibe ohne Überwachung	✓	✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓		✓	✓	✓	✓		
	413	wie 411 aber ohne Basismaß für SUG [2]	✓	✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓		✓	✓	✓	✓		
	490	Abrichter	✓	✓	✓	✓	✓						✓	✓	✓	✓						✓	✓	✓							
Drehwerkzeuge	500	Schruppstahl	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓						
	510	Schlichtstahl	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓						
	520	Einstechstahl	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓						
	530	Abstechstahl	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓						
	540	Gewindestahl	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓						
	550	Pilzstahl / Formstahl (WZV)	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓						
	560	Drehbohrer (ECOCUT)	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓						
	580	Messtaster mit Parameter Schneidenlage	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓						
SW [3]	700	Nutsäge		✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓				✓	✓	✓							

[1] CLDATA = "cutter location data" (Werkzeugpositionsdaten nach DIN66215)

[2] SUG = Schleifscheibenumfangsgeschwindigkeit

[3] SW = Sonderwerkzeuge

Hinweis

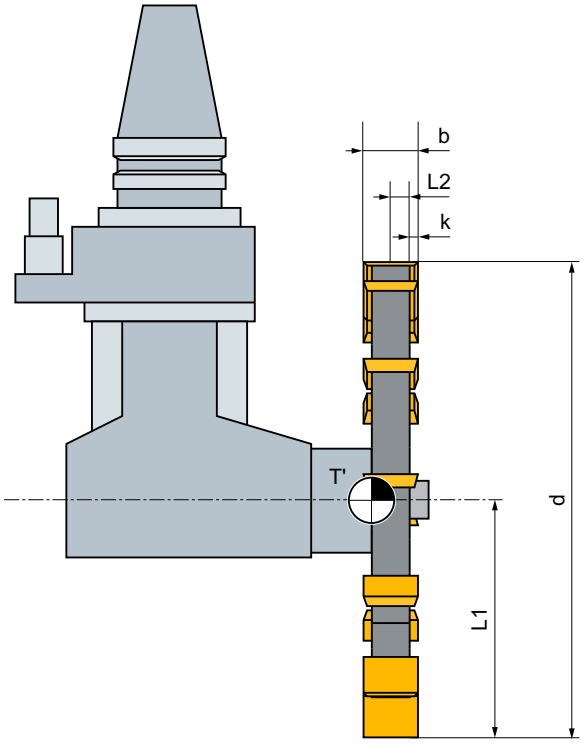
Innerhalb der Gruppe Drehwerkzeuge hat der Werkzeugtyp keine Bedeutung.

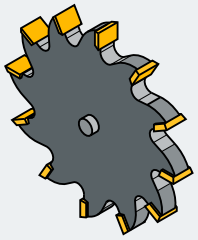
Insbesondere bei den Schleifwerkzeugen (400-499) sind auch nicht aufgeführte Nummern zulässig.

Verrechnung der Werkzeugkorrekturdaten

Die in den Werkzeugkorrekturdaten abgelegten Einzelkomponenten verrechnet die NC-Steuerung zu einer resultierenden Größe (z. B. Gesamtlänge, Gesamtradius). Das jeweilige Gesamtmaß kommt bei Aktivierung des Korrekturspeichers zur Wirkung. Die Verrechnung dieser Werte in den Achsen wird durch den Werkzeugtyp und die aktuelle Bearbeitungsebene G17, G18 oder G19 festgelegt.

Beispiel: Werkzeugtyp Nutsäge (Typ 700)





L1 = d/2
L2 = b/2 - k

T'	Werkzeughalterbezugspunkt
L1	Geometrie - Länge 1
L2	Geometrie - Länge 2
d	Durchmesser
b	Nutbreite
k	Überstand

	Geometrie	Verschleiß	Basismaß	Einheit
Werkzeuglängenkorrektur				
Länge 1	$TC_DP3	$TC_DP12	$TC_DP21	mm
Länge 2	$TC_DP4	$TC_DP13	$TC_DP22	mm
Länge 3	$TC_DP5	$TC_DP14	$TC_DP23	mm
Sägeblattkorrektur				
Durchmesser d	$TC_DP6	$TC_DP15		mm
Nutbreite b	$TC_DP7	$TC_DP16		mm
Überstand k	$TC_DP8	$TC_DP17		mm

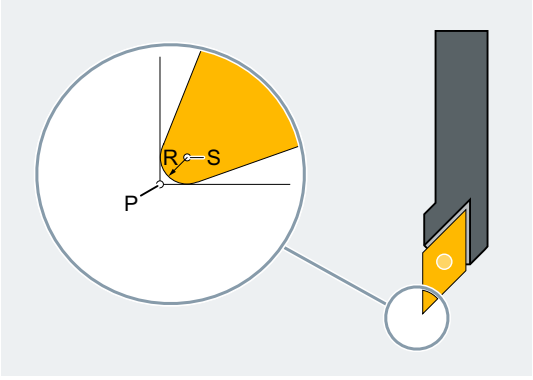
Die Breite des Sägeblatts wird bei der Werkzeugradiuskorrektur (G40/G41/G42) wie folgt berücksichtigt:

Funktion	Bedeutung
G40	keine Sägeblattkorrektur
G41	Sägeblattkorrektur links
G42	Sägeblattkorrektur rechts

2.4.3 WZ-Parameter 2: Schneidenlage

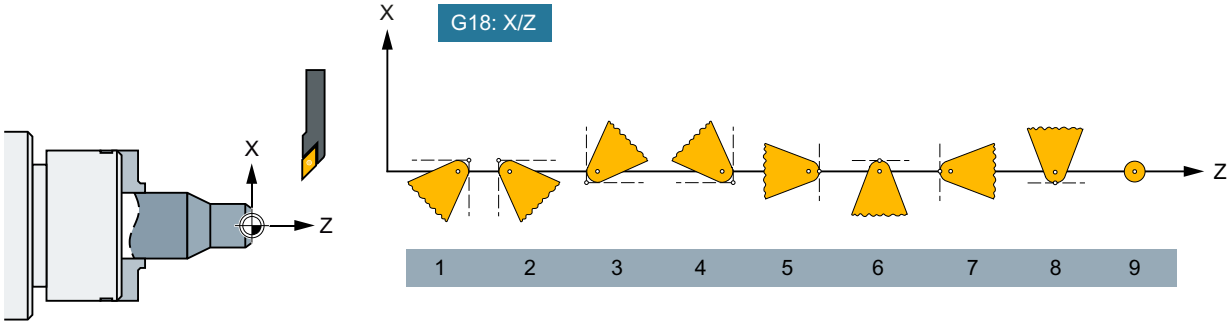
Die Schneidenlage beschreibt die Lage der Werkzeugspitze P in Bezug auf den Schneidenmittelpunkt S. Sie wird in den WZ-Parameter 2 eingetragen.

Die Schneidenlage wird zusammen mit dem Schneidenradius (WZ-Parameter 8) zur Berechnung der Werkzeugradiuskorrektur bei Drehwerkzeugen (Werkzeugtyp 5xx) benötigt.

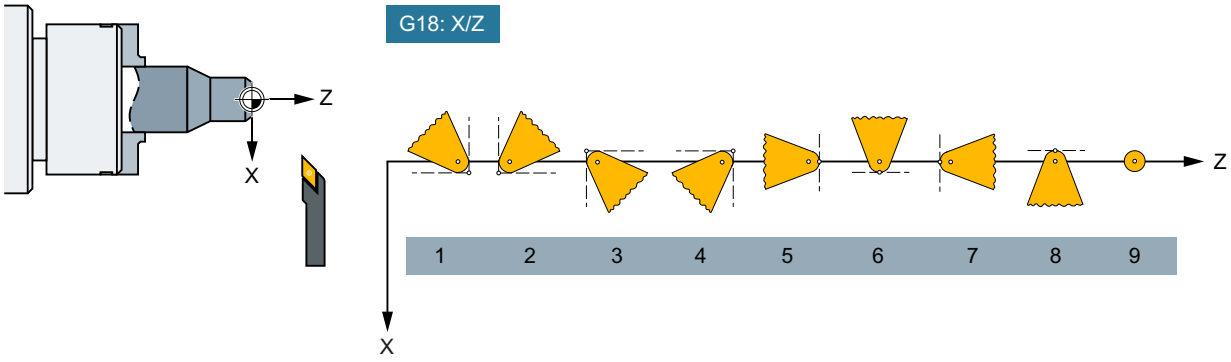


P Werkzeugspitze
S Schneidenmittelpunkt
R Schneidenradius

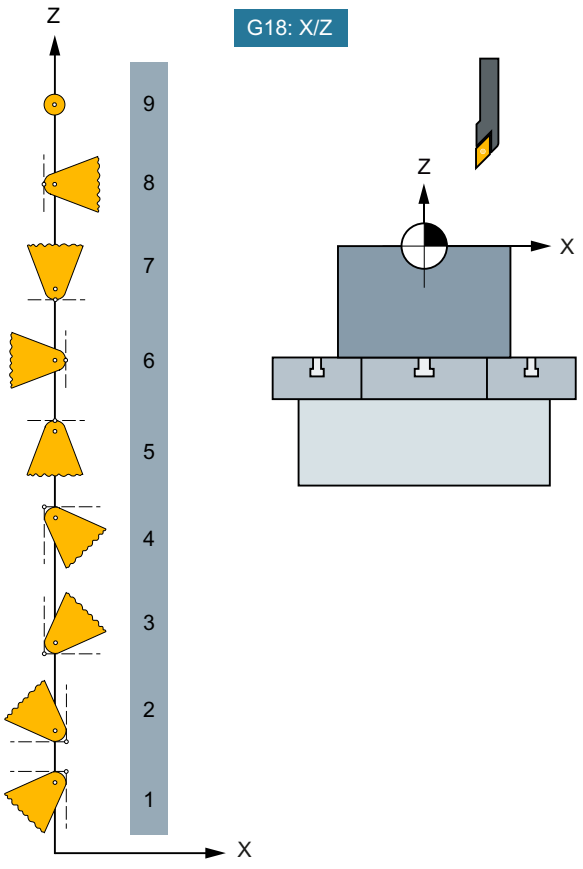
Schneidenlage (1 ... 9) bei Bearbeitung hinter der Drehmitte



Schneidenlage (1 ... 9) bei Bearbeitung vor der Drehmitte



Schneidenlage (1 ... 9) bei Karusselldrehmaschinen



Besonderheiten

- Wird zur Ermittlung der WZ-Längenkorrektur als Bezugspunkt nicht der Punkt P, sondern der Schneidenmittelpunkt S benützt, ist als WZ-Schneidenlage die Kennung 9 einzugeben.
- Die Kennung 0 (null) als WZ-Schneidenlage ist nicht zulässig.

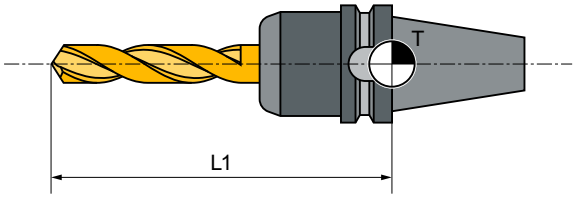
2.4.4 WZ-Parameter 3 ... 5: Geometrie - Werkzeuglängen

Die Längen der Werkzeuge werden für die Geometrie-Werkzeuglängenkorrektur benötigt. Sie werden als WZ-Länge 1 bis 3 in die WZ-Parameter 3 bis 5 eingegeben.

Welche WZ-Längen mindestens erforderlich sind, ist vom WZ-Typ abhängig:

Werkzeugtyp	Erforderliche WZ-Längen
WZ-Typ 12x, 140, 145, 150	WZ-Länge 1
WZ-Typ 13x	WZ-Länge 1 ... 3 (je nach Ebene G17/G18/G19)
WZ-Typ 2xx	WZ-Länge 1
WZ-Typ 5xx	WZ-Länge 1 ... 3

Beispiel: Spiralbohrer (WZ-Typ 200)



T Werkzeugträgerbezugspunkt

L1 Länge 1 (WZ-Parameter 3)

Hinweis

Unabhängig vom WZ-Typ werden immer alle drei WZ-Parameter 3 bis 5 (WZ-Länge 1 bis 3) in den drei Geometrieachsen verrechnet. Wenn für einen WZ-Typ mehr WZ-Längen in die WZ-Parameter 3 bis 5 eingegeben werden als mindestens erforderlich sind, dann werden diese zusätzlichen WZ-Längen in den Geometrieachsen ohne Alarm verrechnet.

Erst die Addition der Geometrie-Werkzeuglängenkorrektur (WZ-Parameter 3 bis 5) und der Verschleiß-Werkzeuglängenkorrektur (WZ-Parameter 12 bis 14) legt die aktive Größe des Werkzeugs fest. Zur Verrechnung der gesamten Werkzeuglängenkorrektur in den Geometrieachsen wird auch die Basismaß/Adaptermaß-Werkzeuglängenkorrektur addiert.

2.4.5 WZ-Parameter 6 ... 11: Geometrie - Werkzeugform

Mit den WZ-Parametern 6 bis 11 wird die Form des Werkzeugs festgelegt. Die Angaben werden für die Geometrie-Werkzeugradiuskorrektur benötigt.

In den meisten Fällen wird nur der WZ-Parameter 6 (WZ-Radius 1) benutzt:

WZ-Parameter	Bedeutung	Verwendung	
6	WZ-Radius 1	Für folgende WZ-Typen muss der WZ-Radius angegeben werden:	
		1xx	Fräswerkzeuge
		5xx	Drehwerkzeuge Bei Drehwerkzeugen muss auch WZ-Parameter 2: Schneidenlage (Seite 42) angegeben werden.
		Hinweis WZ-Typ 2xx: Für Bohrwerkzeuge muss kein WZ-Radius angegeben werden.	
7	WZ-Radius 2	nicht verwendet	
8	WZ-Länge 4	nicht verwendet	
9	WZ-Länge 5	nicht verwendet	
10	WZ-Winkel 1	nicht verwendet	
11	WZ-Winkel 2	nicht verwendet	

2D-WRK mit Konturwerkzeugen

Zur Definition von Konturwerkzeugen mit mehreren Werkzeugschneiden können der minimale und maximale Grenzwinkel angegeben werden. Die beiden Grenzwinkel beziehen sich jeweils auf den Vektor vom Schneidenmittelpunkt zum Schneidenbezugspunkt und werden im Gegenuhrzeigersinn gezählt.

WZ-Winkel 1	Minimaler Grenzwinkel je Werkzeugschneide
WZ-Winkel 2	Maximaler Grenzwinkel je Werkzeugschneide

Weitere Informationen

Zur Eingabe der Werkzeugform in die WZ-Parameter 6 bis 11 und die Verrechnung in den drei Geometrieachsen durch die Geometrie-Werkzeugradiuskorrektur siehe:

Programmierhandbuch NC-Programmierung; Kapitel "Grundlagen" > "Werkzeugkorrekturen" > "2½ D-Werkzeugkorrektur"

2.4.6 WZ-Parameter 12 ... 14: Verschleiß - Werkzeuglängen

Während mit der Geometrie-Werkzeuglängenkorrektur (WZ-Parameter 3 bis 5) die Größe des Werkzeugs eingestellt wird, kann mit der Verschleiß-Werkzeuglängenkorrektur die Veränderung der aktiven Werkzeuggröße korrigiert werden.

Die aktive Werkzeuggröße kann sich durch folgende Einflüsse ändern:

- Unterschiede zwischen der WZ-Aufnahme in der WZ-Messmaschine und der WZ-Aufnahme in der Werkzeugmaschine
- Verschleiß des Werkzeugs durch die Einsatzdauer bei der Bearbeitung
- Definition eines Schlichtaufmaßes

Die Geometrie-Werkzeuglängenkorrektur (WZ-Parameter 3 bis 5) und die Verschleiß-Werkzeuglängenkorrektur (WZ-Parameter 12 bis 14) werden addiert (Geometrie-WZ-Länge 1 mit Verschleiß-WZ-Länge 1, ...) und stellen so die **Größe des aktiven Werkzeugs** dar.

2.4.7 WZ-Parameter 15 ... 20: Verschleiß - Werkzeugform

Während mit der Geometrie-Werkzeugradiuskorrektur (WZ-Parameter 6 bis 11) die Form des Werkzeugs eingestellt wird, kann mit der Verschleiß-Werkzeugradiuskorrektur die Veränderung der aktiven Werkzeugform korrigiert werden.

Die aktive Werkzeuggröße kann sich durch folgende Einflüsse ändern:

- Verschleiß des Werkzeugs durch die Einsatzdauer bei der Bearbeitung
- Definition eines Schlichtaufmaßes

Die Geometrie-Werkzeugradiuskorrektur (WZ-Parameter 6 bis 11) und die Verschleiß-Werkzeugradiuskorrektur (WZ-Parameter 15 bis 20) werden addiert (Geometrie-WZ-Radius 1 mit Verschleiß-WZ-Radius 1) und stellen so die **Form des aktiven Werkzeugs** dar.

2.4.8 WZ-Parameter 21 ... 23: Basismaß / Adaptermaß

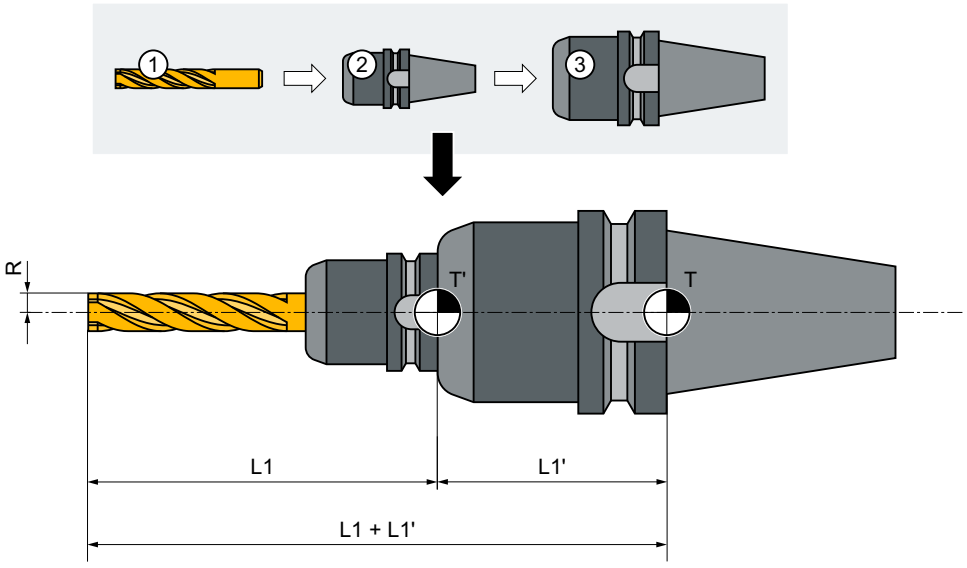
Das Basismaß/Adaptermaß kann verwendet werden, wenn der Bezugspunkt des Werkzeughalters (WZ-Größe) nicht mit dem Bezugspunkt des Werkzeugträgers übereinstimmt.

Das ist in folgenden Situationen der Fall:

- WZ und WZ-Adapter wurden getrennt vermessen, aber schon zusammengebaut in die Maschine eingesetzt (WZ-Größe und Adaptergröße werden getrennt in eine WZ-Schneide eingegeben).
  → Siehe Beispiel 1.
- Das WZ wird in eine zweite Werkzeugaufnahme eingesetzt, die sich an einer anderen Position befindet (z. B. Vertikal - und Horizontalspindel).
- Die Werkzeugaufnahmen eines Werkzeugrevolvers besitzen unterschiedliche Positionen.
  → Siehe Beispiel 2.

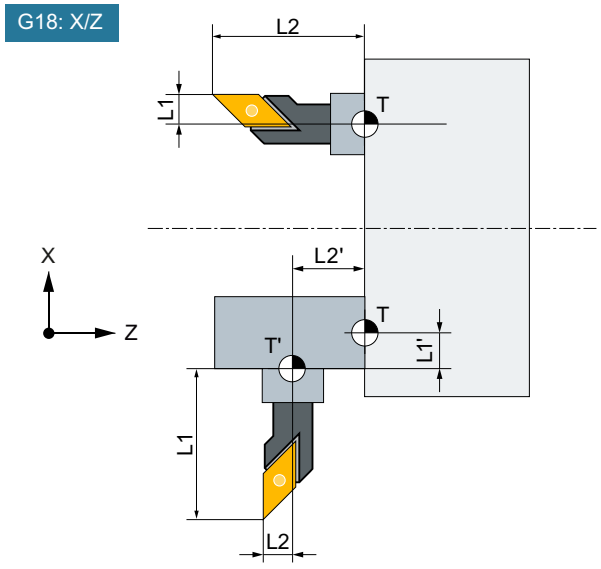
Damit der Werkzeughalterbezugspunkt T' zum Werkzeugträgerbezugspunkt T in den drei Geometrieachsen (3-dimensional) korrigiert werden kann, wirken unabhängig von WZ-Typ immer alle 3 Basis-Längen. D. h., auch ein Spiralbohrer (WZ-Typ 200) mit einer Werkzeuglängenkorrektur (Länge 1) kann ein Basismaß/Adaptermaß in 3 Achsen haben.

Beispiel 1: WZ und WZ-Adapter wurden getrennt vermessen, aber schon zusammengebaut in die Maschine eingesetzt



①	Werkzeug
②	Werkzeugaufnahme
③	Werkzeugadapter
T	Adapterbezugspunkt (bei eingestecktem Werkzeug = Werkzeugträgerbezugspunkt)
T'	Werkzeughalterbezugspunkt
L1	Geometrie - Länge 1
L1'	Adaptermaß - Länge 1
L1 + L1'	Gesamtlänge L1
R	Radius

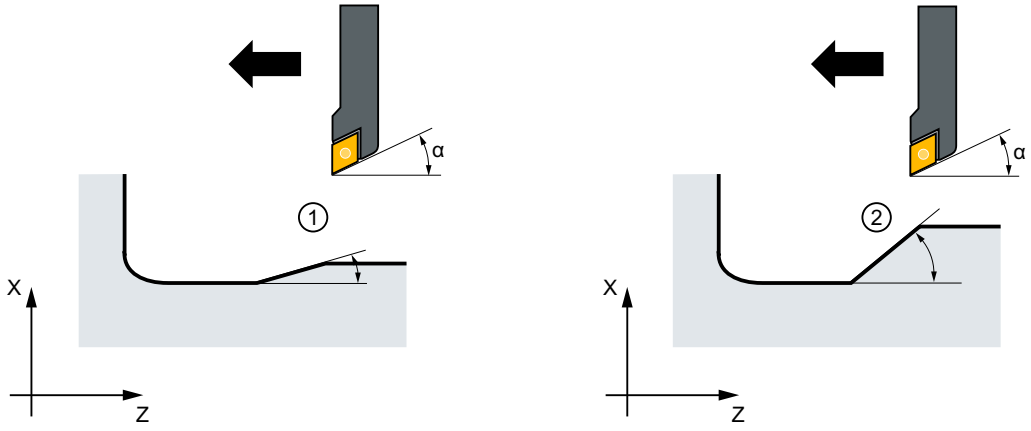
Beispiel 2: Die Werkzeugaufnahmen eines Werkzeugrevolvers besitzen unterschiedliche Positionen



T	Werkzeugträgerbezugspunkt
T'	Werkzeughalterbezugspunkt
L1	Geometrie - Länge 1
L2	Geometrie - Länge 2
L1'	Basismaß - Länge 1
L2'	Basismaß - Länge 2

2.4.9 WZ-Parameter 24: Freiwinkel

Bestimmte Drehzyklen, in denen Verfahrbewegungen mit Hinterschneiden erzeugt werden, überwachen den Freiwinkel des aktiven Werkzeugs auf eine mögliche Konturverletzung.



α Freiwinkel

① Zu bearbeitende Kontur wird nicht verletzt.

② Zu bearbeitende Kontur würde verletzt werden.

Der Freiwinkel wird in den WZ-Parameter 24 eingegeben.

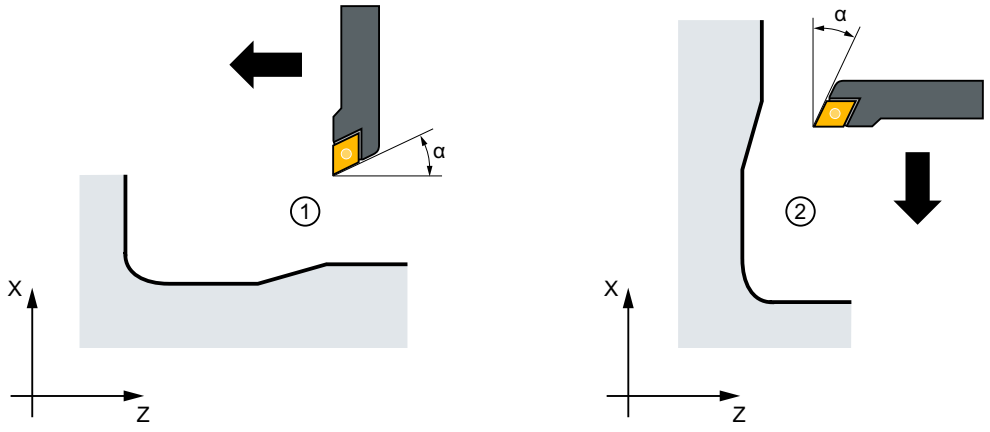
Wertebereich: 0° ... 90° (ohne Vorzeichen)

Hinweis

Wird im WZ-Parameter 24 eine null eingegeben, erfolgt in den Drehzyklen keine Überwachung beim Hinterschneiden.

Längs- oder Planbearbeitung

Der Freiwinkel ist abhängig von der Bearbeitungsart unterschiedlich einzugeben. Soll ein Werkzeug für Längs- **und** Planbearbeitung eingesetzt werden, müssen bei unterschiedlichen Freiwinkeln zwei WZ-Schneiden definiert werden.



α Freiwinkel
① Längsbearbeitung
② Planbearbeitung

Weitere Informationen

Ausführliche Informationen zum Freiwinkel siehe Programmierhandbuch Zyklen.

2.5 2D-Werkzeugradiuskorrektur

2.5.1 Allgemeine Informationen

Hinweis

Zur Werkzeugradiuskorrektur (WRK) siehe:
Weitere Informationen
Programmierhandbuch Grundlagen

Nur die Programmieranleitung beschreibt die vollständigen programmtechnischen Möglichkeiten der Werkzeugradiuskorrektur (WRK) mit ihren Sonderfällen.

Warum WRK?

Die im Teileprogramm programmierte Kontur (Geometrie) des Werkstücks soll unabhängig von den in der Fertigung eingesetzten Werkzeugen sein. Dazu ist es nötig, die Werte für Werkzeuglänge und Werkzeugradius einem aktuellen Korrekturspeicher zu entnehmen. Durch die WRK kann mit dem aktuellen Werkzeugradius die Äquidistante zur programmierten Kontur in der Ebene bestimmt werden.

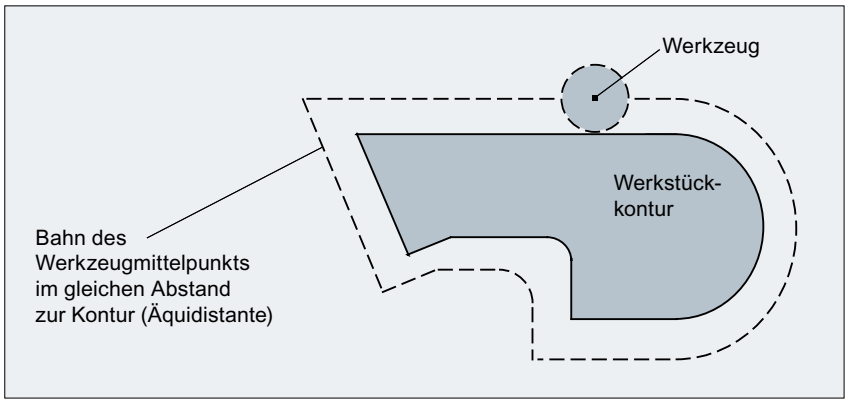


Bild 2-2 Werkstückkontur (Geometrie) mit Äquidistante

WRK in der Ebene

Die WRK wirkt in der aktuellen Ebene (`G17` bis `G19`) bei folgenden Interpolationsarten:

• Linearinterpolation	...	`G0, G1`
• Kreisinterpolation	...	`G2, G3, CIP`
• Helikalinterpolation	...	`G2, G3`
• Splineinterpolation	...	`ASPLINE, BSPLINE, CSPLINE`
• Polynominterpolation	...	`POLY`

2.5.2 Anwahl der WRK (G41/G42)

Korrekturrichtung

Die WRK ermittelt eine äquidistante Bahn zur programmierten Kontur. Die Korrektur kann in Bewegungsrichtung links oder rechts von der programmierten Kontur erfolgen.

Befehl	Bedeutung
`G41`	WRK links von der Kontur in Bewegungsrichtung
`G42`	WRK rechts von der Kontur in Bewegungsrichtung
`G40`	Abwahl der WRK

Zwischensätze

Während der aktiven WRK werden in der Regel nur Programmsätze mit Positionen von Geometrieachsen in der aktuellen Ebene programmiert. Trotzdem können bei aktiver WRK auch einzelne Zwischensätze programmiert werden. Zwischensätze sind Programmsätze, die nicht mindestens die Position einer Geometrieachse in der aktuellen Ebene enthalten:

- Positionen in der Zustellachse
- Hilfsfunktionen
- etc.

Die max. Anzahl der Zwischensätze kann vorgegeben werden im Maschinendatum:

MD20250 $MC_CUTCOM_MAXNUM_DUMMY_BLOCKS (Satzanzahl ohne Verfahrbewegung bei WRK)

Besonderheiten

- Die Anwahl der WRK kann nur in einem Programmsatz mit `G0` (Eilgang) oder `G1` (Linearinterpolation) erfolgen.
- Spätestens im Programmsatz mit der WRK-Anwahl muss ein Werkzeug eingewechselt (T-Funktion) und die WZ-Schneide/WZK (`D1` bis `D9`) aktiviert werden.
- Bei einer WZ-Schneide/WZK von `D0` erfolgt keine WRK-Anwahl.
- Wird bei der WRK-Anwahl nur eine Geometrieachse der Ebene programmiert, wird die zweite Achse der Ebene automatisch ergänzt (letzte progr. Position).
- Wird im Satz mit der WRK-Anwahl keine Geometrieachse der aktuellen Ebene programmiert, findet keine Anwahl statt.
- Wird im Satz nach der WRK-Anwahl die WRK abgewählt (`G40`), findet keine Anwahl statt.
- Bei Anwahl der WRK wird das Anfahrverhalten durch die Anweisungen `NORM`/`KONT` bestimmt.

2.5.3 An- und Abfahrverhalten (NORM/KONT/KONTC/KONTT)

NORM und KONT

Mit den Anweisungen `NORM` und `KONT` kann das Anfahrverhalten (Anwahl der WRK mit `G41`/`G42`) und das Abfahrverhalten (Abwahl der WRK mit `G40`) gesteuert werden:

Befehl	Bedeutung
`NORM`	Normaleinstellung im Anfangspunkt/Endpunkt (Grundstellung)
`KONT`	Kontur umfahren im Anfangspunkt/Endpunkt
`KONTC`	krümmungsstetiges An-/Abfahren
`KONTT`	tangentenstetiges An-/Abfahren

Besonderheiten

- KONT unterscheidet sich von NORM nur bei einer WZ-Startposition, die hinter der Kontur liegt.

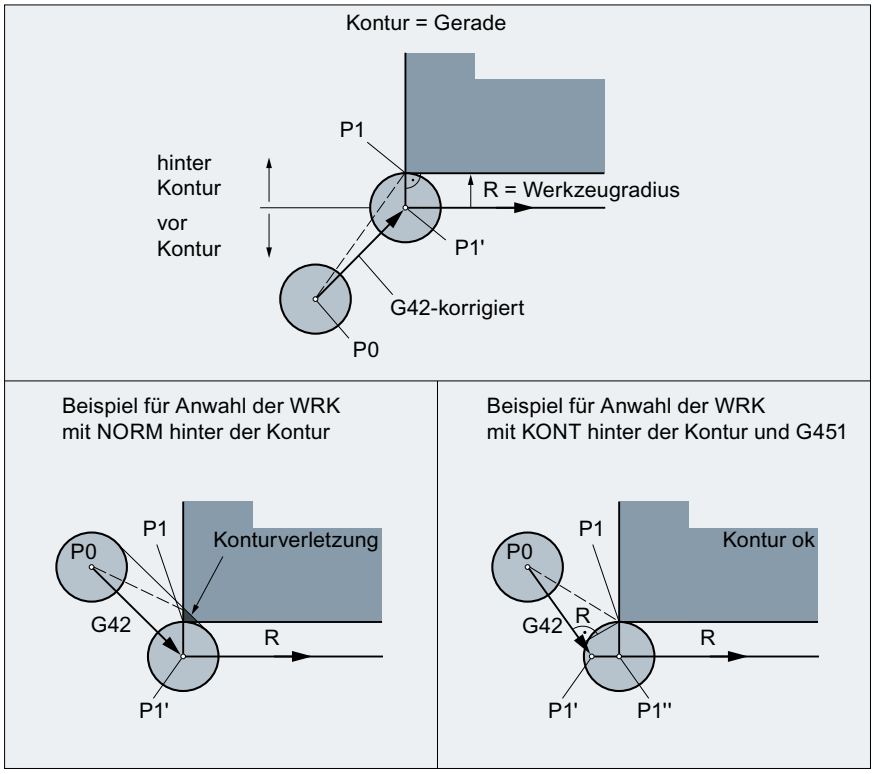


Bild 2-3 Beispiel für Anwahl der WRK mit KONT oder NORM vor und hinter der Kontur

- KONT und G450/G451 (Eckenverhalten an Außenecken) wirkt gemeinsam und bestimmt so das An- Abfahrverhalten bei WRK.
- Bei Abwahl der WRK wird das Abfahrverhalten durch die Anweisungen NORM/KONT bestimmt.

Randbedingungen

Bei den beiden folgenden Varianten sind die An- bzw. Abfahrsätze Polynome. Sie stehen deshalb nur bei Steuerungsvarianten zur Verfügung, bei denen die Polynominterpolation freigegeben ist:

- KONTT
  Mit KONTT wird tangentenstetig an die Kontur angefahren bzw. tangentenstetig von der Kontur abgefahren. Die Krümmungen am Satzübergang sind im Allgemeinen nicht stetig.
- KONTC
  Mit KONTC ist beim Übergang vom Anfahrsatz bzw. zum Abfahrsatz nicht nur die Tangente stetig, sondern auch die Krümmung, so dass in diesem Fall bei der Aktivierung / Deaktivierung **kein Beschleunigungssprung** mehr auftreten kann.

Obwohl KONTC die Eigenschaft KONTT mit umfasst, ist die tangentenstetige Variante KONTT allein verfügbar, weil durch die geforderte Krümmungsstetigkeit bei KONTC mitunter unerwünschte Konturen entstehen können.

Achsen

Die Stetigkeitsbedingung wird in allen **drei** Achsen eingehalten. Es ist somit zulässig, beim An/Abfahren gleichzeitig eine Wegkomponente senkrecht zur Korrekturebene zu programmieren.

Für die originalen An- bzw. Abfahrsätze mit KONTT/KONTC sind nur **Linearsätze** zugelassen. Diese programmierten Linearsätze werden in der Steuerung durch die entsprechenden Polynomkurven ersetzt.

Ausschluss

KONTT und KONTC stehen bei den 3D-Varianten der Werkzeugradiuskorrektur (CUT3DC, CUT3DCC, CUT3DF) nicht zur Verfügung.

Werden sie dennoch programmiert, wird steuerungsintern ohne Fehlermeldung auf NORM umgeschaltet.

Beispiel KONTC

In den beiden folgenden Bildern ist eine typische Anwendung für das krümmungsstetige An- und Abfahren dargestellt:

In der Kreismitte beginnend wird an den Vollkreis angefahren. Dabei sind im Satzendpunkt des Anfahrsatzes dessen Richtung und dessen Krümmungsradius gleich den Werten des folgenden Kreises. In den beiden An-/Abfahrsätzen wird gleichzeitig in Z-Richtung zugestellt.

Das zugehörige NC-Programmsegment sieht folgendermaßen aus:

```
$TC_DP1[1,1]=121                                Fräser
$TC_DP6[1,1]=10                                 Radius 10 mm
N10 G1 X0 Y0 Z60 G64 T1 D1 F10000
N20 G41 KONTC X70 Y0 Z0
N30 G2 I-70                                     Vollkreis
N40 G40 G1 X0 Y0 Z60
N50 M30
```

Erklärung:

In diesem Beispiel wird ein Vollkreis mit dem Radius 70 mm in der X-Y-Ebene bearbeitet. Da das Werkzeug einen Radius von 10 mm hat, ergibt sich als resultierende Werkzeugmittelpunktsbahn ein Kreis mit dem Radius 60 mm. Start-/ Endpunkt liegen bei X0 Y0 Z60, so dass gleichzeitig zur An-/Abfahrbewegung in der Korrekturebene eine Bewegung in Z-Richtung ausgeführt wird.

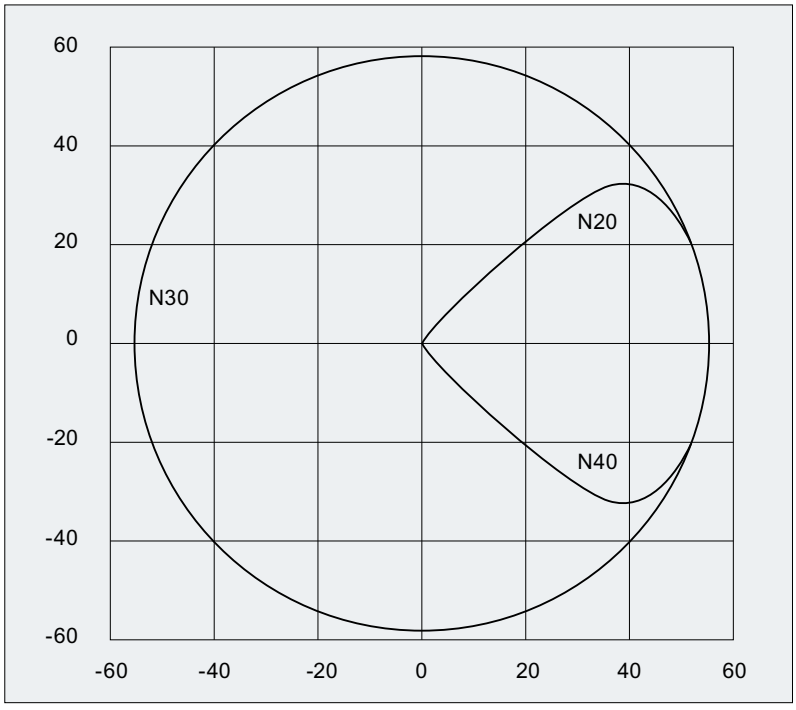


Bild 2-4 Krümmungsstetiges An- und Abfahren bei der Innenbearbeitung eines Vollkreises: Projektion in die X-Y-Ebene

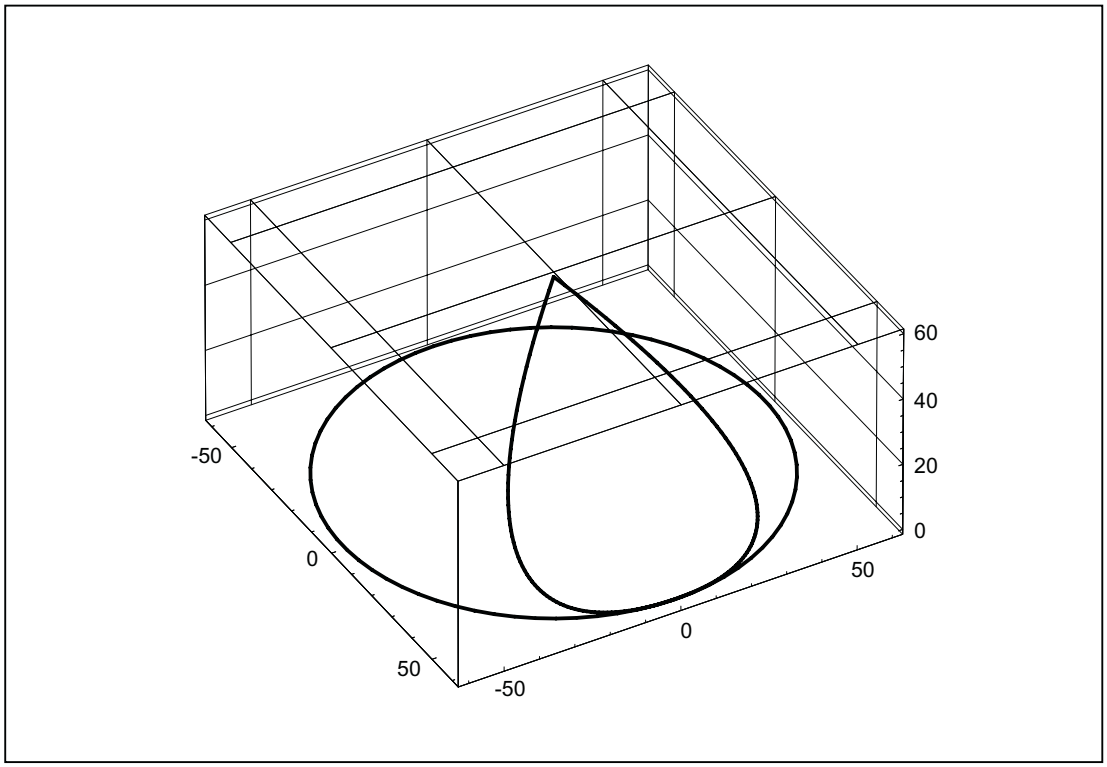


Bild 2-5 Krümmungsstetiges An- und Abfahren bei der Innenbearbeitung eines Vollkreises: Räumliche Darstellung

KONTT und KONTC im Vergleich

Im folgenden Bild ist das unterschiedliche An-/Abfahrverhalten bei KONTT und KONTC dargestellt. Ein Kreis mit dem Radius 20 mm um den Mittelpunkt bei X0 Y-40 wird mit einem Werkzeug mit 20 mm Radius an der Außenseite korrigiert. Es ergibt sich deshalb eine kreisförmige Bewegung des Werkzeugmittelpunkts mit dem Radius 40 mm. Der Endpunkt des Abfahrsatzes liegt bei X40 Y30. Der Übergang zwischen dem Kreissatz und dem Abfahrsatz liegt im Nullpunkt. Wegen der verlangten Krümmungsstetigkeit bei KONTC führt der Abfahrsatz zunächst eine Bewegung mit negativer Y-Komponente aus. Dies wird häufig unerwünscht sein. Der Abfahrsatz mit KONTT zeigt dieses Verhalten nicht. Allerdings tritt in diesem Fall am Satzübergang ein Beschleunigungssprung auf.

Ist der KONTT- bzw. KONTC-Satz nicht der Ab- sondern der Anfahrsatz, ergibt sich exakt die gleiche Kontur, die lediglich in umgekehrter Richtung durchlaufen wird, d.h. **An- und Abfahrverhalten sind symmetrisch.**

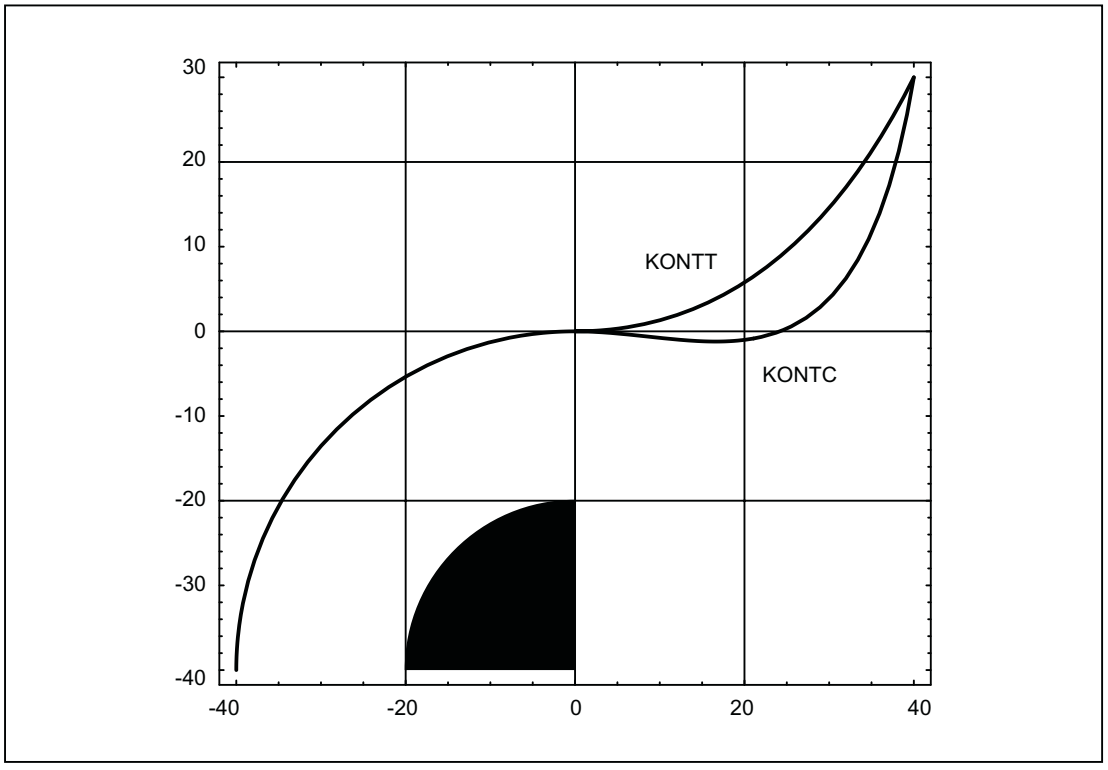


Bild 2-6 Unterschiede von KONTT und KONTC

Hinweis

Aus dem Bild ist ersichtlich, dass eine an den Viertelkreis der Kontur anschließende Gerade z. B. bis X20 Y-20 bei Ab-/Anfahren an X0, Y0 mit KONTC eine Konturverletzung zur Folge hätte.

2.5.4 Weiches An- und Abfahren

2.5.4.1 Funktion

Bedeutung

Die Funktion "Weiches An- und Abfahren (WAB)" dient dazu, im Startpunkt einer Kontur unabhängig von der Lage des Ausgangspunktes tangential anzufahren.

Das Anfahrverhalten kann dabei durch eine Reihe von ergänzenden Parametern variiert und speziellen Erfordernissen angepasst werden.

Die beiden Funktionen "Weiches Anfahren" und "Weiches Abfahren" sind weitgehend symmetrisch zueinander. Im Folgenden wird deshalb nur das Anfahren ausführlich beschrieben und auf Abweichungen beim Abfahren hingewiesen.

Teilbewegungen

Es gibt beim weichen An- und Abfahren maximal 4 Teilbewegungen mit folgenden Positionen:

- Startpunkt der Bewegung P_0
- Zwischenpunkte P_1, P_2, und P_3
- Endpunkt P_4

Die Punkte P_0, P_3 und P_4 sind immer definiert. Die Zwischenpunkte P_1 und P_2 können je nach Parametrierung und den geometrischen Verhältnissen entfallen.

Beim Abfahren werden die Punkte in der umgekehrten Reihenfolge, d. h. beginnend mit P_4 und endend mit P_0 durchlaufen.

2.5.4.2 Parameter

Das Verhalten der Funktion weiches An- und Abfahren wird durch maximal 9 Parameter bestimmt.

Satzweise wirksamer G-Befehl zur Bestimmung der An- bzw. Abfahrkontur

Dieser G-Befehl kann nicht entfallen.

- `G147`: Anfahren mit einer Geraden
- `G148`: Abfahren mit einer Geraden
- `G247`: Anfahren mit einem Viertelkreis
- `G248`: Abfahren mit einem Viertelkreis

- G347: Anfahren mit einem Halbkreis
- G348: Abfahren mit einem Halbkreis

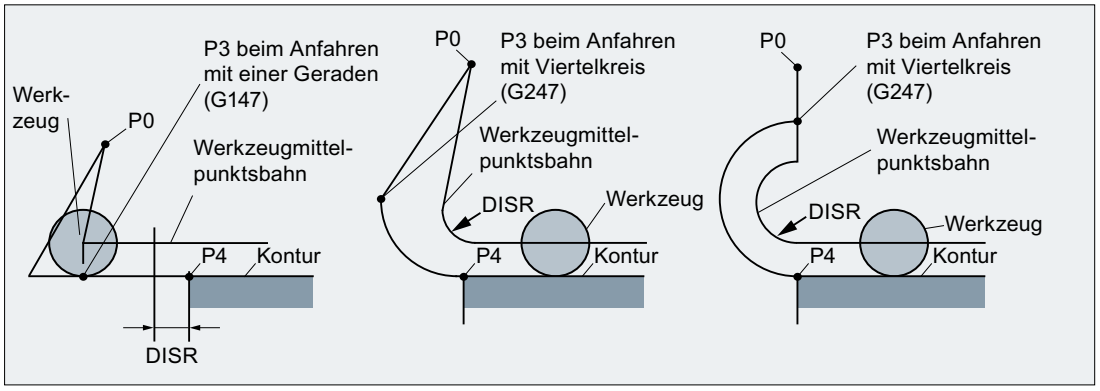


Bild 2-7 Anfahrverhalten in Abhängigkeit von G147 bis G347 und DISR (bei gleichzeitiger Aktivierung der Werkzeugradiuskorrektur)

Modaler G-Befehl zur Bestimmung der An- bzw. Abfahrrichtung

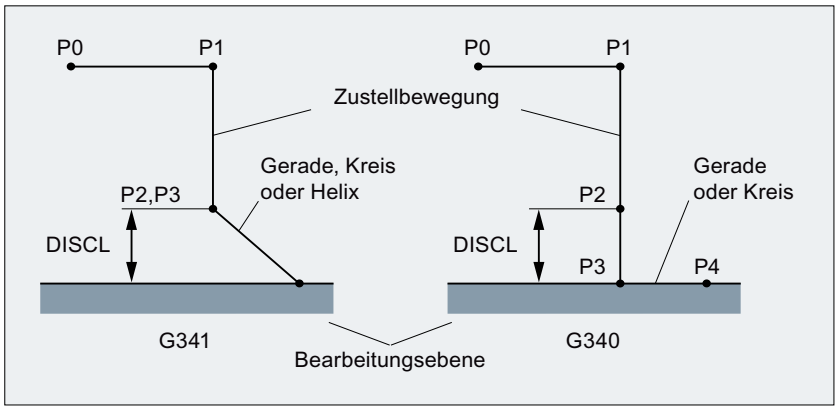
Dieser G-Befehl ist nur dann von Bedeutung, wenn die Anfahrkontur ein Viertel- oder ein Halbkreis ist. Die An- bzw. Abfahrrichtung kann wie folgt bestimmt werden:

- G140:
  Bestimmung der An- bzw. Abfahrrichtung mit Hilfe der aktiven Werkzeugradiuskorrektur. (G140 ist Grundstellungswert.)
  Bei positivem Werkzeugradius:
  - G41 aktiv → anfahren von links
  - G42 aktiv → anfahren von rechts
    Ist keine Werkzeugradiuskorrektur aktiv (G40), ist das Verhalten wie bei G143. Ein Alarm wird in diesem Fall nicht ausgegeben. Ist der Radius des aktiven Werkzeugs 0, so wird die Anfahr- bzw. Abfahrseite so bestimmt, als wäre der Werkzeugradius positiv.
- G141:
  Anfahren von links an die Kontur bzw. nach links von der Kontur wegfahren.
- G142:
  Anfahren von rechts an die Kontur bzw. nach rechts von der Kontur wegfahren.
- G143:
  Automatische Bestimmung der Anfahrrichtung, d. h. es wird von der Seite aus an die Kontur angefahren, auf der der Startpunkt relativ zur Tangente im Startpunkt des Folgesatzes (P_4) liegt.

Hinweis

Beim **Abfahren** wird entsprechend die Tangente im Endpunkt des Vorgängersatzes verwendet. Ist beim Abfahren der Endpunkt nicht programmiert, d. h. er soll implizit bestimmt werden, so ist G143 beim Abfahren nicht zulässig, da zwischen Anfahrseite und Lage des Endpunktes eine wechselseitige Abhängigkeit besteht. Wird G143 in diesem Fall dennoch programmiert, wird ein Alarm ausgegeben. Gleiches gilt, wenn bei aktivem G140 wegen nicht aktiver Werkzeugradiuskorrektur automatisch auf das G143-Verhalten umgeschaltet wird.

Modaler G-Befehl (G340, G341), der die Aufteilung der Bewegung vom Start- zum Endpunkt auf die einzelnen Sätze definiert



`G34 0:` Das charakteristische Anfahren von P_0 bis P_4 ist im Bild dargestellt.

Ist `G247` oder `G347` aktiv (Viertel- oder Halbkreis) und der Startpunkt P_3 liegt nicht in der durch den Endpunkt P_4 definierten Bearbeitungsebene, wird statt eines Kreises eine Helix eingefügt. Der Punkt P_2 ist nicht definiert bzw. er fällt mit P_3 zusammen.

Die Kreisebene bzw. die Helixachse wird dabei durch die im WAB-Satz aktive Ebene (`G17` - `G19`) bestimmt, d. h. vom Folgesatz wird nicht die Starttangente selbst zur Bestimmung des Kreises verwendet, sondern deren Projektion in die aktive Ebene.

Die Bewegung vom Punkt P_0 zum Punkt P_3 erfolgt in zwei Geraden mit der Geschwindigkeit, die vor dem WAB-Satz wirksam war.

`G34 1:` Das charakteristische Anfahren von P_0 bis P_4 ist im Bild dargestellt.

P_3 und P_4 liegen in der Bearbeitungsebene, so dass bei `G247` bzw. `G347` nie eine Helix, sondern immer ein Kreis eingefügt wird.

Bild 2-8 Ablauf der Anfahrbewegung abhängig von `G340/G341`

Hinweis

In allen Fällen, in denen die Lage der aktiven Ebene `G17` - `G19` eingeht (Kreisebene, Helixachse, Zustellbewegungen senkrecht zur aktiven Ebene), wird ein eventuell aktiver drehender Frame berücksichtigt.

DISR

`DISR` gibt die Länge der Anfahrgeraden bzw. den Radius bei Anfahrkreisen an.

An-/Abfahren mit Geraden

Bei An-/Abfahren mit Geraden gibt `DISR` den Abstand der Fräserkante vom Startpunkt der Kontur an, d. h. die Länge der Geraden ergibt sich bei aktiver WRK als Summe von Werkzeugradius und programmierten Wert von `DISR`.

Ein Alarm wird ausgegeben:

- wenn `DISR` negativ ist und der Betrag größer ist als der Werkzeugradius (die Länge der resultierenden Anfahrgeraden ist kleiner oder gleich null).

An-/Abfahren mit Kreisen

Bei An-/Abfahren mit Kreisen gibt DISR immer den Radius der Werkzeugmittelpunktsbahn an. Ist Werkzeugradiuskorrektur aktiviert, wird intern ein Kreis mit einem solchen Radius erzeugt, dass auch in diesem Fall die Werkzeugmittelpunktsbahn aus dem programmierten Radius resultiert.

Ein Alarm wird ausgegeben beim An- bzw. Abfahren mit Kreisen:

- wenn der Radius des intern erzeugten Kreises Null oder negativ ist.
- wenn DISR nicht programmiert ist.
- wenn der Radius einen Wert ≤ 0 hat.

DISCL

DISCL gibt den Abstand des Punktes P2 von der Bearbeitungsebene an.

Soll die Position des Punktes P_2 auf der Achse senkrecht zur Kreisebene absolut angegeben werden, ist der Wert in der Form DISCL = AC(....) zu programmieren.

Ist DISCL nicht programmiert, so sind die Punkte P_1, P_2 und P_3 bei G340 identisch und die Anfahrkontur wird von P_1 zu P_4 gebildet.

Es wird überwacht, dass der durch DISCL definierte Punkt zwischen P_1 und P_3 liegt, d. h. bei allen Bewegungen, die eine Komponente senkrecht zur Bearbeitungsebene haben (Zustellbewegungen, Anfahrbewegung von P_3 nach P_4), muss diese Komponente das gleiche Vorzeichen haben. Eine Richtungsumkehr ist nicht zugelassen. Gegebenenfalls wird ein Alarm ausgegeben.

Bei der Erkennung der Richtungsumkehr wird eine Toleranz zugelassen, definiert durch das Maschinendatum:

MD20204 $MC_WAB_CLEARANCE_TOLERANCE (Richtungsumkehr bei WAB)

Wenn P_2 außerhalb des durch P_1 und P_3 definierten Bereichs liegt, die Abweichung aber kleiner oder gleich der parametrierten Toleranz ist, dann wird angenommen, dass P_2 in der durch P_1 bzw. P_3 definierten Ebene liegt.

Beispiel:

Es wird bei G17 ausgehend von der Position Z=20 des Punktes P_1 angefahren. Die durch P_3 definierte WAB-Ebene ist bei Z=0. Der durch DISCL definierte Punkt muss deshalb zwischen diesen beiden Punkten liegen. MD20204=0.010. Liegt P_2 zwischen 20.000 und 20.010 bzw. zwischen 0 und -0.010, so wird angenommen, es sei der Wert 20.0 bzw. 0.0 programmiert. Der Alarm wird ausgegeben, wenn die Z-Position von P_2 größer 20.010 oder kleiner -0.010 ist.

Abhängig von der relativen Lage von Startpunkt P_0 und Endpunkt P_4 in Bezug auf die Bearbeitungsebene, erfolgen die Zustellbewegungen in negative (Normalfall beim Anfahren) oder in positive (Normalfall beim Abfahren) Richtung, d. h. dass es z. B. bei G17 zulässig ist, dass die Z-Komponente des Endpunkt P_4 größer ist als die des Startpunkts P_0.

Programmierung des Endpunktes P_4 (bzw. P_0 beim Abfahren) in der Regel mit X... Y... Z...

Mögliche Programmierung des Endpunktes P_4 beim Anfahren

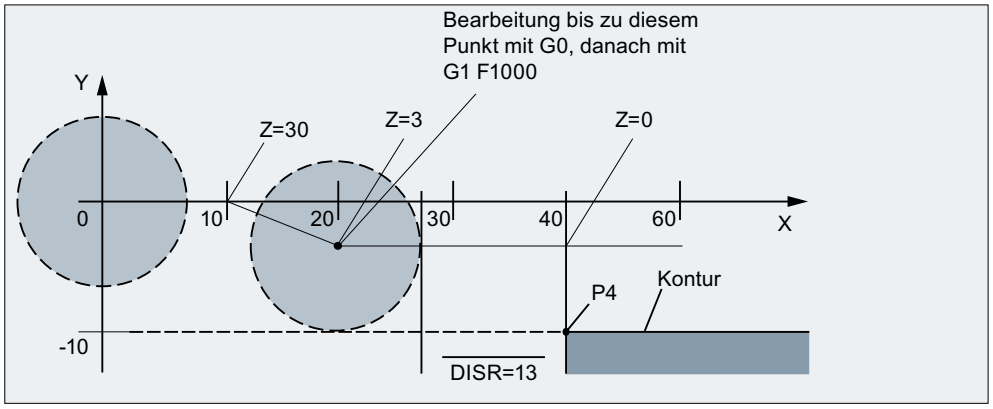
Endpunkt P_4 kann im WAB-Satz selbst programmiert sein.

P_4 kann durch den Endpunkt des nächsten Verfahrsatzes bestimmt werden.

Zwischen dem WAB-Satz und dem nächsten Verfahrsatz können weitere Sätze (Dummysätze) ohne Bewegung der Geometrieachsen eingefügt werden.

Der Endpunkt gilt beim Anfahren dann als im WAB-Satz selbst programmiert, wenn mindestens eine Geometrieachse der Bearbeitungsebene (X oder Y bei `G17`) programmiert wurde. Ist im WAB-Satz nur die Position der Achse senkrecht zur Bearbeitungsebene (Z bei `G17`) programmiert, so wird diese Komponente aus dem WAB-Satz entnommen, die Position in der Ebene dagegen aus dem Folgesatz. Ist in diesem Fall die Achse senkrecht zur Bearbeitungsebene zusätzlich auch im Folgesatz programmiert, wird ein Alarm ausgegeben.

Beispiel:



Programmcode	Kommentar
`$TC_DP1[1,1]=120`	`; Fräserwerkzeug T1/D1`
`$TC_DP6[1,1]=7`	`; Werkzeug mit 7mm Radius`
`N10 G90 G0 X0 Y0 Z30 D1 T1`	
`N20 X10`	
`N30 G41 G147 DISCL=3 DISR=13 Z=0 F1000`	
`N40 G1 X40 Y-10`	
`N50 G1 X50`	
`...`	
`...`	

`N30`/`N40` kann ersetzt werden durch:

```
N30 G41 G147 DISCL=3 DISR=13 X40 Y-10 Z0 F1000
```

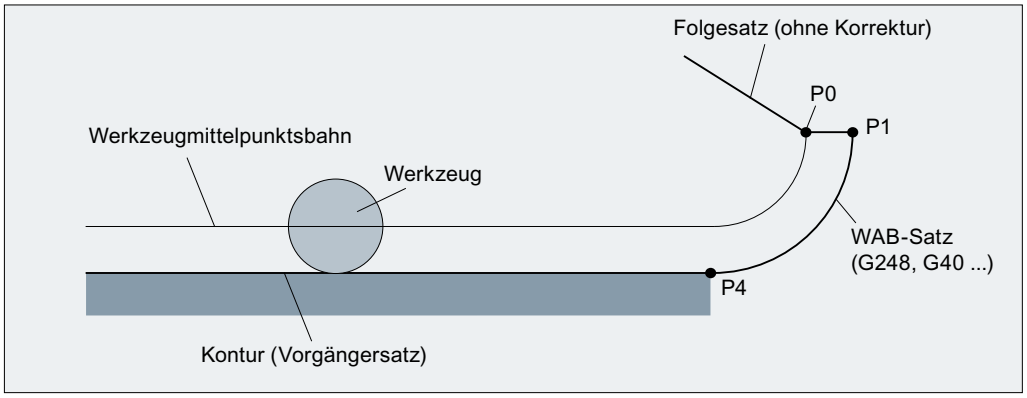
oder:

```
N30 G41 G147 DISCL=3 DISR=13 F1000
N40 G1 X40 Y-10 Z0
```

Mögliche Programmierung des Endpunktes P_0 beim Abfahren

Die Endposition wird immer aus dem WAB-Satz selbst entnommen, unabhängig wie viele Achsen programmiert wurden. Es sind folgende Fälle zu unterscheiden:

1. Im WAB-Satz ist keine Geometrieachse programmiert.
   Die Kontur endet in diesem Fall im Punkt P_2 (bzw. im Punkt P_1 wobei P_1 und P_2 zusammenfallen). Die Position in den Achsen, die die Bearbeitungsebene bilden, ergibt sich aus der Wegfahrkontur (Endpunkt der Geraden bzw. des Kreises). Die Achskomponente senkrecht dazu wird durch `DISCL` definiert. Ist in diesem Fall `DISCL = 0`, verläuft die Bewegung deshalb vollständig in der Ebene.
2. Im WAB-Satz ist nur die Achse senkrecht zur Bearbeitungsebene programmiert.
   Die Kontur endet in diesem Fall im Punkt P_1. Die Position der übrigen beiden Achsen ergibt sich wie unter 1.



   Abfahren mit WAB bei gleichzeitiger Deaktivierung der WRK
   Ist der WAB-Abfahrsatz gleichzeitig der Deaktivierungssatz der Werkzeugradiuskorrektur, so wird in 1. und 2. ein zusätzlicher Weg von P_1 nach P_0 derart eingefügt, dass sich bei der Deaktivierung der Werkzeugradiuskorrektur am Ende der Wegfahrkontur keine Bewegung ergibt, d. h. dieser Punkt definiert dann nicht eine Position auf einer zu korrigierenden Kontur, sondern den Werkzeugmittelpunkt.
3. Es ist mindestens eine Achse der Bearbeitungsebene programmiert.
   Die eventuell fehlende zweite Achse der Bearbeitungsebene wird aus ihrer letzten Position im Vorgängersatz modal ergänzt. Die Position der Achse senkrecht zur Bearbeitungsebene wird - abhängig davon, ob diese Achse programmiert ist oder nicht - wie unter 1. oder 2. gebildet. Die so gebildete Position definiert den Endpunkt P_0.
   Die Abwahl der Werkzeugradiuskorrektur muss nicht gesondert behandelt werden, da der programmierte Punkt P_0 bereits unmittelbar die Position des Werkzeugmittelpunktes am Ende der Gesamtkontur definiert.
   Start- und Endpunkt der WAB-Kontur (P_0 bzw. P_4)dürfen sowohl beim An- als auch beim Abfahren zusammenfallen.

Geschwindigkeit des Vorgängersatzes (typisch G0)

Mit dieser Geschwindigkeit werden alle Bewegungen vom Punkt P_0 bis zum Punkt P_2 ausgeführt, d. h. die Bewegung parallel zur Bearbeitungsebene und der Teil der Zustellbewegung bis zum Sicherheitsabstand.

Programmierung der Vorschubgeschwindigkeit mit FAD

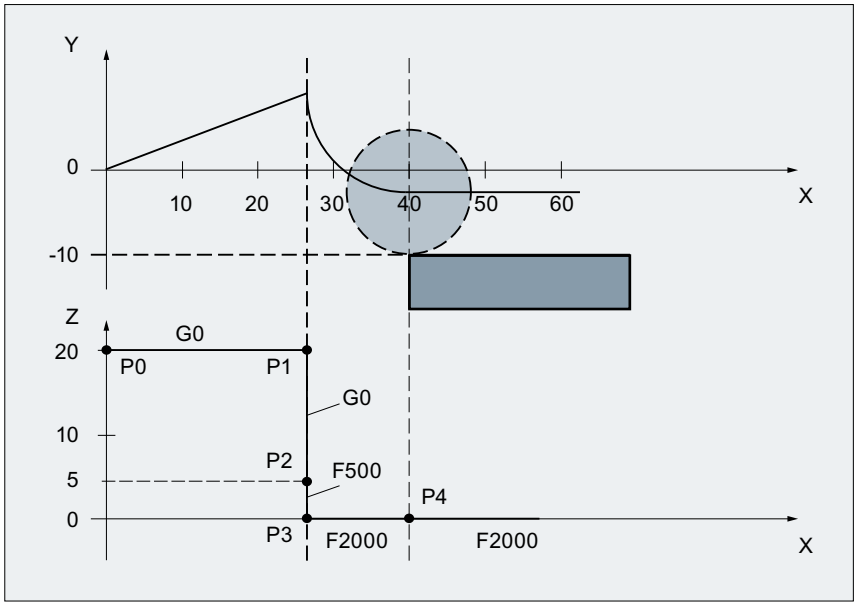
FAD programmiert bei ...	
G340	Vorschubgeschwindigkeit von P_2 bzw. P_3 zu P_4.
G341	Vorschubgeschwindigkeit der Zustellbewegung senkrecht zur Bearbeitungsebene von P_2 nach P_3.

Wird FAD nicht programmiert, wird dieser Teil der Kontur mit der modal wirksamen Geschwindigkeit des Vorgängersatzes verfahren, falls im WAB-Satz kein F-Wort programmiert ist, das die Geschwindigkeit festlegt.

Programmierverhalten:

FAD=0 oder negativ	→	Alarmausgabe
FAD=...	→	programmierter Wert wirkt entsprechend dem aktiven G-Befehl der Gruppe 15 (Vorschubtyp; G93, G94 usw.)
FAD=PM(...)	→	programmierter Wert wird unabhängig vom aktiven G-Befehl der Gruppe 15 als Linearvorschub (wie G94) interpretiert
FAD=PR(...)	→	programmierter Wert wird unabhängig vom aktiven G-Befehl der Gruppe 15 als Umdrehungsvorschub (wie G95) interpretiert

Beispiel:



Programmcode	Kommentar
$TC_DP1[1,1]=120	; Fräserwerkzeug T1/D1
$TC_DP6[1,1]=7	; Werkzeug mit 7mm Radius
N10 G90 G0 X0 Y0 Z20 D1 T1	
N20 G41 G341 G247 DISCL=AC(5) DISR=13FAD 500 X40 Y-10 Z=0 F2000	
N30 X50	
N40 X60	

Programmcode	Kommentar
...	

Programmierung Vorschub F

Dieser Vorschubwert ist ab dem Punkt P_3 (bzw. ab dem Punkt P_2,falls `FAD` nicht programmiert ist) wirksam. Wird im WAB-Satz kein F-Wort programmiert, so wirkt die Geschwindigkeit des Vorgängersatzes. Die durch `FAD` definierte Geschwindigkeit wird für Folgesätze nicht übernommen.

2.5.4.3 Geschwindigkeiten

Geschwindigkeiten beim Anfahren

In den beiden folgenden Darstellungen für das Anfahren wird davon ausgegangen, dass in dem auf den WAB-Satz folgenden Satz keine neue Geschwindigkeit programmiert wurde. Ist dies nicht der Fall, wird ab dem Punkt P_4 die neue Geschwindigkeit wirksam.

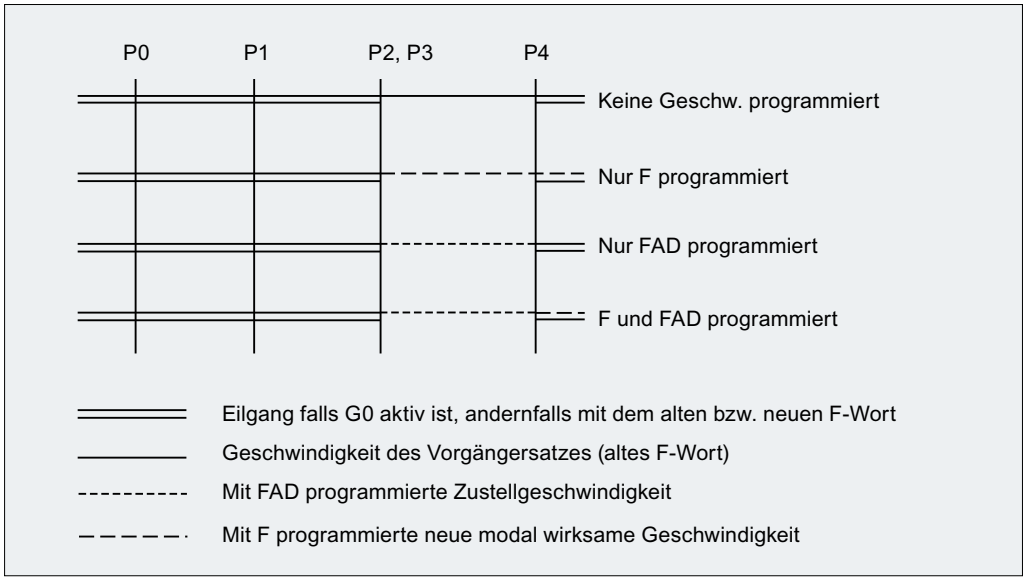


Bild 2-9 Geschwindigkeiten in den WAB-Teilsätzen beim Anfahren mit `G340`

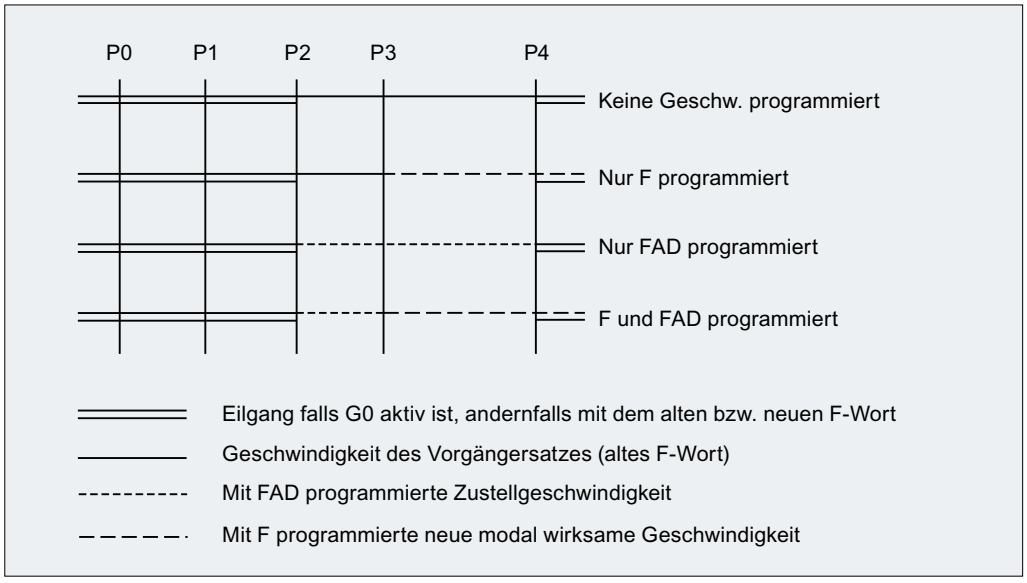


Bild 2-10 Geschwindigkeiten in den WAB-Teilsätzen beim Anfahren mit G341

Geschwindigkeiten beim Abfahren

Beim Abfahren sind die Rollen von modal wirksamem Vorschub aus dem Vorgängersatz und dem im WAB-Satz programmierten Vorschubwert vertauscht, d. h. die eigentliche Abfahrkontur (Gerade, Kreis, Helix) wird mit dem alten Vorschub verfahren, eine neu mit F-Wort programmierte Geschwindigkeit gilt entsprechend ab dem Punkt P_2 bis zum Punkt P_0.

Ist ebenes Wegfahren aktiv und FAD ist programmiert, wird die Strecke P_3 nach P_2 mit FAD verfahren, andernfalls mit der alten Geschwindigkeit. Für die Strecke P_4 nach P_2 gilt immer das letzte in einem Vorgängersatz programmierte F-Wort. G0 wird in diesen Sätzen nicht wirksam.

Von P_2 nach P_0 wird mit dem im WAB-Satz programmierten F-Wort verfahren bzw., falls kein F-Wort programmiert wurde, mit dem modal wirksamen F-Wort aus einem Vorgängersatz. Dies gilt unter der Bedingung, dass G0 nicht aktiv ist.

Sollen beim Abfahren die Sätze von P_2 nach P_0 mit Eilgang verfahren werden, so muss G0 vor dem WAB-Satz oder im WAB-Satz selbst aktiviert werden. Ein eventuell zusätzlich programmiertes F-Wort hat dann in den WAB-Sätzen selbst keine Bedeutung mehr. Es bleibt jedoch modal für Folgesätze wirksam.

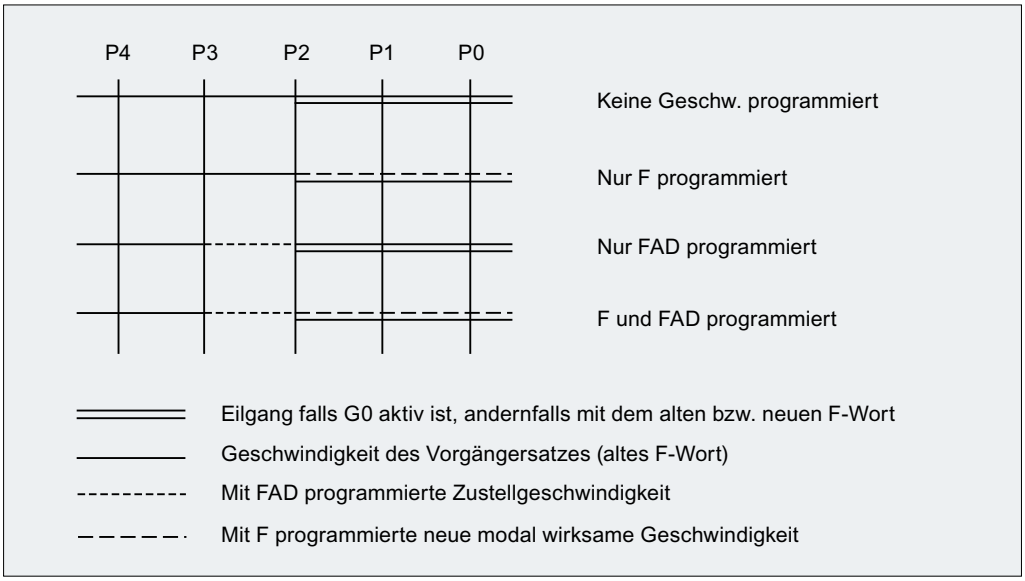


Bild 2-11 Geschwindigkeiten in den WAB-Teilsätzen beim Abfahren

2.5.4.4 Systemvariablen

Die Punkte P_3 und P_4 können beim Anfahren als Systemvariablen im WKS gelesen werden:

$P_APR:	Lesen von P_3 (Aufstartpunkt) im WKS	
$P_AEP:	Lesen von P_4 (Konturanfangspunkt) im WKS	
$P_APDV	=1	Wenn der Inhalt von $P_APR und $P_AEP gültig ist, d. h. wenn diese die zum letzten programmierten WAB-Anfahrsatz gehörenden Positionswerte enthalten.
	=0	Die Positionen eines älteren WAB-Anfahrsatzes werden gelesen.

Eine Änderung des WKS zwischen WAB-Satz und Leseoperation hat keinen Einfluss auf die Positionswerte.

2.5.4.5 Randbedingungen

Randbedingungen

- In einem WAB-Satz können beliebige weitere NC-Befehle programmiert werden (z. B. Hilfsfunktionsausgaben, Synchronachsbewegungen, Positionierachsbewegungen usw.). Diese werden beim Anfahren im ersten und beim Abfahren im letzten Teilsatz ausgeführt.
- Wird der Endpunkt P_4 nicht aus dem WAB-Satz, sondern aus einem nachfolgenden Verfahrsatz entnommen, so wird die eigentliche WAB-Kontur (Gerade, Viertel- oder Halbkreis) in diesem Satz abgefahren.
  Der letzte Teilsatz des ursprünglichen WAB-Satzes enthält dann keine Verfahrinformation für die Geometrieachsen. Er wird aber dennoch stets ausgegeben, da in diesem Satz eventuell weiter Aktionen (z. B. Einzelachsen) ausgeführt werden müssen.

- Es müssen immer mindestens zwei Sätze betrachtet werden:
  - der WAB-Satz selbst
  - der Satz, der die Anfahr- bzw. Abfahrrichtung definiert

  Zwischen diesen beiden Sätzen können weitere Sätze programmiert werden.
  Die Anzahl möglicher Zwischensätze wird begrenzt mit dem Maschinendatum:
  MD20202 $MC_WAB_MAXNUM_DUMMY_BLOCKS (Max. Satzanzahl ohne Verfahrbewegung bei WAB)
- Wird in einem Anfahrsatz gleichzeitig die Werkzeugradiuskorrektur aktiviert, so ist der erste Geradensatz der WAB-Kontur der Aktivierungssatz.
  Die gesamte Kontur, die von der WAB-Funktion erzeugt wurde, wird von der Werkzeugradiuskorrektur so behandelt, als ob sie explizit programmiert worden wäre (Kollisionsüberwachungen, Schnittpunktberechnungen, Anfahrverhalten `NORM` / `KONT`).
- Die Richtung der Zustellbewegung und die Lage der Kreisebene bzw. der Helixachse wird ausschließlich durch die aktive Ebene (`G17` - `G19`) - gegebenenfalls gedreht mit einem aktiven Frame - bestimmt.
- Beim Anfahren darf zwischen dem WAB-Satz und dem Folgesatz, der die Tangentenrichtung definiert, kein Vorlaufstopp eingefügt werden.
  Ein Vorlaufstopp in diesem Fall - gleichgültig ob explizit programmiert oder von der Steuerung automatisch eingefügt - führt zu einem Alarm.

Verhalten bei REPOS

Wird ein WAB-Zyklus unterbrochen und repositioniert, so wird bei `RMIBL` an der Unterbrechungsstelle wieder aufgesetzt. Der Aufsetzpunkt bei `RMEBL` ist der Endpunkt des letzten WAB-Satzes, bei `RMBBL` entsprechend der Startpunkt des ersten WAB-Satzes.

Wird `RMIBL` zusammen mit `DISPR` (Wiederanfahren im Abstand `DISPR` vor dem Unterbrechungspunkt) programmiert, kann der Wiederanfahrpunkt in einem Teilsatz des WAB-Zyklus vor dem Unterbrechungsteilsatz liegen.

2.5.4.6 Beispiele

Beispiel 1

Folgende Bedingungen sind gegeben:

- Weiches Anfahren wird in Satz `N20` aktiviert
- X=40 (Endpunkt); Y=0; Z=0
- Anfahrbewegung erfolgt mit Viertelkreis (`G247`)
- Anfahrrichtung nicht programmiert, es wirkt `G140`, d.h. da WRK aktiv (`G42`) und Korrekturwert positiv ist (10), wird Kontur von rechts angefahren
- Intern erzeugte Anfahrkreis (WAB-Kontur) hat Radius 20, so dass der Radius der Werkzeugmittelpunktsbahn gleich dem programmierten Wert `DISR=10` ist

- Die Anfahrbewegung erfolgt wegen G341 mit einem Kreis in der Ebene, so dass sich der Startpunkt (20, -20, 0) ergibt
- Der Punkt P2 hat wegen DISCL=5 die Position (20, -20, 5) und der Punkt P1 liegt wegen Z30 in N10 bei (20, -20, 30)

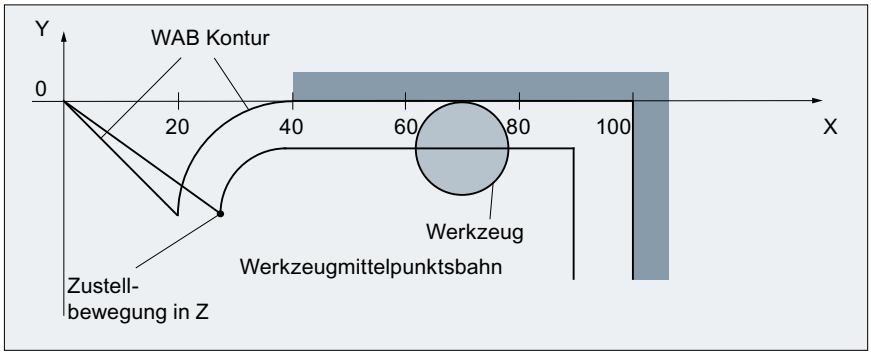


Bild 2-12 Kontur Beispiel 1

Teileprogramm:

Programmcode	Kommentar
$TC_DP1[1,1]=120	; Werkzeugdefinition T1/D1
$TC_DP6[1,1]=10	; Radius
N10 G0 X0 Y0 Z30	
N20 G247 G341 G42 NORM D1 T1 Z0 FAD=1000 F=2000 DISCL=5 DISR=10	
N30 X40	
N40 X100	
N50 Y-30	
...	

Beispiel 2

Folgende Bedingungen sind beim Anfahren gegeben:

- Weiches Anfahren wird in Satz N20 aktiviert
- Anfahrbewegung erfolgt mit Viertelkreis (G247)
- Anfahrrichtung nicht programmiert, es wirkt G140, d. h. da WRK aktiv (G41) ist, wird Kontur von links angefahren
- Konturoffset OFFN=5 (N10)
- aktueller Werkzeugradius=10, damit ist der effektive Korrekturradius für WRK=15; der Radius der WAB-Kontur ist damit gleich 25, so dass der Radius der Werkzeugmittelpunktsbahn gleich DISR=10 wird

- Endpunkt des Kreises ergibt sich aus N30, da in N20 nur Z-Position programmiert ist
- Zustellbewegung
  - von Z20 nach Z7 (DISCL=AC(7)) im Eilgang
  - anschließend nach Z0 mit FAD=200
  - Anfahrkreis in X-Y-Ebene und Folgesätze mit F1500
    (Damit diese Geschwindigkeit in den Folgesätzen wirksam wird, muss der aktive G-Code G0 in N30 mit G1 überschrieben werden. Andernfalls würde die Kontur mit G0 weiter bearbeitet werden.)

Folgende Bedingungen sind beim Abfahren gegeben:

- Weiches Abfahren wird in Satz N60 aktiviert
- Abfahrbewegung erfolgt mit Viertelkreis (G248) und Helix (G340)
- FAD nicht programmiert, da bei G340 ohne Bedeutung
- Z=2 im Startpunkt; Z=8 im Endpunkt, da DISCL=6
- Bei DISR=5 ist Radius der WAB-Kontur=20, der der Werkzeugmittelpunktsbahn=5
- Nach dem Kreissatz folgen die Wegfahrbewegung von Z8 nach Z20 und die Bewegung parallel zur X-Y-Ebene zum Endpunkt bei X70 Y0

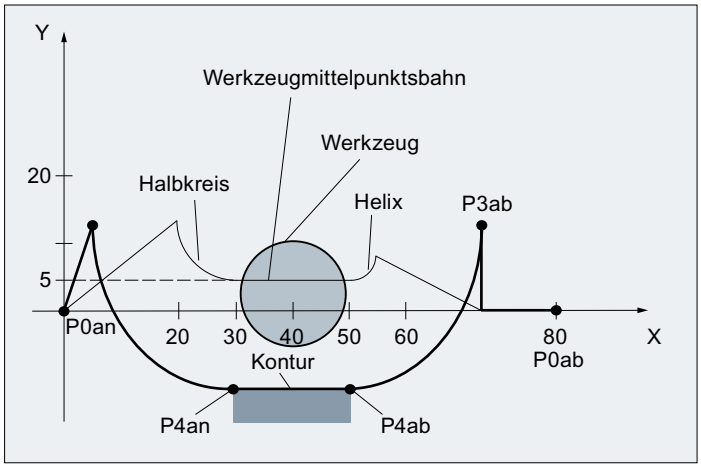


Bild 2-13 Kontur Beispiel 2

Teileprogramm:

Programmcode	Kommentar
$TC_DP1[1,1]=120	; Werkzeugdefinition T1/D1
$TC_DP6[1,1]=10	; Radius
N10 G0 X0 Y0 Z20 G64 D1 T1 OFFN = 5	; (P0an)
N20 G41 G247 G341 Z0 DISCL = AC(7) DISR = 10 F1500 FAD=200	; (P3an)
N30 G1 X30 Y-10	; (P4an)
N40 X40 Z2	
N50 X50	; (P4ab)
N60 G248 G340 X70 Y0 Z20 DISCL = 6 DISR = 5 G40 F10000	; (P3ab)
N70 X80 Y0	; (P0ab)

Programmcode	Kommentar
N80 M 30	

Hinweis

Die derart erzeugte Kontur wird durch die Werkzeugradiuskorrektur modifiziert, die im WAB-Anfahrsatz aktiviert und im WAB-Abfahrsatz deaktiviert wird.

Die Werkzeugradiuskorrektur berücksichtigt einen effektiven Radius von 15, der sich aus der Summe von Werkzeugradius (10) und Konturoffset (5) zusammensetzt. Der resultierende Radius der Werkzeugmittelpunktsbahn im Anfahrsatz ist deshalb 10, im Abfahrsatz 5.

2.5.5 Abwahl der WRK (G40)

Anweisung G40

Die Abwahl der WRK erfolgt mit der Anweisung `G40`.

Besonderheiten

- Die Abwahl der WRK kann nur in einem Programmsatz mit `G0` (Eilgang) oder `G1` (Linearinterpolation) erfolgen.
- Wird bei aktiver WRK `D0` programmiert, dann erfolgt keine Abwahl, sondern die Fehlermeldung 10750.
- Wird im Satz mit der WRK-Abwahl eine Geometrieachse programmiert, findet eine Abwahl der WRK statt, auch wenn diese nicht in der aktuellen Ebene liegt.

2.5.6 Korrektur an Außenecken

G450/G451

Mit den G-Befehlen `G450`/`G451` kann das Verhalten bei unstetigen Satzübergängen an Außenecken gesteuert werden:

Befehl	Bedeutung
`G450`	Unstetige Satzübergänge mit Übergangskreis
`G451`	Unstetige Satzübergänge mit Schnittpunkt der Äquidistanten

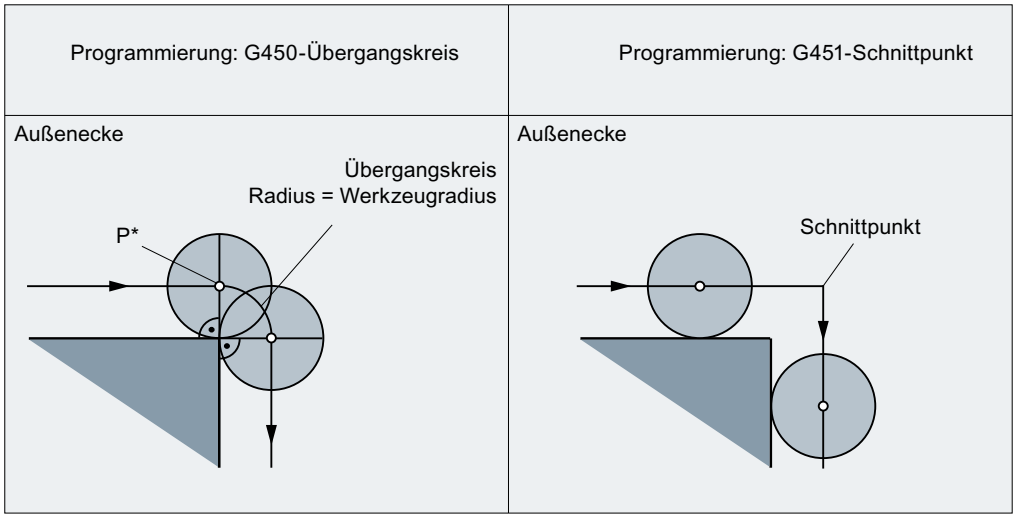


Bild 2-14 Beispiel für 90°-Außenecke mit G450 und G451

G450 (Übergangskreis)

Bei aktivem G-Befehl G450 vollzieht der Mittelpunkt des Werkzeugs bei Außenecken eine Kreisbahn mit dem Werkzeugradius. Die Kreisbahn beginnt mit der Normalenstellung (senkrecht zur Bahntangente) im Endpunkt des vorherigen Bahnstückes (Programmsatzes) und endet in Normalenstellung im Anfangspunkt des neuen Bahnstückes (Programmsatzes).

Bei sehr flachen Außenecken nähert sich das Verfahren mit G450 (Übergangskreis) und G451 (Schnittpunkt) immer mehr an (siehe Abschnitt "Sehr flache Außenecken").

Sollen spitze Außenkonturecken entstehen, muss das Werkzeug von der Kontur abgehoben werden (siehe Abschnitt "DISC").

DISC

Mit G450 - Übergangskreis können keine scharfen Außenkonturecken entstehen, weil durch den Übergangskreis die Werkzeugmittelpunktsbahn so geführt wird, dass die WZ-Schneide auf der Außenecke (programmierte Position) stillsteht. Sollen mit G450 trotzdem scharfe Außenecken bearbeitet werden, kann mit der Anweisung DISC im Programm eine Überhöhung programmiert werden. Dadurch wird aus dem Übergangskreis ein Kegelschnitt und die WZ-Schneide hebt von der Außenecke ab.

Der Wertebereich der Anweisung DISC beträgt 0 bis 100 in Schritten von 1.

Wert	Bedeutung
DISC = 0	Überhöhung abgeschaltet, Übergangskreis wirksam
DISC = 100	Überhöhung so groß, dass sich theoretisch ein Verhalten wie bei Schnittpunkt (G451) ergibt.

Der programmierbare Maximalwert für DISC kann eingestellt werden über das Maschinendatum:

MD20220 $MC_CUTCOM_MAX_DISC (max. Wert für DISC)

Sinnvolle Werte für `DISC` liegen in der Regel nicht über 50.

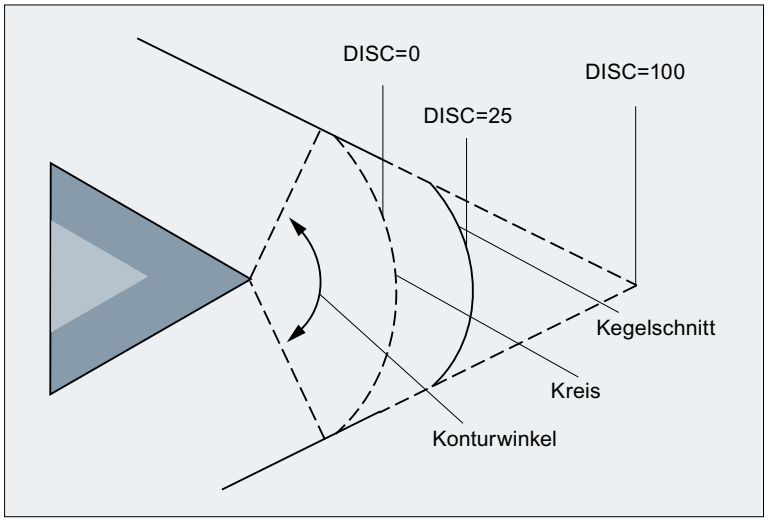


Bild 2-15 Beispiel: Überhöhung mit `DISC`=25

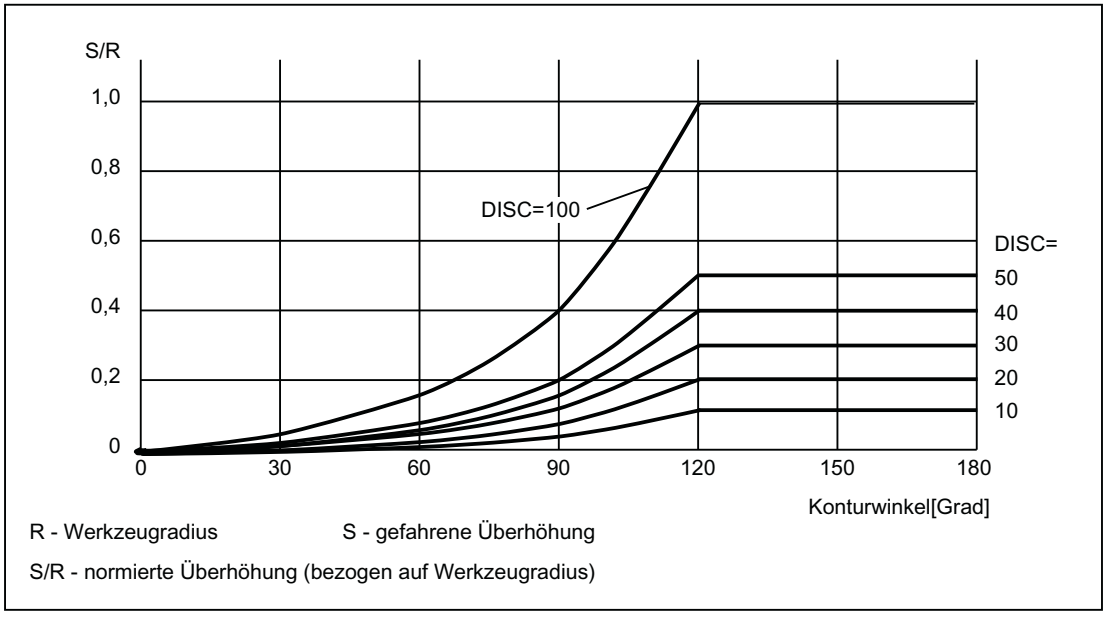


Bild 2-16 Überhöhung mit `DISC` abhängig von Konturwinkel

G451 (Schnittpunkt)

Bei aktivem G-Befehl `G451` wird die Position (der Schnittpunkt) angefahren, die sich aus den Bahnlinien (nur bei Gerade, Kreis oder Helix) ergibt, die sich im Abstand des Werkzeugradius zur programmierten Kontur befinden (Mittelpunktsbahn des Werkzeugs). Splines und Polynome werden grundsätzlich nicht verlängert.

Sehr spitze Außenecken

Bei sehr spitzen Außenecken kann es mit G451 zu langen Leerwegen kommen. Deshalb wird bei sehr spitzen Außenecken automatisch von G451 (Schnittpunkt) auf G450 (Übergangskreis, ggf. mit DISC) umgeschaltet.

Der Grenzwinkel (Konturwinkel) für diese automatische Umschaltung (Schnittpunkt → Übergangskreis) kann vorgegeben werden im Maschinendatum:

MD20210 $MC_CUTCOM_CORNER_LIMIT (Maximalwinkel für Ausgleichssätze bei WRK)

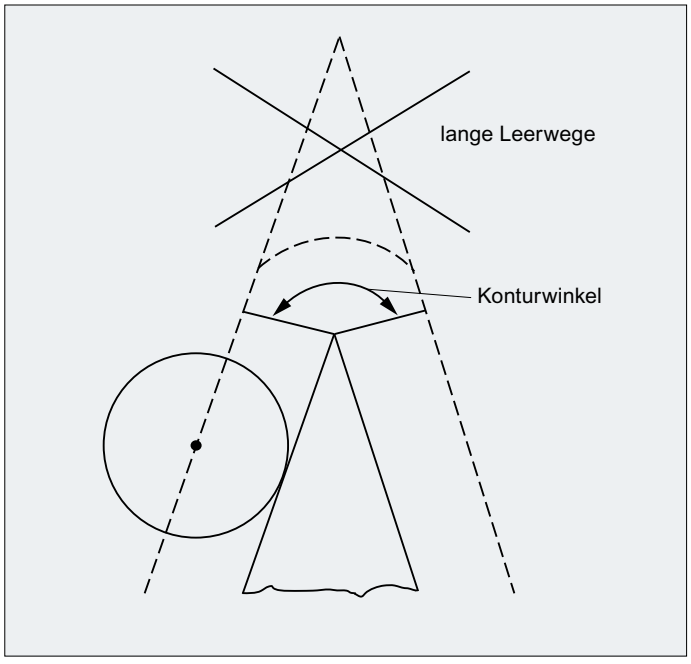


Bild 2-17 Beispiel für automatische Umschaltung auf Übergangskreis

Sehr flache Außenecken

Bei sehr flachen Außenecken nähern sich die Verfahren mit G450 (Übergangskreis) und G451 (Schnittpunkt) immer mehr an. In diesem Fall macht es keinen Sinn mehr, einen Übergangskreis einzufügen. Insbesondere bei der 5-Achsbearbeitung darf an diesen Außenecken kein Übergangskreis eingefügt werden, weil es sonst im Bahnsteuerbetrieb (G64) zu Geschwindigkeitseinbußen kommt. Deshalb wird bei sehr flachen Außenecken automatisch von G450 (Übergangskreis, ggf. mit DISC) auf G451 (Schnittpunkt) umgeschaltet.

Der Grenzwinkel (Konturwinkel) für diese automatische Umschaltung (Übergangskreis → Schnittpunkt) kann vorgegeben werden im Maschinendatum:

MD20230 $MC_CUTCOM_CURVE_INSERT_LIMIT (Maximalwinkel für Schnittpunktberechnung bei WRK)

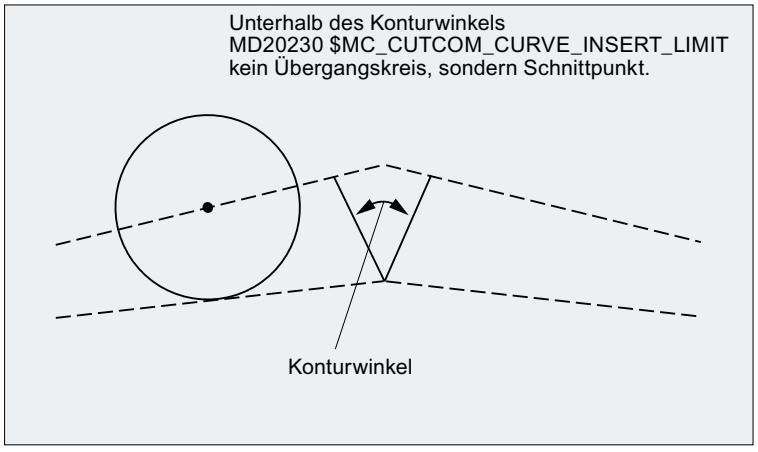
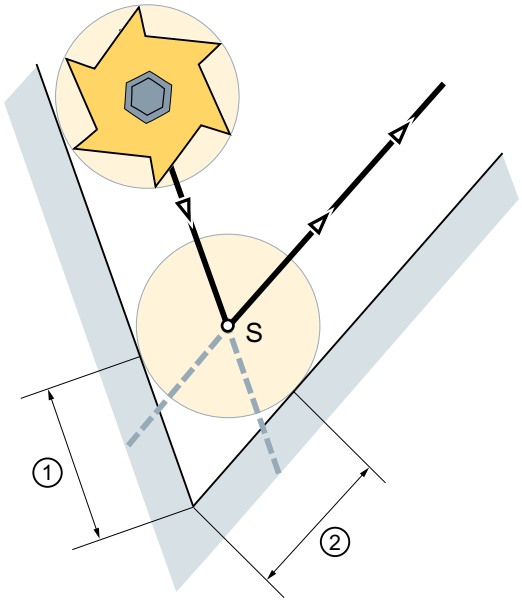


Bild 2-18 Beispiel für automatische Umschaltung auf Schnittpunkt

2.5.7 Korrektur an Innenecken

Schnittpunkt

Bilden zwei aufeinanderfolgende Sätze eine Innenecke, so wird versucht, einen Schnittpunkt der beiden Äquidistanten zu finden. Wird ein Schnittpunkt gefunden, wird die programmierte Kontur bis zum Schnittpunkt verkürzt:

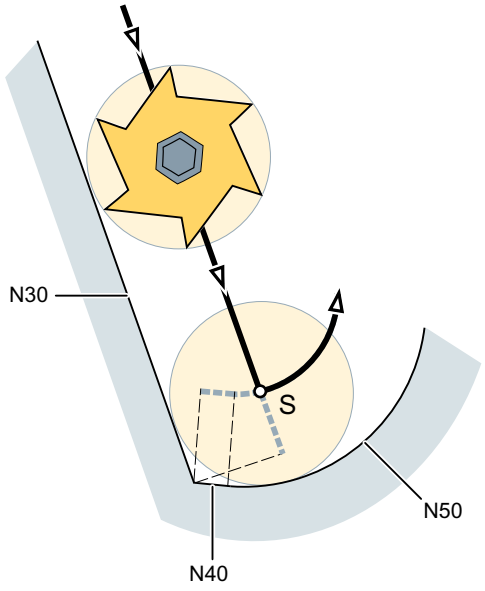


S: Schnittpunkt
① Erster Satz am Ende verkürzt.
② Zweiter Satz am Anfang verkürzt.

Bild 2-19 Programmierte Kontur wird verkürzt

Kein Schnittpunkt

Bei Innenecken kann der Fall eintreten, dass zwischen zwei aufeinander folgenden Sätzen kein Schnittpunkt gefunden wird. In solchen Fällen betrachtet die Steuerung automatisch den nächsten Satz und versucht, mit der Äquidistanten dieses Satzes einen Schnittpunkt zu finden:



S: Schnittpunkt

Bild 2-20 Vorausschauende Konturberechnung

Diese automatische Betrachtung der nächsten Sätze, die vorausschauende Konturberechnung, wird immer so lange durchgeführt, bis eine per Maschinendatum einstellbare Satzanzahl erreicht ist:

MD20240 $MC_CUTCOM_MAXNUM_CHECK_BLOCKS (Sätze für vorausschauende Konturberechnung bei WRK)

Wird innerhalb der vorausschauend betrachteten Sätze kein Schnittpunkt gefunden, bleibt die Programmbearbeitung stehen und ein Alarm wird gesetzt.

Mehrere Schnittpunkte

Bei Innenecken kann auch der Fall eintreten, dass die vorausschauende Konturberechnung mehrere Schnittpunkte der Äquidistanten in mehreren aufeinander folgenden Sätzen findet. Dabei wird immer der letzte Schnittpunkt als gültiger Schnittpunkt festgelegt. Die vorherigen Schnittpunkte werden nicht angefahren:

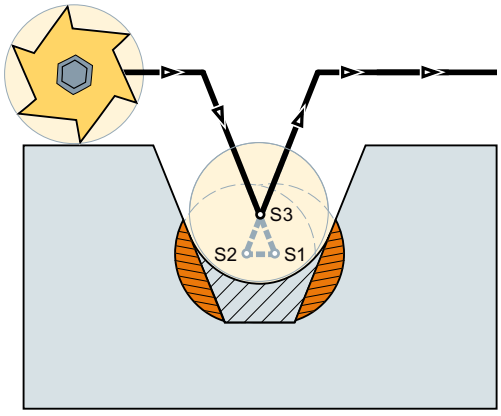


Bild 2-21 In diesem Beispiel wird die Tasche wird nur soweit ausgeräumt, wie es ohne Konturverletzung möglich ist.

Für weitere Informationen siehe auch Kapitel "Kollisionsüberwachung ("Flaschenhalserkennung") (Seite 77)".

Besonderheiten

Werden mehrere Schnittpunkte mit dem nächsten Satz gefunden, ist der Schnittpunkt gültig, der am nächsten beim Startpunkt des ersten Satzes liegt.

2.5.8 Kollisionsüberwachung ("Flaschenhalserkennung")

2.5.8.1 Funktion

Mit der Kollisionsüberwachung ("Flaschenhalserkennung") wird bei aktiver Werkzeugradiuskorrektur durch vorausschauende Konturberechnung überprüft, ob sich die Äquidistanten von **nicht benachbarten** Sätzen schneiden. Hierdurch lassen sich mögliche Kollisionen rechtzeitig erkennen und aktiv durch die Steuerung verhindern.

Die maximale Anzahl an Sätzen, die dabei vorausschauend betrachtet werden, ist durch ein kanalspezifisches Maschinendatum einstellbar (siehe "Parametrierung (Seite 78)").

Wird ein Schnittpunkt gefunden, werden die dazwischen liegenden Bewegungen in der Korrekturebene nicht ausgeführt. Es wird Alarm 10763 angezeigt:

"Die Bahnkomponente des Satzes in der Korrekturebene wird Null."

Das NC-Programm wird nicht unterbrochen. Die Engstelle ("Flaschenhals") wird umfahren. Alle anderen in den ausgelassenen Sätzen enthaltenen Bewegungen und ausführbaren Anweisungen (M-Befehle, Verfahren von Positionierachsen usw.) werden an der Position des

letzten gefundenen Schnittpunkts in der Reihenfolge ausgeführt, in der sie im NC-Programm enthalten sind.

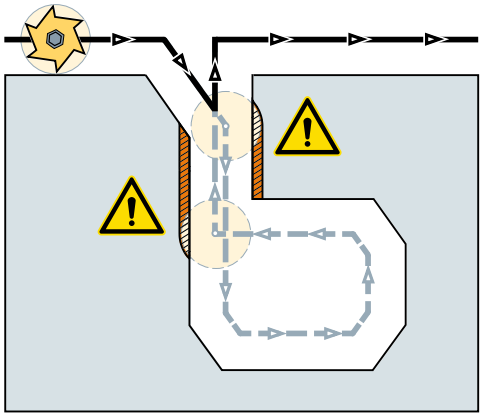


Bild 2-22 Umfahren einer Engstelle ("Flaschenhals") bei aktiver Kollisionsüberwachung (8 Sätze vorausschauend)

Aktivierung / Deaktivierung

Die Aktivierung / Deaktivierung der Funktion erfolgt im NC-Programm über die Befehle der G-Gruppe 23.

Siehe "Programmierung (Seite 78)".

2.5.8.2 Parametrierung

Maximale Anzahl von Sätzen für vorausschauende Konturberechnung

Die maximale Anzahl von Sätzen, die vorausschauend betrachtet werden sollen, wird eingestellt über:

MD20240 $MC_CUTCOM_MAXNUM_CHECK_BLOCKS (Sätze für vorausschauende Konturberechnung bei WRK)

Alarm 10763 unterdrücken

Die Anzeige von Alarm 10763 (Seite 77) kann durch folgende Einstellung unterdrückt werden:

MD11410 $MN_SUPPRES_ALARM_MASK (Maske zur Unterstützung spezieller Alarmausgaben), Bit 1 = 1

2.5.8.3 Programmierung

Die Kollisionsüberwachung ("Flaschenhalserkennung") bei aktiver WRK wird im NC-Programm mit den Befehlen der G-Gruppe 23 ein- bzw. ausgeschaltet.

Syntax

```
G41/G42 CDON
...
CDOF/CDOF2
```

Bedeutung

CDON:	Kollisionsüberwachung ("Flaschenhalserkennung") einschalten Mit CDON wird über eine einstellbare (MD20240 (Seite 78)) Anzahl an Sätzen hinweg überprüft, ob sich die Werkzeugwege von **nicht benachbarten** Sätzen schneiden. Hierdurch lassen sich mögliche Kollisionen rechtzeitig erkennen und aktiv durch die Steuerung verhindern.
CDOF:	Kollisionsüberwachung ("Flaschenhalserkennung") ausschalten Mit CDOF wird für den aktuellen Satz beim **vorhergehenden** Verfahrsatz (an Innenecken) nach einem gemeinsamen Schnittpunkt gesucht, gegebenenfalls auch in weiter zurückliegenden Sätzen. Wird ein Schnittpunkt gefunden, werden keine weiteren Sätze betrachtet. Bei Außenecken wird zwischen zwei aufeinander folgenden Sätzen immer ein Schnittpunkt gefunden. **Hinweis:** Mit CDOF lässt sich die fehlerhafte Erkennung von Engstellen vermeiden, die z. B. auf fehlende Informationen im NC-Programm zurückzuführen sind.
CDOF2:	Kollisionsüberwachung für das 3D-Umfangsfräsen ausschalten Mit CDOF2 wird die Werkzeugkorrekturrichtung aus benachbarten Satzteilen ermittelt. CDOF2 wirkt nur beim 3D-Umfangsfräsen und hat bei allen anderen Bearbeitungsarten (z. B. 3D-Stirnfräsen) die gleiche Bedeutung wie CDOF.

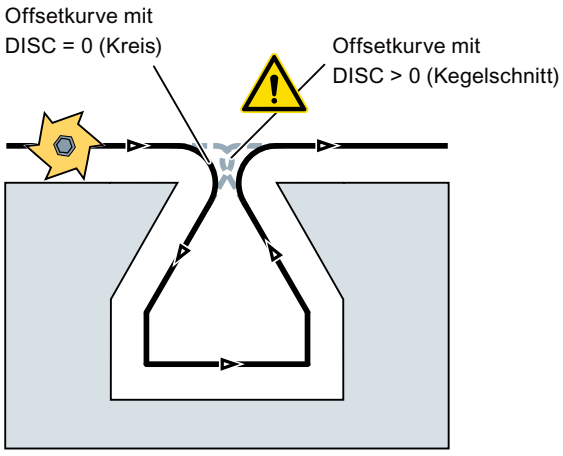
2.5.8.4 Randbedingungen

Programmtest

Um Programmstopps zu vermeiden, sollte beim Programmtest aus der Reihe der eingesetzten Werkzeuge immer das Werkzeug mit dem größten Radius verwendet werden.

"Flaschenhalserkennung" durch Überhöhung an Außenecken

Bei der Schnittpunktüberprüfung nicht benachbarter Sätze werden nicht die programmierten Originalkonturen überprüft, sondern die zugehörigen berechneten Äquidistanten. Dies kann dazu führen, dass bei Außenecken ein "Flaschenhals" erkannt wird, obwohl das tatsächlich nicht der Fall ist. Die Ursache hierfür liegt darin, dass bei DISC > 0 die berechnete Werkzeugbahn nicht äquidistant zur programmierten Originalkontur verläuft.



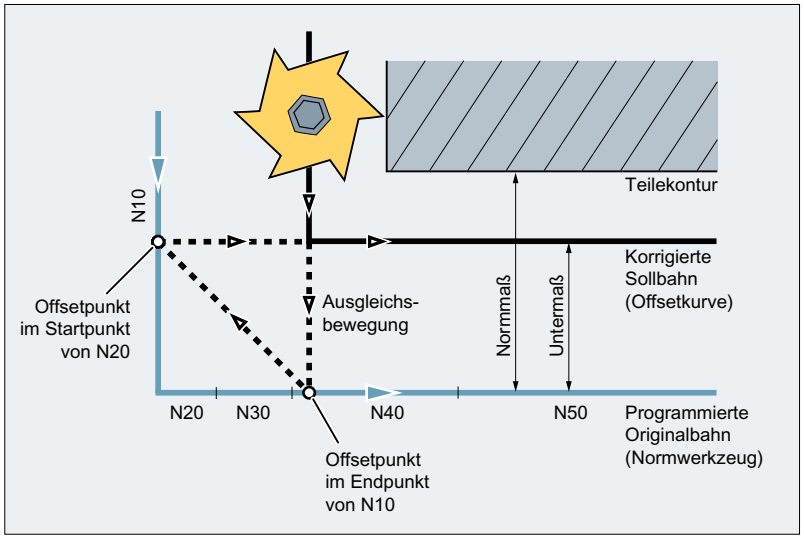
2.5.8.5 Beispiel

Auswirkung der Kollisionsüberwachung an einem Beispiel

Das NC-Programm beschreibt die Mittelpunktsbahn eines Normwerkzeugs. Die Kontur für ein aktuell verwendetes Werkzeug ergibt ein Untermaß, welches im folgenden Bild zur Verdeutlichung der geometrischen Verhältnisse unrealistisch groß dargestellt ist.

Außerdem soll für das Beispiel gelten, dass die Steuerung nur drei Sätze überblickt:

MD20240 $MC_CUTCOM_MAXNUM_CHECK_BLOCKS = 3



Da ein Schnittpunkt nur zwischen den Offsetkurven der beiden Sätze N10 und N40 existiert, müssten die beiden Sätze N20 und N30 ausgelassen werden. Im Beispiel ist der Steuerung der Satz N40 noch nicht bekannt, wenn N10 abschließend bearbeitet werden muss. Damit kann nur ein einzelner Satz ausgelassen werden.

Bei aktiven CDOF2 wird die im Bild dargestellte Ausgleichsbewegung ausgeführt und nicht angehalten. In dieser Situation würde ein aktives CDOF oder CDON zu einem Alarm führen.

2.5.9 Sätze mit veränderlichem Korrekturwert

Randbedingungen

Ein veränderlicher Korrekturwert ist bei allen Interpolationsarten zulässig (auch Kreis- und Spline-Interpolation).

Ebenso ist ein Wechsel des Vorzeichens (und damit ein Wechsel der Korrekturseite) zulässig.

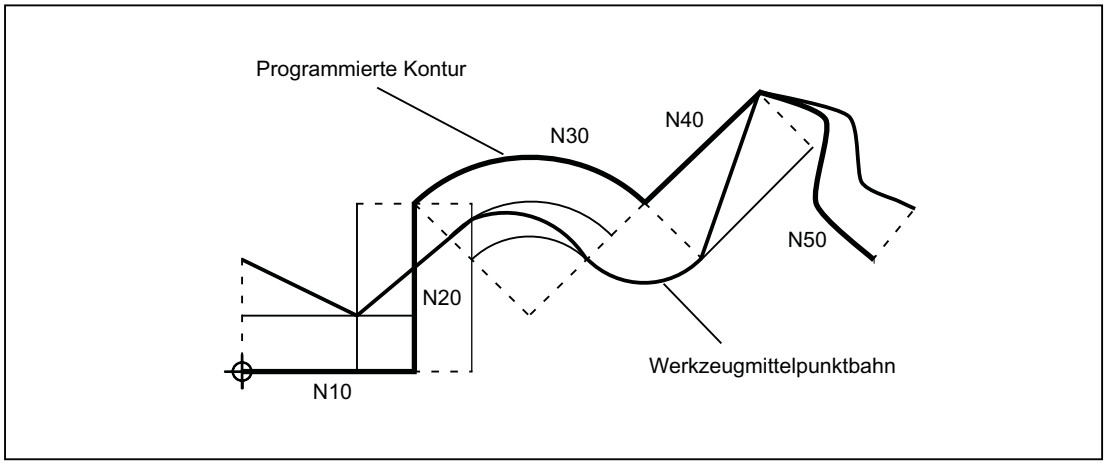


Bild 2-23 Werkzeugradiuskorrektur bei veränderlichem Korrekturwert

Schnittpunktberechnung

Bei der Schnittpunktberechnung in Sätzen mit veränderlichem Korrekturwert wird der Schnittpunkt der Offsetkurven (Werkzeugbahnen) immer mit der Annahme eines konstanten Korrekturwertes berechnet.

Ist der Satz mit dem veränderlichen Korrekturwert der erste der beiden zu betrachtenden Sätze in Verfahrrichtung, so wird der Korrekturwert am Satzende zur Berechnung herangezogen, andernfalls der Korrekturwert am Satzanfang.

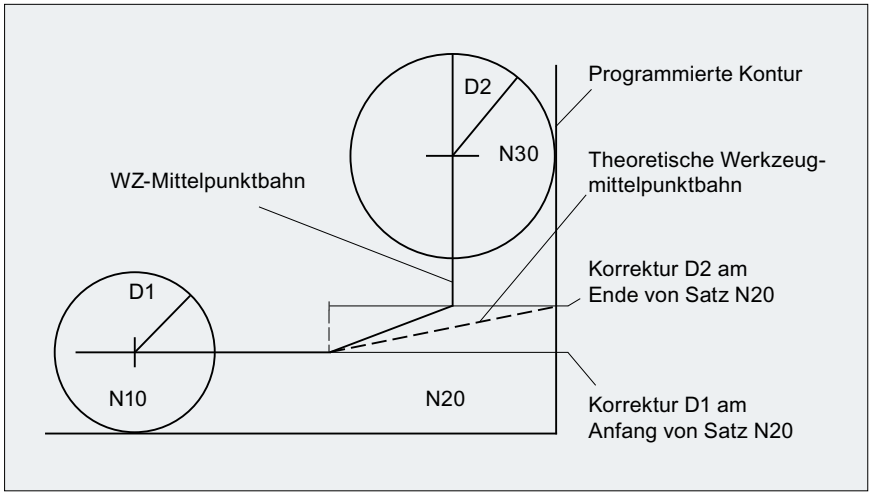


Bild 2-24 Schnittpunktberechnung bei veränderlichem Korrekturwert

Einschränkung

Wird bei der Bearbeitung an der Kreisinnenseite der Korrekturradius größer als der programmierte Kreisradius, so wird die Bearbeitung mit dem folgenden Alarm abgelehnt:

Alarm 10758 "Krümmungsradius mit veränderlichem Korrekturwert zu klein"

Geschlossene Kontur stabil erhalten

Wird ein Radius zweier Kreise geringfügig erhöht, kann dies einen dritten Satz erfordern, damit die geschlossene Kontur stabil erhalten bleibt. Dies ist dann der Fall, wenn zwei benachbarte Sätze, die zwei mögliche Schnittpunkte für eine geschlossene Kontur bilden, aufgrund der Korrektur übersprungen werden.

Eine stabil geschlossene Kontur kann dadurch erreicht werden, dass nicht der zweite mögliche Schnittpunkt gewählt wird, sondern der erste Schnittpunkt:

SD42496 $SC_CUTCOM_CLSD_CONT ≠ 0 (Verhalten der WRK bei geschlossener Kontur)

Damit wird im Falle des Überspringens dieses Satzes immer der zweite mögliche Schnittpunkt erreicht. Ein dritter Satz wird damit überflüssig.

2.5.10 Alarmverhalten

Alarm im Vorlauf

Tritt im Vorlauf ein Alarm der Werkzeugradiuskorrektur auf, wird die Bearbeitung im Hauptlauf am nächsten erreichbaren Satzende angehalten, d. h. in der Regel am Ende des aktuell interpolierten Satzes (bei aktivem Look-Ahead dann, wenn die Achsen zum Stillstand gekommen sind).

Alarme bei Vorlaufstopp und aktiver Werkzeugradiuskorrektur

Die Werkzeugradiuskorrektur benötigt zur Bestimmung des Endpunktes eines Satzes in der Regel mindestens einen nachfolgenden Verfahrsatz (bei Flaschenhälsen auch mehr). Da bei aktivem Vorlaufstopp ein solcher Satz nicht zur Verfügung steht, wird deshalb auf den Offsetpunkt im letzten Satz verfahren. Entsprechend wird im ersten Satz nach einem Vorlaufstopp der Offsetpunkt im Startpunkt angefahren.

Die Kontur, die sich dabei ergibt, kann von der Kontur, die sich ohne Vorlaufstopp ergeben würde, erheblich abweichen. Insbesondere sind Konturverletzungen möglich. Es wurde deshalb folgendes Settingdatum eingeführt:

SD42480 $MC_STOP_CUTCOM_STOPRE (Alarmreaktion bei WRK und Vorlaufstopp)

Abhängig von dessen Wert bleibt das Verhalten der Werkzeugradiuskorrektur gegenüber dem bisherigen Stand unverändert, bzw. wird bei einem Vorlaufstopp während aktiver Werkzeugradiuskorrektur ein Alarm ausgegeben und die Programmbearbeitung angehalten.

Der Bediener kann diesen Alarm quittieren und das NC-Programm mit NC-Start unverändert fortsetzen oder mit `RESET` abbrechen.

2.5.11 Schnittpunktverfahren für Polynome

Funktion

Bilden zwei Kurven bei aktiver Werkzeugradiuskorrektur eine Außenecke, wird abhängig vom G-Befehl der 18. Gruppe (Eckenverhalten bei Werkzeugkorrektur; `G450` / `G451`) und unabhängig vom Typ der beteiligten Kurven (Geraden, Kreise, Polynome):

- ein Kegelschnitt zum Umfahren der Ecke eingefügt
  oder
- die beteiligten Kurven werden so extrapoliert, dass sie einen Schnittpunkt bilden

Wird bei aktivem `G451` kein Schnittpunkt gefunden oder ist der Winkel, den die beiden beteiligten Kurven bilden, zu spitz, wird automatisch auf das Einfügeverfahren umgeschaltet.

Das Schnittpunktverfahren für Polynome wird freigegeben mit dem Maschinendatum:

MD20256 $MC_CUTCOM_INTERS_POLY_ENABLE (Schnittpunktverfahren für Polynome möglich)

Hinweis

Wird dieses Maschinendatum auf inaktiv gesetzt, dann wird stets ein (gegebenenfalls sehr kurzer) Satz eingefügt (auch wenn die Übergänge nahezu tangential sind). Diese kurzen Sätze führen im `G64`–Betrieb zwangsläufig zu unerwünschten Geschwindigkeitseinbrüchen.

2.5.12 G461/G462: Erweiterung An- / Abfahrstrategie

Funktion

In bestimmten geometrischen Sonderfällen werden gegenüber der bisherigen Realisierung erweiterte An- und Abfahrstrategien beim Aktivieren bzw. Deaktivieren der Werkzeugradiuskorrektur benötigt (siehe folgendes Bild).

Hinweis

Im Folgenden wird immer nur die Situation bei Deaktivieren der Werkzeugradiuskorrektur dargestellt. Das Verhalten beim Anfahren ist dazu völlig symmetrisch.

Beispiel

```
G42 D1 T1                          ; Werkzeugradius 20mm
...
G1 X110 Y0
N10 X0
N20 Y10
N30 G40 X50 Y50
```

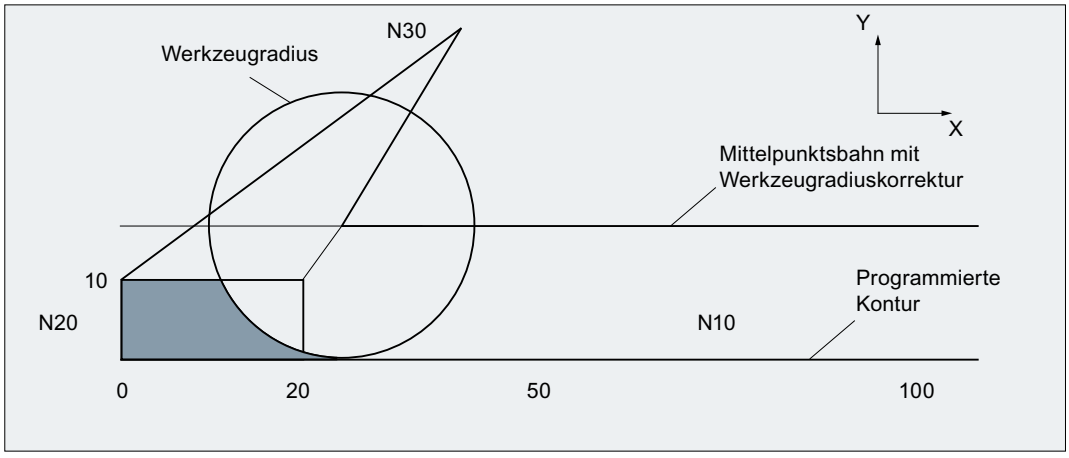


Bild 2-25 Abfahrverhalten bei G460

Der letzte Satz mit aktiver Werkzeugradiuskorrektur (N20) ist so kurz, dass beim aktuellen Werkzeugradius kein Schnittpunkt der Offsetkurve mit dem Vorgängersatz (oder einem weiter davor liegenden Satz) mehr existiert. Es wird deshalb ein Schnittpunkt zwischen den Offsetkurven von Folgesatz und Vorgängersatz gesucht, d. h. im Beispiel zwischen N10 und N30. Die für den Abfahrsatz verwendete Kurve ist dabei keine echte Offsetkurve, sondern eine Gerade vom Offsetpunkt im Endpunkt des Satzes N20 zum programmierten Endpunkt von N30. Wird ein Schnittpunkt gefunden, wird er angefahren. Der im Bild farbig dargestellte Bereich wird dann nicht bearbeitet, wie es mit dem verwendeten Werkzeug eigentlich möglich wäre.

G460

Bei G460 ist das An- bzw. Abfahrverhalten wie bisher.

G461

Wenn kein Schnittpunkt des letzten WRK-Satzes mit einem Vorgängersatz möglich ist, wird die Offsetkurve dieses Satzes mit einem Kreis verlängert, dessen Mittelpunkt im Endpunkt des nicht korrigierten Satzes liegt, und dessen Radius gleich dem Werkzeugradius ist.

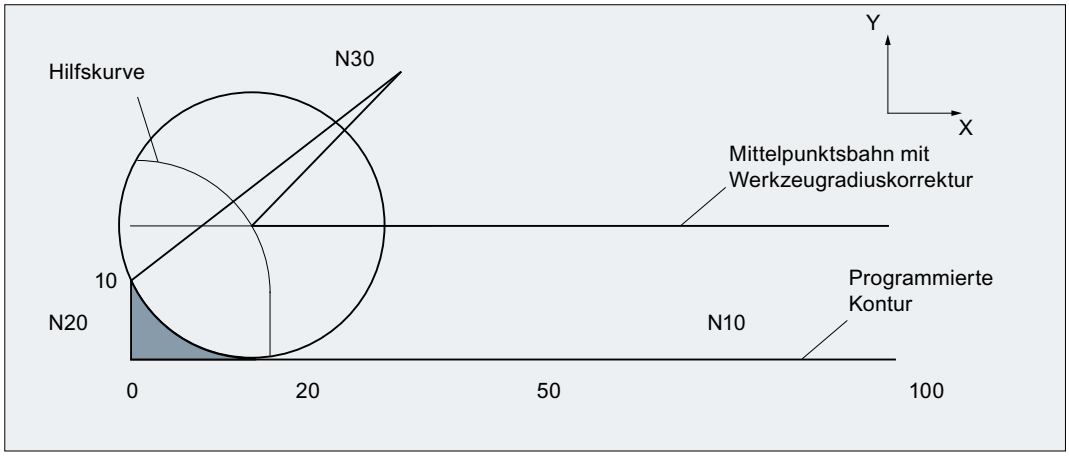


Bild 2-26 Abfahrverhalten bei G461

Die Steuerung versucht, diesen Kreis mit einem der Vorgängersätze zu schneiden. Dabei wird bei aktivem CDOF die Suche abgebrochen, wenn ein Schnittpunkt gefunden wurde, d. h. es wird nicht überprüft, ob auch noch Schnittpunkte mit weiter in der Vergangenheit liegenden Sätzen existieren.

Bei aktivem CDON wird auch dann, wenn bereits ein Schnittpunkt gefunden wurde, nach weiteren Schnittpunkten gesucht.

Ein so gefundener Schnittpunkt ist der neue Endpunkt eines Vorgängersatzes und der Startpunkt des Deaktivierungssatzes. Der eingefügte Kreis dient nur zur Schnittpunktberechnung und hat selbst keine Verfahrbewegung zur Folge.

Hinweis

Wird kein Schnittpunkt gefunden, wird folgender Alarm ausgegeben:

Alarm "10751 Kollisionsgefahr"

G462

Wenn kein Schnittpunkt des letzten WRK-Satzes mit einem Vorgängersatz möglich ist, wird beim Abfahren mit G462 (Grundstellung) im Endpunkt des letzten Satzes mit Werkzeugradiuskorrektur eine Gerade eingefügt (der Satz wird durch seine Endtangente verlängert):

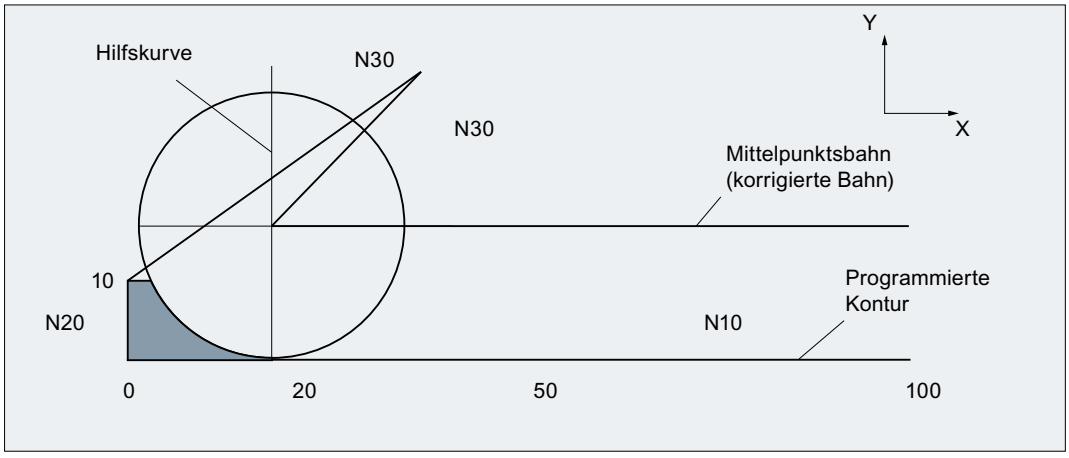


Bild 2-27 Abfahrverhalten bei G462

Die Schnittpunktsuche verläuft dann identisch zu der bei G461.

Bei G462 wird die im Beispielprogramm von N10 und N20 gebildete Ecke nicht soweit ausgeräumt, wie es mit dem verwendeten Werkzeug möglich wäre. Dieses Verhalten kann aber dennoch notwendig sein, wenn die Teilekontur (abweichend von der programmierten Kontur) im Beispiel links von N20 auch bei größeren Werten von y als 10 mm nicht verletzt werden darf.

Ist KONT aktiv (Kontur im Start- oder Endpunkt umfahren), wird unterschieden, ob der Endpunkt vor oder hinter der Kontur liegt.

Endpunkt vor der Kontur

Liegt der Endpunkt vor der Kontur, ist das Abfahrverhalten gleich wie bei NORM. Diese Eigenschaft ändert sich auch nicht, wenn der letzte Kontursatz bei G451 mit einer Geraden oder einem Kreis verlängert wird. Zusätzliche Umfahrungsstrategien, um eine Konturverletzung in der Nähe des Konturendpunktes zu vermeiden, sind deshalb nicht notwendig.

Endpunkt hinter der Kontur

Liegt der Endpunkt hinter der Kontur, wird immer abhängig von G450 / G451 ein Kreis bzw. eine Gerade eingefügt. G460 - G462 hat dann keine Bedeutung.

Hat der letzte Verfahrsatz in dieser Situation keinen Schnittpunkt mit einem Vorgängersatz, kann sich nun ein Schnittpunkt mit dem eingefügten Konturelement oder mit dem Geradenstück vom Endpunkt des Umfahrungskreises zum programmierten Endpunkt ergeben.

Ist das eingefügte Konturelement ein Kreis (G450) und bildet dieser mit dem Vorgängersatz einen Schnittpunkt, ist dieser gleich dem Schnittpunkt, der sich auch bei NORM und G461 ergeben würde. Im Allgemeinen bleibt jedoch ein zusätzliches Stück des Kreises zu verfahren. Für den linearen Teil des Abfahrsatzes ist keine Schnittpunktberechnung mehr notwendig.

Im zweiten Fall (wenn kein Schnittpunkt des eingefügten Konturelements mit den Vorgängersätzen gefunden wird) wird auf den Schnittpunkt zwischen der Abfahrgeraden und einem Vorgängersatz verfahren.

Es kann sich somit bei aktivem G461 bzw. G462 nur dann ein gegenüber G460 verändertes Verhalten ergeben, wenn entweder NORM aktiv ist, oder das Verhalten bei KONT geometrisch bedingt identisch zu dem bei NORM ist.

Hinweis

Das Anfahrverhalten ist symmetrisch zum Abfahrverhalten.

Das An- bzw. Abfahrverhalten wird vom Zustand des G-Befehles im An- bzw. Abfahrsatz bestimmt. Das Anfahrverhalten kann deshalb unabhängig vom Abfahrverhalten eingestellt werden.

Beispiel:

Programm für die Verwendung von G461 beim Anfahren:

```
N10 $TC_DP1[1,1]=120                    ; Werkzeugtyp Fraeser
N20 $TC_DP6[1,1]=10                     ; Radius
N30 X0 Y0 F10000 T1 D1
N40 Y20
N50 G42 X50 Y5 G461
N60 Y0 F600
N70 X30
N80 X20 Y-5
N90 X0 Y0 G40
N100 M30
```

2.5.13 Werkzeuge mit relevanter Schneidenlage

Für Werkzeuge mit relevanter Schneidenlage ist Folgendes zu beachten:

- Für die Berechnung von Schnittpunkten mit dem An- bzw. Abfahrsatz wird die Gerade zwischen den Schneidenmittelpunkten am Satzanfang und am Satzende verwendet. Die Differenz zwischen Schneidenbezugspunkt und Schneidenmittelpunkt wird dieser Bewegung überlagert.
  Beim An- bzw. Abfahren mit `KONT` erfolgt die Überlagerung im linearen Teilsatz der An- bzw. Abfahrbewegung. Die geometrischen Verhältnisse sind deshalb bei Werkzeugen mit und ohne relevante Schneidenlage identisch.
- Der Wechsel eines Werkzeugs bei aktiver Werkzeugradiuskorrektur, bei dem sich der Abstand zwischen Schneidenmittelpunkt und Schneidenbezugspunkt ändert, ist in Kreissätzen und in Verfahrsätzen mit rationalen Polynomen mit einem Nennergrad > 4 verboten. Bei anderen Interpolationsarten ist ein Wechsel im Gegensatz zum bisherigen Zustand auch bei aktiver Transformation (z. B. `TRANSMIT`) zulässig.
- Bei der Werkzeugradiuskorrektur mit variabler Werkzeugorientierung ist die Transformation vom Schneidenbezugspunkt auf den Schneidenmittelpunkt nicht durch eine einfache Nullpunktverschiebung realisierbar. Werkzeuge mit relevanter Schneidenlage werden deshalb beim 3D-Umfangsfräsen verboten (Alarm).

Hinweis

Für das Stirnfräsen ist das Thema nicht relevant, da hier ohnehin nur definierte Werkzeugtypen ohne relevante Schneidenlage zugelassen sind. (Werkzeuge mit einem nicht ausdrücklich zugelassenen Werkzeugtyp werden als Kugelkopffräser mit dem angegebenen Radius behandelt. Die Angabe einer Schneidenlage wird ignoriert.)

2.6 2 1/2 D-Werkzeugradiuskorrektur

Die 2½ D-Werkzeugradiuskorrektur ist zu verwenden, wenn für die Bearbeitung von schrägen Flächen nicht die Ausrichtung des Werkzeugs, sondern das **Werkstück** gedreht wird. Die Aktivierung erfolgt durch die Befehle `CUT2D`, `CUT2DD`, `CUT2DF` oder `CUT2DFD`.

Werkzeuglängenkorrektur

Die Werkzeuglängenkorrektur wird immer bezogen auf die raumfeste, nicht gedrehte Arbeitsebene eingerechnet.

2½ D-Werkzeugradiuskorrektur für Konturwerkzeuge

Die 2½ D-Werkzeugradiuskorrektur für Konturwerkzeuge wird aktiviert, wenn zusammen mit `CUT2D`, `CUT2DD`, `CUT2DF` oder `CUT2DFD` einer der beiden Befehle `G41` (Werkzeugradiuskorrektur links von der Kontur) oder `G42` (Werkzeugradiuskorrektur rechts von der Kontur) programmiert wird. Sie dient zur automatischen Schneidenauswahl für nicht

rotationssymmetrische Werkzeuge, mit denen stückweise einzelne Kontursegmente bearbeitet werden können.

Hinweis

Bei nicht aktiver 2½ D-Werkzeugradiuskorrektur verhält sich ein Konturwerkzeug wie ein normales Werkzeug, das nur aus der ersten Schneide besteht.

2½ D-Werkzeugradiuskorrektur bezogen auf ein Differenzwerkzeug

Die auf ein Differenzwerkzeug bezogene 2½ D-Werkzeugradiuskorrektur wird durch die Befehle `CUT2DD` oder `CUT2DFD` aktiviert. Sie ist anzuwenden, wenn sich die programmierte Kontur auf die Mittelpunktsbahn eines Differenzwerkzeugs bezieht und die Bearbeitung mit einem davon abweichenden Werkzeug erfolgt. Bei der Berechnung der 2½ D-Werkzeugradiuskorrektur werden dann nur der Verschleißwert des Radius des aktiven Werkzeugs ($TC_DP_15) und die gegebenenfalls programmierten Werkzeugkorrekturoffsets OFFN und TOFFR eingerechnet. Der Grundradius ($TC_DP6) des aktiven Werkzeugs wird **nicht** eingerechnet.

Syntax

```
CUT2D
CUT2DD
CUT2DF
CUT2DFD
```

Bedeutung

`CUT2D`:	Aktivierung der 2½ D-Radiuskorrektur
`CUT2DD`:	Aktivierung der auf ein Differenzwerkzeug bezogenen 2½ D-Radiuskorrektur
`CUT2DF`:	Aktivierung der 2½ D-Radiuskorrektur, Werkzeugradiuskorrektur relativ zum aktuellen Frame bzw. schrägen Ebene
`CUT2DFD`:	Aktivierung der auf ein Differenzwerkzeug bezogenen 2½ D-Radiuskorrektur relativ zum aktuellen Frame bzw. schrägen Ebene

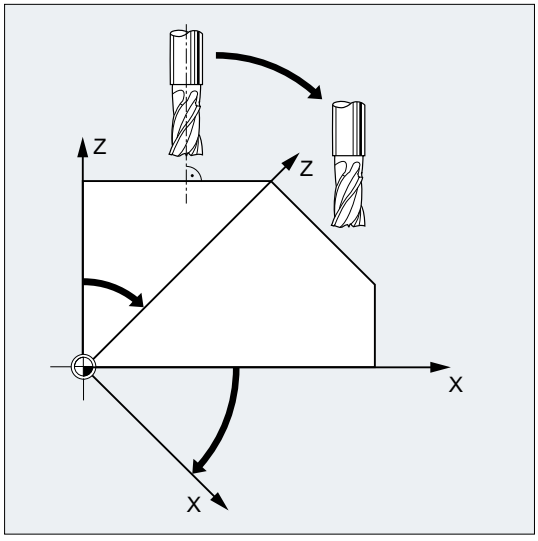
Weitere Informationen

Konturwerkzeuge

- Freigabe
  Die Freigabe der Werkzeugradiuskorrektur für Konturwerkzeuge erfolgt kanalspezifisch über:
  MD28290 $MC_MM_SHAPED_TOOLS_ENABLE
- Werkzeugtyp
  Die Werkzeugtypen von Konturwerkzeugen werden kanalspezifisch festgelegt über:
  MD20370 $MC_SHAPED_TOOL_TYPE_NO
- Schneiden
  Jedem Konturwerkzeug können in beliebiger Reihenfolge eine Anzahl Schneiden (D-Nummern) zugeordnet werden. Die maximale Anzahl von Schneiden pro Werkzeug wird parametriert über:
  MD18106 $MN_MM_MAX_CUTTING_EDGE_PERTOOL
  Die erste Schneide eines Konturwerkzeugs ist die Schneide, die bei der Aktivierung des Werkzeugs angewählt wird. Wird z. B. in einem Programm durch die Befehle `T3 D5` die fünfte Schneide (D5) des dritten Werkzeugs (T3) aktiviert, dann definiert D5 und die nachfolgenden Schneiden entweder mit einem Teil oder alle zusammen das Konturwerkzeug. Die vor D5 liegenden Schneiden werden ignoriert.

2½ D-Werkzeugradiuskorrektur ohne Drehung der Korrekturebene (CUT2D, CUT2DD)

Wird ein Frame programmiert, der eine Drehung enthält, wird bei `CUT2D` bzw. `CUT2DD` die Ebene, in der die Werkzeugradiuskorrektur (Korrekturebene) stattfindet, **nicht mitgedreht**. Die Werkzeugradiuskorrektur wird bezogen auf die **nicht gedrehte** Arbeitsebene (`G17`, `G18`, `G19`) eingerechnet. Die Werkzeuglängenkorrektur wirkt relativ zur Korrekturebene.

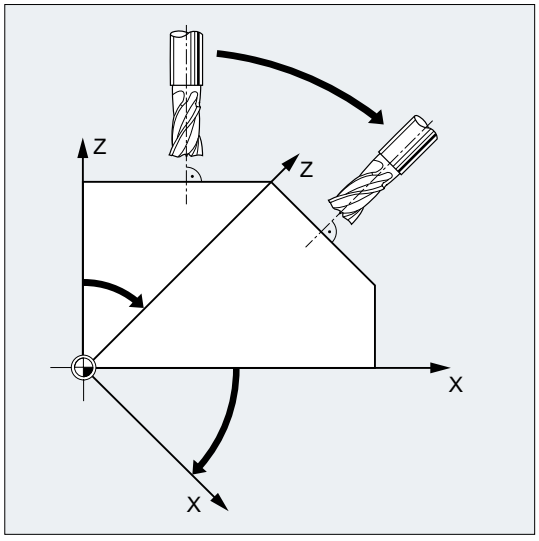


Für die Bearbeitung von schrägen Flächen müssen die Werkzeugkorrekturwerte entsprechend definiert oder unter Einsatz der Funktionalitäten zur "Werkzeuglängenkorrektur für orientierbare Werkzeuge" berechnet werden.

2½ D-Werkzeugradiuskorrektur mit Drehung der Korrekturebene (CUT2DF, CUT2DFD)

Wird ein Frame programmiert, der eine Drehung enthält, wird bei `CUT2DF` bzw. `CUT2DFD` die Ebene, in der die Werkzeugradiuskorrektur (Korrekturebene) stattfindet, **mitgedreht**. Die Werkzeugradiuskorrektur wird bezogen auf die **gedrehte** Arbeitsebene (`G17`, `G18`, `G19`) eingerechnet. Die Werkzeuglängenkorrektur wirkt aber weiterhin relativ zur **nicht gedrehten** Arbeitsebene.

Voraussetzung: An der Maschine muss die Werkzeugorientierung senkrecht zur gedrehten Arbeitsebene einstellbar sein und zur Bearbeitung eingestellt werden.



Hinweis

Die Werkzeuglängenkorrektur wirkt weiterhin relativ zur nicht gedrehten Arbeitsebene.

Weitere Informationen siehe Funktionshandbuch "Werkzeuge".

Siehe auch

Werkzeuglängenkorrektur (Seite 19)

2.7 Werkzeugradiuskorrektur konstant halten

Bedeutung

Die Funktion "Werkzeugradiuskorrektur konstant halten" dient dazu, die Werkzeugradiuskorrektur für eine Anzahl von Sätzen zu unterdrücken, wobei jedoch eine durch die Werkzeugradiuskorrektur in vorhergehenden Sätzen aufgebaute Differenz zwischen der programmierten und der tatsächlich abgefahrenen Bahn des Werkzeugmittelpunktes als Verschiebung beibehalten wird.

Sie kann z. B. dann vorteilhaft eingesetzt werden, wenn beim Zeilenfräsen in den Umkehrpunkten mehrere Verfahrsätze notwendig sind, die von der Werkzeugradiuskorrektur erzeugten Konturen (Umfahrungsstrategien) jedoch nicht erwünscht sind.

Aktivierung

Die Funktion "Werkzeugradiuskorrektur konstant halten" wird mit dem G-Befehl CUTCONON (CUTter compensation CONstant ON) aktiviert und mit dem G-Befehl CUTCONOF (CUTter compensation CONstant OFF) deaktiviert.

CUTCONON und CUTCONOF bilden eine modale G-Gruppe.

Grundstellungswert ist CUTCONOF.

Die Funktion ist unabhängig von der Art der Werkzeugradiuskorrektur (2D-WRK, ...) einsetzbar.

Normalfall

Im Normalfall ist vor der Aktivierung der Korrekturunterdrückung die Werkzeugradiuskorrektur bereits aktiv, und sie ist noch aktiv, wenn die Korrekturunterdrückung wieder deaktiviert wird.

Im letzen Verfahrsatz vor CUTCONON wird auf den Offsetpunkt im Satzendpunkt gefahren. Alle folgenden Sätze, in denen die Korrekturunterdrückung aktiv ist, werden ohne Korrektur verfahren. Sie werden dabei jedoch um den Vektor vom Endpunkt des letzten Korrektursatzes zu dessen Offsetpunkt verschoben. Der Interpolationstyp dieser Sätze (linear, zirkular, polynomial) ist beliebig.

Der Deaktivierungssatz der Korrekturunterdrückung, d. h. der Satz, der CUTCONOF enthält, wird normal korrigiert. Er beginnt im Offsetpunkt des Startpunktes. Zwischen dem Endpunkt des Vorgängersatzes, d. h. des letzten programmierten Verfahrsatzes mit aktivem CUTCONON, und diesem Punkt wird ein linearer Satz eingefügt.

Kreissätze, bei denen die Kreisebene senkrecht auf der Korrekturebene steht (vertikale Kreise), werden so behandelt, als ob in ihnen CUTCONON programmiert wäre. Diese implizite Aktivierung der Korrekturunterdrückung wird im ersten Verfahrsatz, der eine Verfahrbewegung in der Korrekturebene enthält und der kein derartiger Kreis ist, automatisch rückgängig gemacht. Vertikale Kreise in diesem Sinne können nur beim Umfangsfräsen auftreten.

Beispiel:

N10	; Definition des Werkzeugs d1
N20 $TC_DP1[1,1]= 110	; Typ
N30 $TC_DP6[1,1]=	; Radius
N40	
N50 X0 Y0 Z0 G1 G17 T1 D1 F10000	
N60	
N70 X20 G42 NORM	

N80 X30	
N90 Y20	
N100 X10 CUTCONON	; Einschalten der Korrekturunterdrückung
N110 Y30 KONT	; Beim Ausschalten der Konturunterdrückung gegebenenfalls Umfahrungskreis einfügen
N120 X-10 CUTCONOF	
N130 Y20 NORM	; Kein Umfahrungskreis beim Ausschalten der WRK
N140 X0 Y0 G40	
N150 M30	

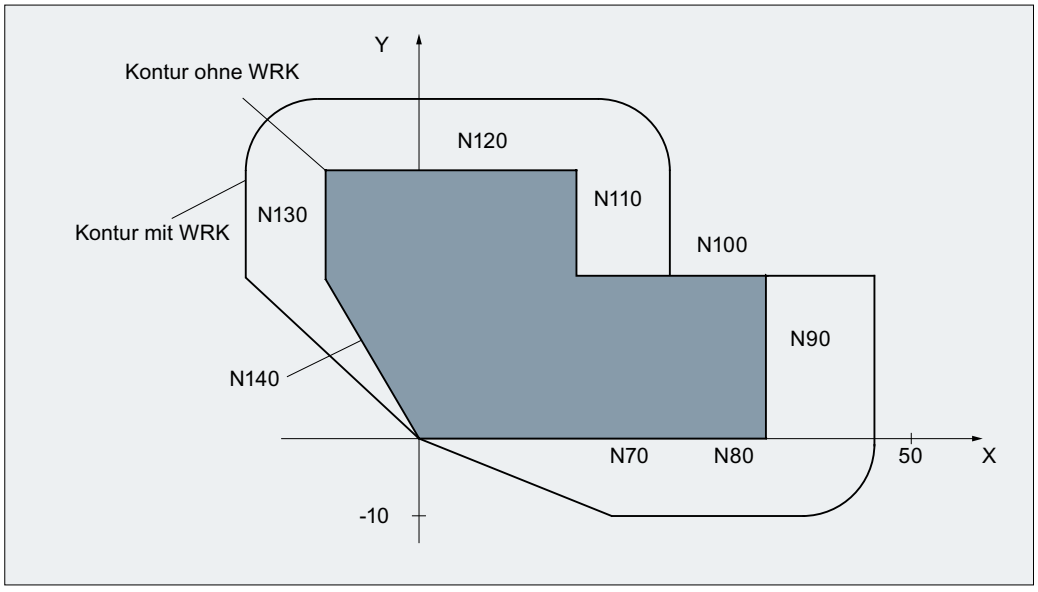


Bild 2-28 Beispielprogramm zur Konturunterdrückung

Sonderfälle

- Ist keine Werkzeugradiuskorrektur aktiv (G40), hat CUTCONON keine Wirkung. Es wird kein Alarm erzeugt. Der G-Befehl bleibt jedoch aktiv.
  Dies ist dann von Bedeutung, wenn in einem späteren Satz die Werkzeugradiuskorrektur mit G41 bzw. mit G42 eingeschaltet werden soll.
- Ein Wechsel des G-Befehls in der 7. G-Gruppe (Werkzeugradiuskorrektur; G40 / G41 / G42) bei aktivem CUTCONON ist zulässig. Ein Wechsel nach G40 wird sofort wirksam.
  Die Verschiebung, mit der die Vorgängersätze verfahren wurden, wird dabei herausgefahren.
- Wird CUTCONON oder CUTCONOF in einem Satz ohne Verfahrbewegung in der aktiven Korrekturebene programmiert, so wird die Wirksamkeit bis zum nächsten Satz mit einer solchen Verfahrbewegung verzögert.

- Wird CUTCONON bei aktiver Werkzeugradiuskorrektur programmiert und vor Programmende nicht mehr aufgehoben, so werden die betroffenen Verfahrsätze mit dem letzten gültigen Offset verfahren.
  Gleiches gilt bei erneuter Programmierung von G41 bzw. G42 im letzten Verfahrsatz eines Programms.
- Wird die Werkzeugradiuskorrektur mit G41 bzw. G42 eingeschaltet und ist gleichzeitig bereits CUTCONON aktiv, wird die Aktivierung der Korrektur bis zum ersten Verfahrsatz mit CUTCONOF verzögert.
- Beim Wiederanfahren an die Kontur mit CUTCONOF wird die 17. G-Gruppe (An- und Abfahrverhalten bei Werkzeugkorrektur; NORM / KONT) ausgewertet, d. h. bei KONT wird gegebenenfalls ein Umfahrungskreis eingefügt. Ein Umfahrungskreis wird unter den gleichen Bedingungen wie bei der Aktivierung der Werkzeugradiuskorrektur mit G41 bzw. G42 eingefügt.
- Die Zahl der Sätze mit unterdrückter Werkzeugradiuskorrektur ist begrenzt:
  MD20252 $MC_CUTCOM_MAXNUM_SUPPR_BLOCKS (Maximale Satzzahl mit Korrekturunterdrückung)
  Wird sie überschritten, wird die Bearbeitung mit einer Fehlermeldung abgebrochen.
  Die Begrenzung ist notwendig, weil beim Repositionieren die interne Satzbearbeitung im letzten Satz vor CUTCONON wieder aufgenommen werden muss.
- Das Verhalten bei erneuter Programmierung von G41 bzw. G42 bei bereits aktiver Werkzeugradiuskorrektur ist ähnlich wie bei der Korrekturunterdrückung.
  Es gelten folgende Abweichungen:
  - Es sind nur Linearsätze zugelassen
  - Ein einzelner Verfahrsatz, der G41 bzw. G42 enthält, wird so modifiziert, dass er im Versatzpunkt des Startpunktes des Folgesatzes endet. Dadurch entfällt das Einfügen eines Zwischensatzes. Gleiches gilt für den letzten einer Folge von Verfahrsätzen, von denen jeder G41 bzw. G42 enthält.
  - Das erneute Anfahren der Kontur erfolgt unabhängig vom G-Befehl der 17. Gruppe (An- und Abfahrverhalten bei Werkzeugkorrektur; NORM / KONT) immer mit NORM.
- Wird G41 / G42 mehrfach in aufeinander folgenden Verfahrsätzen programmiert, so werden alle Sätze bis auf den letzten wie bei CUTCONON bearbeitet.
- Die Art der Konturunterdrückung wird nur im ersten Verfahrsatz einer Folge von Sätzen ausgewertet.
  Ist im ersten Satz sowohl CUTCONON als auch G41 bzw. G42 programmiert, wird das Verhalten beim Deaktivieren der Konturunterdrückung von CUTCONON bestimmt.
  Ein Wechsel von G41 nach G42 bzw. umgekehrt kann in diesem Fall sinnvoll dazu verwendet werden, die Korrekturseite (links bzw. rechts von der Kontur) beim Wiederanfahren zu wechseln.
  Ein Wechsel der Korrekturseite (G41 / G42) kann bei aktiver Konturunterdrückung auch in einem späteren Satz programmiert werden.
- Für alle Sätze mit aktiver Konturunterdrückung ist die Kollisionsüberwachung bzw. Flaschenhalserkennung ausgeschaltet.

2.8 Orientierbare Werkzeugträger

2.8.1 Allgemeine Informationen

Einführung

Bei einer Klasse von Werkzeugmaschinen ist die Orientierung des Werkzeugs (z. B. **durch Umrüsten**) veränderbar. Im Betrieb ist die einmal eingestellte Orientierung jedoch **fest** und kann insbesondere während des Verfahrens nicht verändert werden. Aus diesem Grund ist für derartige Maschinen eine kinematische Orientierungstransformation (3-, 4- oder 5-Achstransformationen, `TRAORI`) weder notwendig noch sinnvoll. Es besteht jedoch die Notwendigkeit, die durch eine Orientierungsänderung bedingten Änderungen der Werkzeuglängenkomponenten zu berücksichtigen, ohne den Anwender mit den dazu erforderlichen Berechnungen zu belasten. Die Steuerung übernimmt diese Berechnungen.

Datenvorgaben

Wenn die Steuerung Werkzeugkorrekturen bei orientierbaren Werkzeugträgern berücksichtigen soll, müssen vorliegen:

- Werkzeugdaten (Geometrie, Verschleiß ...)
- Werkzeugträgerdaten (Angaben zur Geometrie des orientierbaren Werkzeugträgers)

Werkzeugträgeranwahl

Für die Funktion "Orientierbarer Werkzeugträger" muss ein definierter Werkzeugträger der Steuerung angegeben werden. Dafür existiert der NC-Programmbefehl:

```
TCARR = m
```

m: Nummer des Werkzeugträgers

Mit dem Werkzeugträger ist ein Werkzeugträgerdatensatz verbunden, der dessen Geometrie beschreibt.

Die Aktivierung des Werkzeugträgers und seines Datensatzes wirkt sofort, d. h. ab dem nächsten Bewegungssatz.

Zuordnung Werkzeug / Werkzeugträger

Das bisher aktive Werkzeug wird dem neu angewählten Werkzeugträger zugeordnet.

Aus Sicht der Steuerung sind Werkzeugträgernummern m und Werkzeugnummern T frei kombinierbar. In der realen Anwendung können sich Kombinationen aus Bearbeitungsgründen und mechanischen Gründen ausschließen. Die Steuerung prüft nicht auf sinnvolle Kombinationen.

Beschreibung der Kinematik des WZ-Trägers

Die Kinematik des orientierbaren Werkzeugträgers wird mit insgesamt 33 Parametersätzen beschrieben.

Die Daten des Datensatzes sind durch den Anwender bearbeitbar.

Orientierbare Werkzeugträger

Beispiel: Kardanischer Werkzeugträger mit zwei Achsen für die Werkzeugorientierung

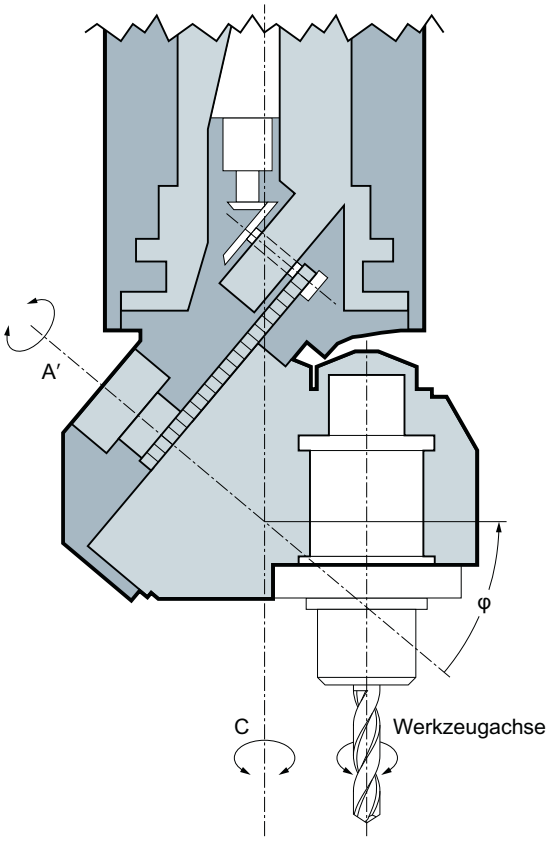


Bild 2-29 Kardanischer Werkzeugträger mit zwei Achsen

Bearbeitung der Werkzeuträger-Datensätze

Es werden zwei Möglichkeiten angeboten:

- Explizite Eintragung im Werkzeugträger-Datensatz aus dem Teileprogramm
- Automatische Übernahme einiger Werte (Winkel) aus einem Frame
  Voraussetzung hierfür ist, dass bei der Werkzeugträgeranwahl zusätzlich `TCOFR` (Tool Carrier Orientation FRame) angegeben wird.
  Die für die Werkzeuglängenberechnung verwendete Werkzeugorientierung wird bei einem Werkzeugträgerwechsel neu aus dem zu diesem Zeitpunkt aktiven Frame ermittelt.

Orientierung in der Z-Richtung

Die G-Befehle TOFRAME definiert einen Frame so, dass die Z-Richtung in diesem Frame gleich der aktuellen Werkzeugorientierung ist.

Wenn kein Werkzeugträger oder ein Werkzeugträger ohne Orientierungsänderung aktiv ist, dann ist die Z-Richtung im neuen Frame:

- bei G17 gleich der alten Z-Richtung.
- bei G18 gleich der alten Y-Richtung.
- bei G19 gleich der alten X-Richtung.

TCOABS bei aktivem Frame

Die absolute Werkzeugträgerorientierung wird eingestellt durch:

TCOABS (Tool Carrier Orientation ABSolut)

Die für die Werkzeuglängenkorrektur berücksichtigte Orientierung ist **unabhängig** von der Orientierung des aktiven Frames.

Es kann jeweils nur eine der Anweisungen TCOABS oder TCOFR gültig sein.

Wechsel des Frames

Der Anwender kann nach der Anwahl des Werkzeuges den Frame wechseln. Das hat keinen Einfluss auf die Komponenten der Werkzeuglängenkorrektur.

Winkel in den Werkzeugträgerdaten:

Die in den Werkzeugträgerdaten abgelegten programmierten Drehwinkel werden durch die mit Frames festgelegten Drehwinkel nicht beeinflusst. Beim Wechsel von TCOFR nach TCOABS werden die ursprünglichen (programmierten) Drehwinkel in den Werkzeugträgerdaten wieder wirksam.

Werkzeugkorrektur-Typen

Die Werkzeugradienkorrektur WRK berücksichtigt die aktuelle Werkzeugorientierung, wenn CUT2D oder CUT3DFS aktiv ist.

Alle anderen Werkzeugkorrektur-Typen

Das sind alle Korrekturtypen der G-Gruppe 22 mit Ausnahme von CUT3DC und CUT3DF. Das Verhalten ist unverändert bezüglich der für die Korrektur verwendeten Ebene. Diese wird unabhängig von der Werkzeug-Orientierung aus dem aktiven Frame bestimmt.

Bei CUT2DF und CUT3DFF wird die für WRK verwendete Korrekturebene **unabhängig** von der aktuellen Werkzeugorientierung aus dem Frame ermittelt. Es wird die aktive Ebene (G17/G18/G19) berücksichtigt.

CUT3DC und CUT3DF

3D-Werkzeugkorrektur für das Umfangsfräsen

3D-Werkzeugkorrektur für das Stirnfräsen bei aktiver 5-Achs-Transformation sind von der Funktion "Orientierbare Werkzeugträger" nicht betroffen.

Die Orientierungsinformation wird durch die aktive kinematische 5-Achs-Transformation ermittelt.

Eingeschränkte Orientierung des Werkzeugträger

Wird durch den Frame eine Orientierung festgelegt, die mit der definierten Werkzeugträgerkinematik nicht erreicht werden kann, wird ein Alarm ausgegeben.

Folgende Kinematiken können nicht jede Orientierung erreichen:

- wenn die beiden Drehachsen, die zur Definition der Kinematik notwendig sind, nicht aufeinander senkrecht stehen und wenn die Werkzeugachse, die die Werkzeugrichtung definiert, nicht senkrecht auf der zweiten Drehachse steht,
  oder
- wenn weniger als zwei Achsen definiert sind.

Nicht drehende Werkzeugträger

Die intern verwendete Werkzeugorientierung ist nur von der Grundorientierung des Werkzeugs und der aktiven Ebene (`G17` - `G19`) abhängig.

Mehrdeutigkeiten

Eine bestimmte, durch Frame vorgegebene Werkzeugorientierung, kann mit zwei Achsen im Allgemeinen durch **zwei** verschiedene Drehwinkelpaare eingestellt werden. Die Steuerung wählt von diesen zwei möglichen Stellungen diejenige aus, bei der die Drehwinkel möglichst nahe an den programmierten Drehwinkeln liegen.

Ablage von Winkeln in den Werkzeugträgerdaten

In ungefähr allen Fällen, in denen eine Mehrdeutigkeit auftreten kann, ist es erforderlich, die aus dem Frame zu erwartenden Winkel in den Werkzeugträgerdaten abzulegen.

Parametersätze

Ein vollständiger Parametersatz für einen orientierbaren Werkzeugträger besteht aus 33 Werten.

Folgende Systemvariablen sind verfügbar:

- $TC_CARR1 bis $TC_CARR33
- $TC_CARR34 bis $TC_CARR65 zusätzlich zur freien Verfügung des Anwenders und für Feinverschiebungen.

Die Bedeutung der einzelnen Parameter wird im Folgenden unterschieden.

Kinematik der Maschine:

$TC_CARR1 bis $TC_CARR20 sowie $TC_CARR23

Dabei definieren $TC_CARR18 bis $TC_CARR20 einen weitern Vektor l_4, der zur Beschreibung der Maschine bei aufgelösten Kinematiken (sowohl Werkzeug als auch Werkstück sind drehbar) benötigt wird.

$TC_CARR21 und $TC_CARR22 enthalten die Kanalachsnamen der Rundachsen, auf deren Position gegebenenfalls bei der Bestimmung der Orientierung des orientierbaren Werkzeugträgers zugegriffen wird.

Kinematiktyp:

$TC_CARR23 mittels eines Buchstabens T, P oder M

Beim Kinematiktyp gibt es die folgenden drei Möglichkeiten, wobei Groß- und Kleinschreibung zulässig ist:

T: Nur das Werkzeug (**T**ool) ist drehbar (Basiswert).
P: Nur das Werkstück (**P**art) ist drehbar.
M: Werkstück und Werkzeug sind drehbar (**M**ixed mode).

Jedes andere außer den genannten drei Zeichen führt bei dem Versuch, den orientierbaren Werkzeugträgern zu aktivieren, zu dem Alarm:

Alarm "14153 Kanal %1 Satz %2 Unbekannter Werkzeugtraegertyp: %3"

Parametrierung der Drehachsen:

$TC_CARR24 bis $TC_CARR33

Mit den Systemvariablen in $TC_CARR24 bis $TC_CARR33 können Offsets, Winkeloffsets, Hirth-Verzahnung sowie Achsgrenzen parametriert werden.

Hinweis

Die Systemvariablen sind sowohl mit als auch ohne aktive Werkzeugverwaltung verfügbar.

Komponenten und Vorbesetzung der Kette / Datensatz

Die Werte $TC_CARR1 bis $TC_CARR20 und $TC_CARR24 - $TC_CARR33 im Werkzeugträger-Datensatz sind vom NC-Sprachformat-Typ `REAL`.

Die Werte $TC_CARR21 und $TC_CARR22 für die Achsnamen der ersten Drehachse (v_1) und der zweiten Drehachse (v_2) sind vom NC-Sprachformat-Typ AXIS. Sie sind alle mit Null vorbesetzt.

Der Wert $TC_CARR23 wird mit dem Großbuchstaben "T" (nur das Werkzeug ist drehbar) vorbesetzt.

$TC_CARRn[m]

$TC_CARR[0]= 0 hat Sonderbedeutung

Systemvariablen für orientierbare Werkzeugträger

$TC_CARRn[m]

n: Parameter 1...33

m: Nummer des orientierbaren WZ-Trägers 1...Wert des Maschinendatums:
MD18088 $MN_MM_NUM_TOOL_CARRIER (Maximale Anzahl definierbarer Werkzeugträger)

Beschreibung	NC-Variable	Sprach-Format	Vorbelegung
x-Komponente des Offsetvektors l_1	$TC_CARR1	REAL	0
y-Komponente des Offsetvektors l_1	$TC_CARR2	REAL	0
z-Komponente des Offsetvektors l_1	$TC_CARR3	REAL	0
x-Komponente des Offsetvektors l_2	$TC_CARR4	REAL	0
y-Komponente des Offsetvektors l_2	$TC_CARR5	REAL	0
z-Komponente des Offsetvektors l_2	$TC_CARR6	REAL	0
x-Komponente der Drehachse v_1	$TC_CARR7	REAL	0
y-Komponente der Drehachse v_1	$TC_CARR8	REAL	0
z-Komponente der Drehachse v_1	$TC_CARR9	REAL	0
x-Komponente der Drehachse v_2	$TC_CARR10	REAL	0
y-Komponente der Drehachse v_2	$TC_CARR11	REAL	0
z-Komponente der Drehachse v_2	$TC_CARR12	REAL	0
Drehwinkel α1(in Grad)	$TC_CARR13	REAL	0
Drehwinkel α_2 (in Grad)	$TC_CARR14	REAL	0
x-Komponente des Offsetvektors l_3	$TC_CARR15	REAL	0
y-Komponente des Offsetvektors l_3	$TC_CARR16	REAL	0
z-Komponente des Offsetvektors l_3	$TC_CARR17	REAL	0
x-Komponente des Offsetvektors l_4	$TC_CARR18	REAL	0
y-Komponente des Offsetvektors l_4	$TC_CARR19	REAL	0
z-Komponente des Offsetvektors l_4	$TC_CARR20	REAL	0
Achsname der Drehachse v_1	$TC_CARR21	AXIS	0
Achsname der Drehachse v_2	$TC_CARR22	AXIS	0
Kinematiktyp	$TC_CARR23	CHAR	T
Offset der Drehachse v_1	$TC_CARR24	REAL	0
Offset der Drehachse v_2	$TC_CARR25	REAL	0
Winkeloffset der Drehachse v_1 (Hirthverz.)	$TC_CARR26	REAL	0
Winkeloffset der Drehachse v_2 (Hirthverz.)	$TC_CARR27	REAL	0
Winkelinkrement Drehachse v_1 (Hirthverz.)	$TC_CARR28	REAL	0
Winkelinkrement Drehachse v_2 (Hirthverz.)	$TC_CARR29	REAL	0
Minimalposition Drehachse v_1 (SW-Limit)	$TC_CARR30	REAL	0
Minimalposition Drehachse v_2 (SW-Limit)	$TC_CARR31	REAL	0
Maximalposition Drehachse v_1 (SW-Limit)	$TC_CARR32	REAL	0
Maximalposition Drehachse v_2 (SW-Limit)	$TC_CARR33	REAL	0

Systemvariablen für den Anwender und für Feinverschiebungen

- $TC_CARR34 bis $TC_CARR40
  Enthalten Parameter, die dem Anwender zur freien Verfügung stehen.
- $TC_CARR41 bis $TC_CARR65
  Enthalten Feinverschiebungsparameter, die zu den Werten in den Basisparametern addiert werden können. Der einem Basisparameter zugeordnete Feinverschiebungswert ergibt sich, wenn zur Parameternummer der Wert 40 addiert wird.
- $TC_CARR47 bis $TC_CARR54 sowie $TC_CARR61 bis $TC_CARR63
  Sind nicht definiert und führen bei dem Versuch, lesend oder schreibend darauf zuzugreifen, zu einem Alarm.

Beschreibung	NC-Variable	Sprach-Format	Vorbelegung
Werkzeugträgername	$TC_CARR34	String[32]	""
Achsname 1	$TC_CARR35 *)	String[32]	""
Achsname 2	$TC_CARR36 *)	String[32]	""
Kennung	$TC_CARR37 *)	INT	0
Positionskomponente X	$TC_CARR38 *)	REAL	0
Positionskomponente Y	$TC_CARR39 *)	REAL	0
Positionskomponente Z	$TC_CARR40 *)	REAL	0
x-Komp. Feinversch. des Offsetvektors l_1	$TC_CARR41	REAL	0
y-Komp. Feinversch. des Offsetvektors l_1	$TC_CARR42	REAL	0
z-Komp. Feinversch. des Offsetvektors l_1	$TC_CARR43	REAL	0
x-Komp. Feinversch. des Offsetvektors l_2	$TC_CARR44	REAL	0
y-Komp. Feinversch. des Offsetvektors l_2	$TC_CARR45	REAL	0
z-Komp. Feinversch. des Offsetvektors l_2	$TC_CARR46	REAL	0
x-Komp. Feinversch. des Offsetvektors l_3	$TC_CARR55	REAL	0
y-Komp. Feinversch. des Offsetvektors l_3	$TC_CARR56	REAL	0
z-Komp. Feinversch. des Offsetvektors l_3	$TC_CARR57	REAL	0
x-Komp. Feinversch. des Offsetvektors l_4	$TC_CARR58	REAL	0
y-Komp. Feinversch. des Offsetvektors l_4	$TC_CARR59	REAL	0
z-Komp. Feinversch. des Offsetvektors l_4	$TC_CARR60	REAL	0
Offset der Feinversch. der Drehachse v_1	$TC_CARR64	REAL	0
Offset der Feinversch. der Drehachse v_2	$TC_CARR65	REAL	0
Anmerkungen:			
*)	Die Systemvariablen $TC_CARR35 bis $TC_CARR40 werden in den Messzyklen sowie ShopMill und ShopTurn verwendet.		

2.8.2 Kinematische Zusammenhänge und Maschinenaufbau

Darstellung der kinematischen Kette

Zur Beschreibung der kinematischen Zusammenhänge zwischen einem Bezugspunkt und der Werkzeugspitze wird das Konzept der kinematischen Kette verwendet.

Die Kette gibt alle für den Werkzeugträger-Datensatz erforderlichen Angaben schematisch an. Für den konkreten Fall einer bestimmten Kinematik müssen die entsprechenden Komponenten der Kette mit den realen Vektoren, Längen und Winkeln besetzt werden. Die Kette stellt die maximal zulässige Konstellation dar. Einzelkomponenten können in einfacheren Fällen zu Null werden (z. B. Kinematiken, die nur eine oder überhaupt keine Drehachse haben).

An der Maschine müssen keine Achsen vorhanden sein, die Werkzeug und / oder Werkstücktisch drehen. Die Funktion ist auch dann nutzbar, wenn die Orientierungen z. B. manuell durch Handräder oder Umbau eingestellt werden.

Der Maschinenaufbau wird durch folgende Angaben beschrieben:

- Zwei Drehachsen (v_1 und v_2) mit Angabe von jeweils einem Drehwinkel (α_1 bzw. α_2), der bei Drehung im Uhrzeigersinn und Blick in Richtung des Drehvektors positiv zählt.
- Bis zu vier Offsetvektoren (l_1 bis l_4) für relevante Maschinenabmessungen (Achsabstände, Abstände zu Maschinen- oder Werkzeugreferenzpunkten).

Nullvektoren

Die Vektoren v_1 bzw. v_2 können Null sein. Der jeweils zugehörige Drehwinkel (explizit programmiert oder aus dem aktiven Frame berechnet) muss dann, da die Richtung der Drehachse nicht definiert ist, ebenfalls Null sein. Ist diese Bedingung nicht erfüllt, wird bei der Aktivierung des Werkzeugträgers ein Alarm ausgegeben.

Weniger als zwei Drehachsen

Die Möglichkeit, eine Drehachse nicht zu definieren, ist dann sinnvoll, wenn der zu beschreibende Werkzeugträger das Werkzeug nur in einer Ebene drehen kann. Ein sinnvoller minimaler Datensatz kann deshalb aus einem einzigen von Null verschiedenen Eintrag in den Werkzeugträgerdaten bestehen, nämlich einem Wert in einer der Komponenten von v_1 bzw. v_2 zur Beschreibung einer achsparallelen Drehachse, wobei der zugehörige Drehwinkel α1 bzw. α2 aus einem Frame bestimmt wird.

Weitere Sonderfälle

Die beiden Vektoren v_1 bzw. v_2 können kolinear sein. Damit geht jedoch ein Freiheitsgrad der Orientierung verloren, d. h. eine derartige Kinematik ist gleichartig zu einer solchen, bei der nur eine Drehachse definiert ist. Alle möglichen Orientierungen liegen dann auf einem Kegelmantel. Der Kegelmantel entartet zu einer Geraden, wenn zusätzlich auch Werkzeugorientierung t und v_1 bzw. v_2 kolinear werden, d. h. in diesem Spezialfall ist deshalb eine Orientierungsänderung nicht mehr möglich. Der Kegelmantel entartet zu einer Kreisfläche (d. h. es sind alle Orientierungen in einer Ebene möglich), wenn Werkzeugorientierung t und v_1 bzw. v_2 aufeinander senkrecht stehen.

Es ist zulässig, dass beide Vektoren v_1 und v_2 Null werden. Eine Orientierungsänderung ist dann allerdings nicht mehr möglich. In diesem Spezialfall wirken die eventuell von Null verschiedenen Längen l_1 und l_2 wie zusätzliche Werkzeuglängenkorrekturen, deren Komponenten in den einzelnen Achsen durch einen Wechsel der Ebene (`G17` - `G19`) nicht beeinflusst werden.

Erweiterungen der Kinematikdaten

- Möglichkeit, unmittelbar auf vorhandene Maschinenachsen zuzugreifen, um über die Rundachspositionen die Einstellung der Werkzeugträger zu definieren.
- Erweiterung auf Kinematiken mit drehbarem Werkstück bzw. auf Kinematiken mit drehbarem Werkzeug und drehbarem Werkstück.
- Möglichkeit, für die Rundachspositionen nur diskrete Werte in einem Raster zuzulassen (Hirth-Verzahnungen).

Die Erweiterungen sind kompatibel zu früheren Softwareständen und umfassen die Kinematikdatensätze von $TC_CARR18 bis $TC_CARR23.

Maschinen mit drehbarem Werkzeug

Bei Maschinen mit drehbarem Werkzeug ändert sich an der Definition der Kinematik gegenüber älteren Softwareständen nichts. Insbesondere hat der neu eingeführte Vektor l_4 keine Bedeutung. Ein eventuell von Null verschiedener Inhalt von l_4 wird nicht ausgewertet.

Der Begriff "Orientierbare Werkzeugträger" ist für die neuen Kinematiktypen, bei denen auch der Werkzeugtisch allein oder auch zusätzlich zum Werkzeug gedreht werden kann, eigentlich nicht mehr angemessen. Er wird jedoch aus Kompatibilitätsgründen beibehalten.

Die kinematischen Ketten zur Beschreibung der Maschine mit drehbarem Werkzeug (allgemeiner Fall) sind im folgenden Bild dargestellt:

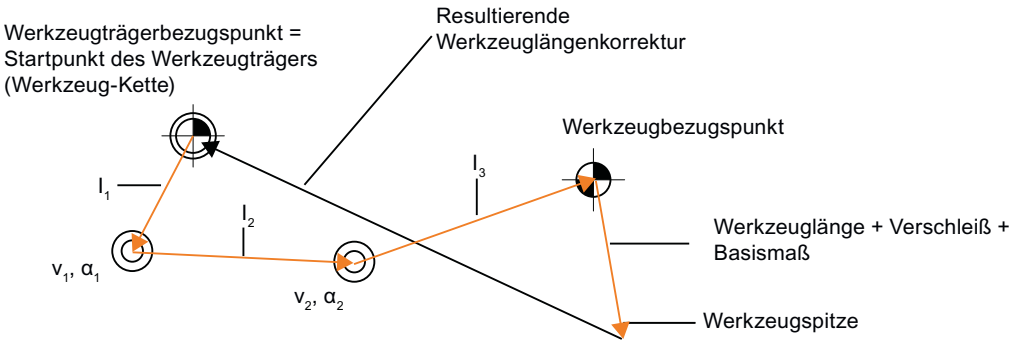


Bild 2-30 Kinematische Kette zur Beschreibung eines Werkzeuges mit Orientierung

Vektoren, die Offsets im drehbaren Kopf beschreiben, sind in Richtung von der Werkzeugspitze zum Bezugspunkt des Werkzeugträgers positiv definiert.

Für Maschinen mit drehbarem Werkzeug wird folgender Kinematiktyp definiert:

$TC_CARR23 mittels des Buchstabens T

Maschinen mit drehbarem Werkstück

Bei Maschinen mit drehbarem Werkstück hat der Vektor l_1 keine Bedeutung. Ein eventuell von Null verschiedener Inhalt wird nicht ausgewertet.

Die zugehörige kinematische Kette der Maschine mit drehbarem Werkstück ist im folgenden Bild dargestellt.

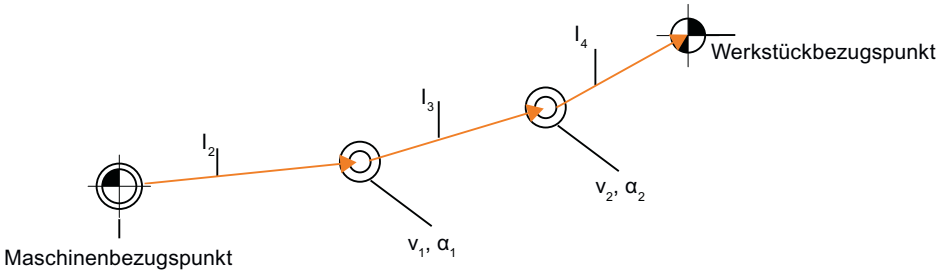


Bild 2-31 Kinematische Kette zur Beschreibung eines drehbaren Werkzeugtisches

Vektoren, die Offsets im drehbaren Tisch beschreiben, sind in Richtung vom Maschinenbezugspunkt zum Tisch positiv definiert.

Für Maschinen mit drehbarem Werkstück wird folgender Kinematiktyp definiert:

$TC_CARR23 mittels des Buchstabens P

Hinweis

Bei Maschinen mit drehbarem Werkstück wird es in der Regel sinnvoll sein, den Maschinenbezugspunkt und den Bezugspunkt des Tisches identisch zu wählen. Eine derartige Wahl der Bezugspunkte hat den Vorteil, dass sich die Lage des Werkstücknullpunkts in Grundstellung (d. h. bei nicht gedrehten Rundachsen) beim Aktivieren des drehbaren Tisches nicht verändert. Die (offene) kinematische Kette (siehe Bild) wird dann geschlossen.

In diesem Sonderfall gilt deshalb: $l_2 = - (l_3 + l_4)$

Maschinen mit aufgelöster Kinematik

Bei Maschinen mit aufgelöster Kinematik (sowohl Werkzeug als auch Werkstück sind drehbar) kann jede der beiden Komponenten nur mit einer Achse gedreht werden.

Die Kinematik des drehbaren Werkzeugs wird mit der ersten Drehachse (v_1) und den beiden Vektoren l_1 und l_2 beschrieben, der drehbare Tisch mit der zweiten Drehachse (v_2) und den beiden Vektoren l_3 und l_4. Die beiden kinematischen Teilketten der Maschine mit drehbarem Werkzeug und drehbarem Werkstück sind im folgenden Bild dargestellt.

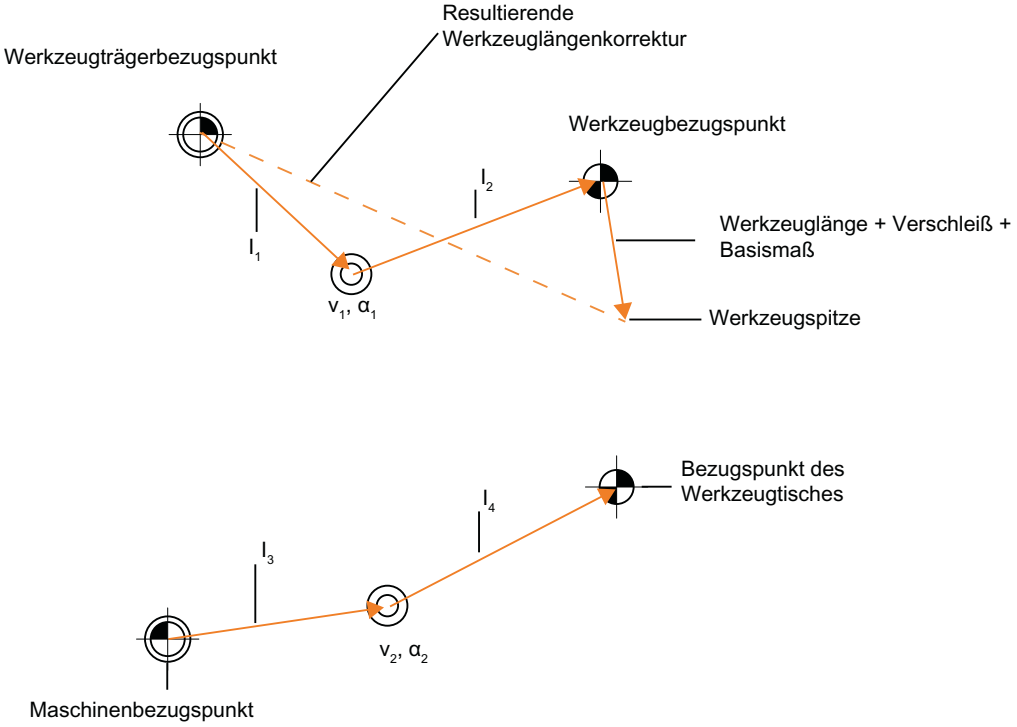


Bild 2-32 Kinematische Kette bei aufgelöster Kinematik

Für Maschinen mit drehbarem Werkzeug und drehbarem Werkstück wird folgender Kinematiktyp definiert:

$TC_CARR23 mittels des Buchstabens M (aufgelöste Kinematik)

Hinweis

Bei Maschinenkinematiken mit aufgelöster Kinematik wird es aus dem gleichen Grund wie bei Maschinen, bei denen nur der Werkzeugtisch drehbar ist, in der Regel sinnvoll sein, den Maschinenbezugspunkt und den Bezugspunkt des Tisches identisch zu wählen. Die (offene) Teilkette zur Beschreibung des Werkzeugtisches (siehe Bild) wird dann geschlossen.

In diesem Sonderfall gilt: $l_3 = - l_4$

Werkzeug drehbar Typen T und M

Bei Maschinenkinematiken mit einem drehbaren Werkzeug (Typen T und M) wirkt der orientierbare Werkzeugträgeranteil, der den Werkzeug- oder Kopfanteil (im Gegensatz zum Tischanteil) beschreibt, zusammen mit dem aktiven Werkzeug wie ein neues Gesamtwerkzeug.

Feinverschiebung

Die Offsetvektoren l_1 bis l_4 sowie die Offsets der Drehachsen v_1 und v_2 können jeweils als Summe eines Basiswertes und einer Feinverschiebung dargestellt werden. Die den Basiswerten zugeordneten Feinverschiebungsparameter ergeben sich, indem **zum Index des Basiswertes der Wert 40 addiert** wird.

Beispiel:

Dem Parameter $TC_CARR5 ist der Feinverschiebungswert $TC_CARR45 zugeordnet.

Hinweis

Zur Bedeutung der für die Feinverschiebung verfügbaren Systemvariablen $TC_CARR41 bis $TC_CARR65 siehe:

Weitere Informationen
Programmierhandbuch Arbeitsvorbereitung; Werkzeugkorrekturen

Aktivierung

Mit der folgenden Einstellung werden die Feinverschiebungswerte zu den Basiswerten addiert:

SD42974 $SC_TOCARR_FINE_CORRECTION = 1 (Feinverschiebung TCARR ein/aus)

Randbedingungen

Der Betrag der zulässigen Feinverschiebung ist begrenzt.

Der maximal zulässige Wert wird festgelegt:

Komponenten	Maschinendatum
• Komponenten der Vektoren l_1 bis l_4:	MD20188 $MC_TOCARR_FINE_LIM_LIN
• Offsets der beiden Drehachsen v_1 und v_2:	MD20190 $MC_TOCARR_FINE_LIM_ROT

Ein unzulässiger Feinverschiebungswert wird erst erkannt, wenn folgende Bedingungen auftreten:

- Ein orientierbarer Werkzeugträger wird aktiviert, der solch einen Wert enthält
- Gleichzeitig ist folgendes Settingdatum gesetzt:
  SD42974 $SC_TOCARR_FINE_CORRECTION

Beschreibung einer Drehung

Ein Datensatz zur Beschreibung einer Drehung besteht aus jeweils einem Vektor v_1 bzw. v_2 zur Angabe der Richtung der Drehachse im Grundzustand und jeweils einem Winkel α1 bzw. α2. Der Drehwinkel wird bei Drehung im Uhrzeigersinn und in Blickrichtung des Drehvektors positiv gezählt.

Die Bestimmung der beiden Werkzeugträgerwinkel α1 und α2 mittels eines Frames ist unabhängig von der aktuell angewählten aktiven Ebene (`G17` - `G19`).

Die Werkzeugorientierung in Grundstellung (beide Winkel α1 und α2 gleich Null) ist (wie im Standardfall auch):

- bei `G17` parallel zu Z.
- bei `G18` parallel zu Y.
- bei `G19` parallel zu X.

Werkzeugträger Daten zuordnen

Beispiel an einer Maschine mit drehbarem Werkzeugträger

An einer Maschine mit orientierbarem Werkzeugträger vom Kinematiktyp T ergeben sich am dargestellten Fräskopf die folgenden Einstellungen:

Komponente des Offsetvektors l_1 =	(-200, 0, 0)
Komponente des Offsetvektors l_2 =	(0, 0, 0)
Komponente des Offsetvektors l_3 =	(-100, 0, 0)
Komponente der Drehachse v_1 =	(1, 0, 0)
Komponente der Drehachse v_2 =	(-1, 0, 1)
Basismaß Bezugspunkt Werkzeug	(0, 0, 250)

Hinweis

Der Werkzeug-Bezugspunkt für das Basismaß wird durch den Referenzpunkt an der Maschine festgelegt.

Weitere Informationen zu den Bezugspunkten im Arbeitsraum siehe Funktionshandbuch "Grundfunktionen", Kapitel "Achsen, Koordinatensysteme, Frames".

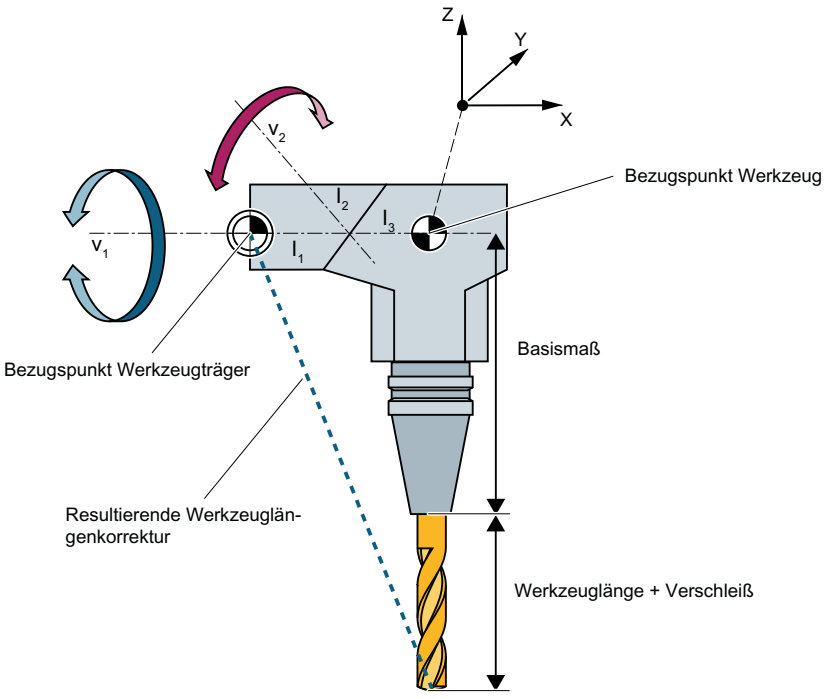


Bild 2-33 Zuordnung der Werkzeugträger-Daten

Für die folgenden Datensatzwerte wurden geeignete Annahmen getroffen:

- Die beiden Drehachsen schneiden sich in einem Punkt.
  Deshalb sind alle Komponenten von l_2 Null.
- Die erste Drehachse liegt in der x-z-Ebene, die zweite Rundachse ist parallel zur x-Achse.
  Durch diese Bedingungen sind die Richtungen von v_1 bzw. v_2 festgelegt (deren Längen sind - solange sie ungleich Null sind - ohne Bedeutung).
- Der Bezugspunkt des Werkzeugträgers liegt vom Schnittpunkt der beiden Rundachsen aus gesehen um 200 mm in negativer x-Richtung.
  Durch diese Bedingung wird l_1 festgelegt.

Zugehörige Datensatzwerte angeben

Für den abgebildeten Werkzeugträger einer Maschine mit drehbarem Werkzeugträger werden folgende zugehörige Datensatzwerte angegeben:

Beschreibung	NC-Variable	Wert
x-Komponente des Offsetvektors l_1	$TC_CARR1	- 200
y-Komponente des Offsetvektors l_1	$TC_CARR2	0
z-Komponente des Offsetvektors l_1	$TC_CARR3	0
x-Komponente des Offsetvektors l_2	$TC_CARR4	0
y-Komponente des Offsetvektors l_2	$TC_CARR5	0
z-Komponente des Offsetvektors l_2	$TC_CARR6	0
x-Komponente der Drehachse v_1	$TC_CARR7	1
y-Komponente der Drehachse v_1	$TC_CARR8	0
z-Komponente der Drehachse v_1	$TC_CARR9	0

Beschreibung	NC-Variable	Wert
x-Komponente der Drehachse v_2	$TC_CARR10	-1
y-Komponente der Drehachse v_2	$TC_CARR11	0
z-Komponente der Drehachse v_2	$TC_CARR12	1
Drehwinkel α1(in Grad)	$TC_CARR13	0
Drehwinkel α2(in Grad)	$TC_CARR14	0
x-Komponente des Offsetvektors l_3	$TC_CARR15	-100
y-Komponente des Offsetvektors l_3	$TC_CARR16	0
z-Komponente des Offsetvektors l_3	$TC_CARR17	0

Erklärungen

Bei der im Beispiel gewählten Werkzeugträgerkinematik bilden die beiden Drehachsen einen Winkel von 45°, so dass die Orientierung nicht jeden beliebigen Wert annehmen kann. Konkret sind in diesem Beispiel keine Orientierungen mit negativer X-Komponente darstellbar.

x-Komponente des Basismaßes:	0
y-Komponente des Basismaßes:	0
z-Komponente des Basismaßes:	250

Hinweis

Aus der Geometrie eines Werkzeugträgers sind die erforderlichen Daten nicht eindeutig zu bestimmen, d. h. der Anwender hat eine gewisse Wahlfreiheit, wo welche Daten abgelegt werden sollen. So besteht im Beispiel die Möglichkeit, für das Basismaß nur eine z-Komponente bis zur zweiten Achse anzugeben. In diesem Fall wäre dann l_2 nicht mehr Null, sondern enthielte die Komponenten des Abstandes dieses Punktes auf der zweiten Achse von einem weiteren Punkt auf der ersten Achse. Dieser Punkt auf der ersten Achse kann ebenfalls frei gewählt werden. Abhängig von der Wahl dieses Punktes muss dann l_1 so gewählt werden, dass der (ebenfalls frei wählbare) Bezugspunkt erreicht wird.

Allgemein gilt: Die Vektorkomponenten, die durch die Drehung einer Achse nicht verändert werden, können beliebig auf die Vektoren "vor" und "nach" der Drehung verteilt werden.

2.8.3 Werkzeugträger mit kinematischen Ketten

Werkzeugträger über kinematische Ketten modellieren

Bei Maschinen, deren Strukturen mit kinematischen Ketten definiert sind, können Maschinenteile, die als Werkzeugträger wirksam sind, ebenfalls über kinematische Ketten beschrieben werden. Identische Geometriedaten für kinematische Ketten und Werkzeugträger können gemeinsam verwaltet werden.

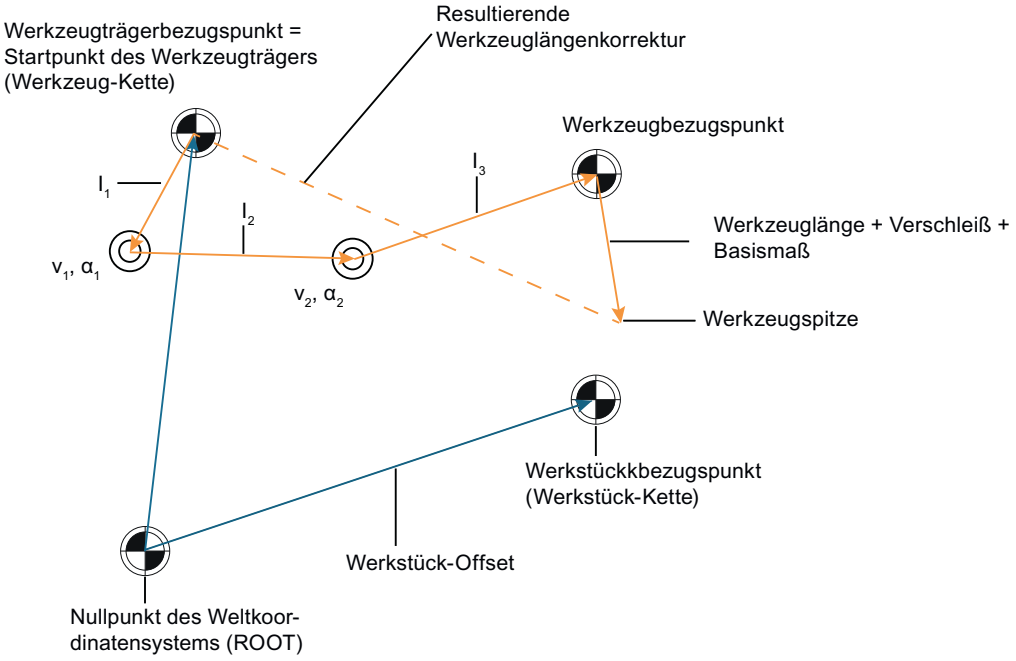
Für die Definition der Maschinenkinematik sind zwei kinematische Ketten notwendig:

- Eine Kette zeigt vom Maschinennullpunkt (Nullpunkt des Weltkoordinatensystems) zum Werkstückbezugspunkt (Part-Teil).
- Die andere Kette zeigt vom Maschinennullpunkt zum Werkzeugträgerbezugspunkt (Tool-Teil).
- Informationen zu kinematischen Ketten finden Sie im Funktionshandbuch "Sonderfunktionen" unter "K7: Kinematische Kette".

Für Werkzeugträger sind drei verschiedene Kinematiktypen möglich, die über $TC_CARR23 parametriert werden.

Folgende Grafiken zeigen beispielhaft eine Kinematik von Maschinen mit drehbarem Werkzeug (Typ T) und drehbarem Werkstück (Typ P) und die aufgelöste Kinematik (Typ M). Die aufgelöste Kinematik (Typ M) ist eine Mischung aus beiden Typen.

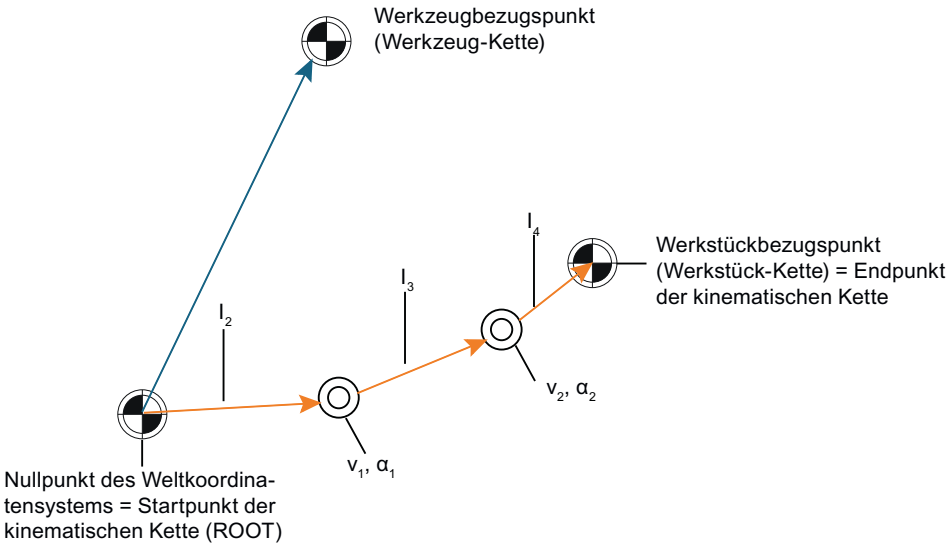
Beispiel für drehbares Werkzeug



Orange:	Kinematische Kette des Werkzeugträgers
Blau:	Kinematische Kette der Maschine

Bild 2-34 Werkzeugträger mit drehbarem Werkzeug

Beispiel für Drehbares Werkstück

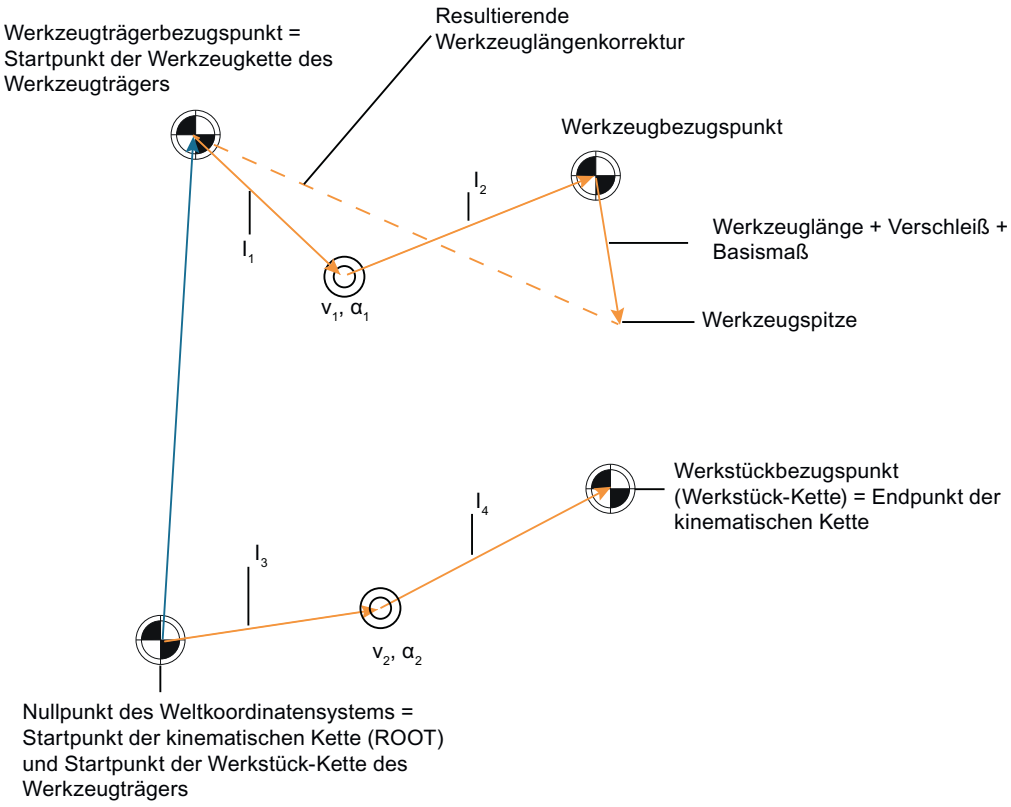


Orange: Kinematische Kette des Werkzeugträgers

Blau: Kinematische Kette der Maschine

Bild 2-35 Werkzeugträger mit drehbarem Werkstück

Beispiel für gemischte Kinematik



l_1 und l_2	Offset der Werkzeug-Kette des Werkzeugträgers
v_1	Rundachse der Werkzeug-Kette
l_3 und l_4	Offset der Werkstück-Kette des Werkzeugträgers
v_2	Rundachse der Werkstück-Kette

Bild 2-36 Gemischte Kinematiken

Die Kinematik des Werkzeugträgers wird zusammen mit der Maschinenkinematik definiert.

Definition des Werkzeugträgers:

- Definition der Eigenschaften des Werkzeugträgers mit $TC_CARR_KIN_CNTRL:
  - Werkzeugträger über kinematische Kette, Werkzeugkette und Tischkette des Werkzeugträgers schließen.
- Der Startpunkt der kinematischen Kette des Werkzeugträgers wird über die Systemvariablen $TC_CARR_KIN_TOOL_START und $TC_CARR_KIN_PART_START (Kettenelementname) eingestellt.
- Definition der Rundachsen des Werkzeugträgers mit $TC_CARR21 (Seite 94) und $TC_CARR22
- Definition des Typs des Werkzeugträgers mit $TC_CARR23 (T, P oder M)

Definition der Kinematik des Werkzeugträgers:

Der Aufbau des Part-Teils einer kinematischen Kette für einen Werkzeugträger kann z. B. wie folgt aussehen:

Definition der Linearachsen der kinematischen Kette (Geometrieachsen X, Y und Z).

Definition eines kinematischen Elements vom Typ "OFFSET" (=I_2) in X-, Y- und Z-Richtung. Entspricht $TC_CARR4 - $TC_CARR6.

Definition eines kinematischen Elements vom Typ "AXIS_ROT" (=v_1) in X-, Y- und Z-Richtung. Entspricht $TC_CARR7 - $TC_CARR9.

Definition eines Elements vom Typ "OFFSET" (=I_3) in X-, Y- und Z-Richtung. Entspricht $TC_CARR15 - $TC_CARR17.

Definition eines Elements vom Typ "AXIS_ROT" (=v_2) in X-, Y- und Z-Richtung. Entspricht $TC_CARR10 - $TC_CARR12.

Definition eines Elements vom Typ "OFFSET" (=I_4) in X-, Y- und Z-Richtung. Entspricht $TC_CARR18 - $TC_CARR20.

Ggf. Definition eines Rundachsoffsets der Achsen v1 und v2. Das Offset wird in die Systemvariable $NK_A_OFF eingetragen werden, die dem Kettenelement vom Typ "AXIS_ROT" zugewiesen ist. Entspricht $TC_CARR24 – TC_CARR25.

Es muss mindestens eine Rundachse definiert sein.

Hinweis

Vorzeichen des Rundachsoffsets

Ein positiver Wert für das Rundachsoffset der Rundachsen in $TC_CARR24 bzw. $TC_CARR25 wird in der kinematischen Kette als negativer Wert eingetragen.

Anwahl des Werkzeugträgers:

Der Werkzeugträger wird über den NC-Programmbefehl im Teileprogramm angewählt:

```
TCARR = <name>
```

Hinweis

Startpunkt der kinematischen Kette

Falls kein Startpunkt der kinematischen Kette des Werkzeugträgers (Werkstück oder Werkzeug) über TC_CARR_KIN_PART_START oder TC_CARR_KIN_TOOL_START definiert wurde, wird automatisch das ROOT-Element (Weltkoordinatensystem) als Startpunkt der Ketten definiert.

$TC_CARR_KIN_CNTRL

Mit $TC_CARR_KIN_CNTRL wird bestimmt, ob die Geometriedaten eines Werkzeugträgers aus kinematischen Kettenelementen oder aus den herkömmlichen Toolcarrierdaten ($TC_CARRxx) gelesen werden sollen. Die Systemvariable ist bitcodiert.

Syntax	Beschreibung	
$TC_CARR_KIN_CNTRL[n]	Steuert die Übernahme der Geometriedaten eines mit kinematischen Ketten definierten Maschinenmodells zur Parametrierung eines Werkzeugträger. Bei Bit0 = 1 werden die Daten aus $TC_CARRxx nicht berücksichtigt, sondern aus den kinematischen Ketten übernommen. Weiter kann eingestellt werden, wie die kinematische Kette geschlossen wird.	
	Datentyp: INT	
n	Index des Werkzeugträgers; die maximale Anzahl von Werkzeugträgern, kann über das Maschinendatum MM_NUM_TOOL_CARRIER eingestellt werden.	
Bit0=1	Der Werkzeugträger wird aus den kinematischen Kettenelementen parametriert. Folgende Daten des Werkzeugträgers, einschließlich der Feinverschiebungen, werden durch Geometriedaten aus kinematischen Kettenelementen ersetzt:	
	Herkömmlicher Werkzeugträger	Kinematische Kette
	$TC_CARR1 – TC_CARR3 (Offsetvektor l_1)	X, Y und Z-Komponente des Offsetvektors l_1
	$TC_CARR4 – TC_CARR6 (Offsetvektor l_2)	X, Y und Z-Komponente des Offsetvektors l_2
	$TC_CARR7 – TC_CARR9 (Drehachsrichtung v_1)	X, Y und Z-Komponente der Drehachse v_1
	$TC_CARR10 – TC_CARR12 (Drehachsrichtung v_2)	X, Y und Z-Komponente der Drehachse v_2
	$TC_CARR15 – TC_CARR17 (Offsetvektor l_3);	X, Y und Z-Komponente des Offsetvektors l_3
	$TC_CARR18 – TC_CARR20 (Offsetvektor l_4)	X, Y und Z-Komponente des Offsetvektors l_4
	$TC_CARR24 – TC_CARR25 (Rundachsoffsets)	
Bit0=0	Der Werkzeugträger wird aus den Daten von $TC_CARRxx parametriert.	
Bit1=1	Die Inhalt des Offsetvektors l_1 wird so geändert, dass der Endpunkt der **Werkzeug**-Kette mit dem Nullpunkt des Weldkoordinatensystems zusammen fällt (Kette wird geschlossen).	
Bit2=1	Die Inhalt des Offsetvektors l_4 wird so geändert, dass der Endpunkt der **Werkstück**-Kette mit dem Nullpunkt des Weldkoordinatensystems zusammen fällt (Kette wird geschlossen).	

$TC_CARR_KIN_TOOL_START und $TC_CARR_KIN_PART_START

Die Systemvariablen $TC_CARR_KIN_TOOL_START und $TC_CARR_KIN_PART_START definieren die Startpunkte der kinematischen Ketten, die zur Parametrierung des Tool- bzw. Part-Teils eines Werkzeugträgers verwendet werden. Falls die Systemvariablen leer sind, ist der Startpunkt der jeweiligen kinematischen Kette das Root-Element der Kinematikbeschreibung.

Die Möglichkeit, der Parametrierung nur eines Teils einer kinematischen Kette, die nicht mit dem Root-Element beginnt, ist besonders dann sinnvoll bzw. notwendig, wenn der Werkzeugträger in seiner ursprünglichen Bedeutung als "Werkzeugverlängerung“ und nicht

zur statischen Modellierung der vollständigen Transformationskinematik einer Werkzeugmaschine eingesetzt werden soll.

Syntax	Beschreibung
$TC_CARR_KIN_TOOL_START[n] = <name> $TC_CARR_KIN_PART_START[n] = <name>	Definiert den Namen der Anfangselemente der Tool- bzw. Werkstück-Kette. Datentyp: STRING
n	Index des Werkzeugträgers; die maximale Anzahl von Werkzeugträgern, kann über das Maschinendatum MM_NUM_TOOL_CARRIER eingestellt werden.

$TC_CARR_KIN_TOOL_END und $TC_CARR_KIN_PART_END

Die Systemvariablen $TC_CARR_KIN_TOOL_END und $TC_CARR_KIN_PART_END definieren die Endpunkte der kinematischen Ketten, die zur Parametrierung des Tool- bzw. Part-Teils eines Werkzeugträger verwendet werden. Mindestens einer dieser beiden Namen muss ungleich dem Nullstring sein.

Syntax	Beschreibung
$TC_CARR_KIN_TOOL_END[n] = <name> $TC_CARR_KIN_PART_END[n] = <name>	Definiert den Namen der Endelemente der Tool- bzw. Werkstück-Kette. Datentyp: STRING
n	Index des Werkzeugträgers; die maximale Anzahl von Werkzeugträgern, kann über das Maschinendatum MM_NUM_TOOL_CARRIER eingestellt werden.

Simulation mit orientierbaren Werkzeugträgern (kinematische Kette)

Hinweis

Für Simulation und Mitzeichnen Maschinendatum setzen

Für eine korrekte Ausführung von Simulation und Mitzeichnen muss das Maschinendatum MD 51226 $MNS_FUNCTION_MASK_SIM Bit 22 gesetzt werden.

Randbedingungen und Hinweise

- Wenn Systemdaten des Werkzeugträgers (TC_CARRxx) durch Daten aus kinematischen Kettenelementen ersetzt werden, wird der Inhalt der Systemvariablen $TC_CARRxx nicht verändert.
- Werkzeugträger und Transformationen, die mit kinematischen Ketten definiert wurden, können gleichzeitig aktiv sein. Dabei ist die Art der Parametrierung des Werkzeugträgers (unmittelbar aus Systemdaten TC_CARR… bzw. aus kinematischen Ketten) unerheblich.
- Die kinematischen Ketten, die zur Parametrierung eines Werkzeugträgers verwendet werden sollen, müssen auf eine Werkzeugträger-Konfiguration abbildbar sein. Falls diese Bedingung nicht erfüllt ist, wird der Alarm 14149 ausgegeben.

Werkzeugträger aus kinematischen Ketten vermessen

Beim Vermessen der Kinematik eines Werkzeugträgers und der nachfolgenden Korrektur müssen die wirksamen Offsetvektoren modifiziert werden. Da ein solcher Offsetvektor im Prinzip aus beliebig vielen Teilvektoren der parametrierenden kinematischen Kette gebildet werden kann, ist das zu korrigierende Kettenelement nicht automatisch identifizierbar.

Syntax	Beschreibung
$TC_CARR_CORR_ELEM[n, m] = <String>	Name des Elements in der kinematischen Kette, aus dem die Werkzeugträgerdaten gelesen werden sollen. Das Element wird als Korrekturelement für den Offsetvektor mit dem Index m verwendet.
	Datentyp: STRING
n	Index des Werkzeugträgers; die maximale Anzahl von Werkzeugträgern, kann über das Maschinendatum MM_NUM_TOOL_CARRIER eingestellt werden.
m	Index des Korrekturvektors (Wertebereich 0...3)

Mit der Systemvariablen $TC_CARR_CORR_ELEM[m, n] können diese Kettenelemente eindeutig bezeichnet werden. Dabei ist m der Index des Werkzeugträger-Datensatzes und n der Index des Korrekturvektors (n = 0...3]).

Länge und Richtung der Vektoren von TCARR auslesen

Über die Systemvariablen $PC_TCARR_OFFSET, $PC_TCARR_AX_VECT und $PC_TCARR_AX_OFFSET ist es möglich die aktuell wirksamen Offsetvektoren, Rundachsvektoren und Rundachsoffsets auszulesen. Für detaillierte Informationen zu Offsetvektoren, siehe Allgemeine Informationen (Seite 94).

Syntax	Beschreibung
$PC_TCARR_OFFSET[m,n] = <value>	Aktueller Wert der n-ten Vektorkomponente des Offsets l_m eines aktiven Werkzeugträgers. Bei einem aktiven Werkzeugträger sind maximal 4 Offsetvektoren (l1 bis l4) definiert. Mit dieser Systmvariable können die Vektorkomponenten ausgelesen werden.
	Datentyp: DOUBLE
m	Offsetvektor[m] (mit m = 1...4) des aktiven Werkzeugträgers.
n	Vektorkomponente n (mit 0 ≤ n ≤ 2) des Vektors, der mit dem Feldindex m ausgewählt wurde.

Syntax	Beschreibung
$PC_TCARR_AX_VECT[m,n] = <value>	Aktueller Wert der n-ten Vektorkomponente der Rundachse v_m eines aktiven Werkzeugträgers. Bei einem aktiven Werkzeugträger sind maximal 2 Rotationsachsen (v1 bis v2) definiert. Mit dieser Systmvariable können die Vektorkomponenten ausgelesen werden.
	Datentyp: DOUBLE
m	Drehachsrichtung v1, v2 mit m = 1...2 des aktiven Werkzeugträgers.
n	Vektorkomponente n (mit 0 ≤ n ≤ 3) des Vektors, der mit m ausgewählt wurde.

Syntax	Beschreibung
$PC_TCARR_AX_OFFSET[m] = <value>	Aktueller Wert des Rotationsoffsets der Rotationsachsen 1 und 2. Bei einem aktivenWerkzeugträger sind maximal 2 Rotationsachsen definiert. Mit dieser Systmvariable können die Positionen der Rundachsen bei Grundstellung ausgelesen werden.
	Datentyp: DOUBLE
m	Rotationsoffset mit m = 1...2 des aktiven Werkzeugträgers.

Werkzeugträger mit mehr als zwei Rundachsen

Bei Werkzeugträgern, die mit mehr als zwei Rundachsen definiert sind, wird die dritte Achse nicht berücksichtigt. Sie beschreibt einen konstanten Winkel (Knick) zwischen den beiden Drehachsen. Die Achse v_2 im Beispiel wird nicht berücksichtigt. Statt dessen wird der Offset l_2 gebildet.

Die relevanten Rundachsen werden über die Systemvariable $TC_CARR_KIN_ROTAX_NAME in der Reihenfolge definiert, in der Sie in der Kette auftreten. Bis zu zwei Rundachsen können definiert werden.

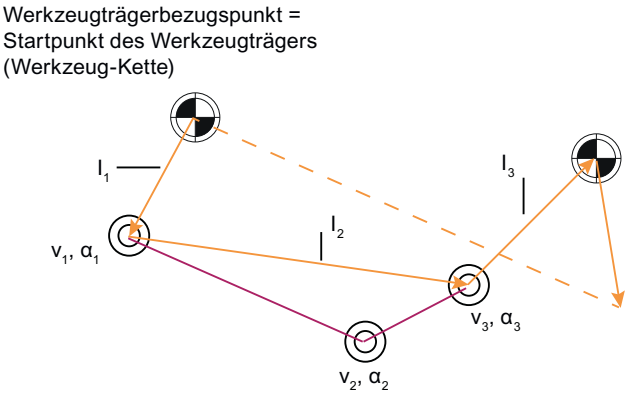


Bild 2-37 Orientierbarer Werkzeugträger mit 3 Drehachsen

Die Rundachsen sind keine Orientierungsachsen, wie sie für die 5-Achs-Transformation verwendet werden.

Rotationsachsen definieren (mehr als zwei Rundachsen)

Wenn ein Werkzeugträger mehr als zwei Rotationsachsen beinhaltet, werden über die relevanten Achsen über die Systemvariable $TC_CARR_KIN_ROTAX_NAME definiert. Dazu wird der Name des Kettenelements in die Systemvariable eingetragen. Die maximal zwei Rundachsen müssen in der Reihenfolge definiert werden, in der sie in der kinematischen Kette definiert sind. Die Einträge müssen mit dem Index "0" beginnen.

Syntax	Beschreibung
$TC_CARR_KIN_ROTAX_NAME[n, m] = <String>	Name des Elements in der kinematischen Kette, das als Rundachse definiert werden soll.
	Datentyp: STRING

Syntax	Beschreibung
n	Index des Werkzeugträgers; die maximale Anzahl von Werkzeugträgern, kann über das Maschinendatum MM_NUM_TOOL_CARRIER eingestellt werden.
m	Index der Rundachsen (Wertebereich 0...1)

Beispiel

Ein Teileprogramm für einen Werkzeugträger über eine kinematische Kette finden Sie unter: Beispiel Orientierbare Werkzeugträger über kinematische Kette (Seite 140).

2.8.4 Maschinenvermessung für Werkzeugträger

2.8.4.1 Orientierbare Werkzeugträger nach Maschinenvermessung modifizieren (CORRTC)

Mit der Funktion `CORRTC` können gemessene kinematische Kettenelemente eines Toolcarriers in spezielle Korrekturelemente geschrieben werden.

Hinweis

Die mit der Funktion CORRTC eingeschriebenen Korrekturwerte sind nicht sofort in der Transformation wirksam. Erst nach Transformationsabwahl, NEWCONF und Transformationsanwahl sind die Korrekturwerte wirksam.

Syntax

```
<_Corr_Status> = CORRTC(<_Corr_Vect>, <_Corr_Index>, <_Corr_Mode>,
[ <_No_Alarm>])
```

Bedeutung

CORRTC:	Funktionsaufruf		
<_Corr_Status>:	Rückgabewert der Funktion		
	Datentyp:	INT	
	Werte:	0	Die Funktion wurde ohne Fehler ausgeführt.
		1	Es ist kein Toolcarrier aktiv.
		2	Der aktive Toolcarrier wurde nicht mit kinematischen Ketten definiert.
		10	Der Aufrufparameter <_Corr_Index> ist negativ.
		11	Der Aufrufparameter <_Corr_Mode> ist negativ.
		12	Ungültiger Verweis auf den Abschnitt einer Teilkette (_CORR_INDEX).
		13	In dem Abschnitt, auf den mit dem Parameter _CORR_INDEX verwiesen wird, ist kein Korrekturelement definiert ($TC_CARR_CORR_ELEM).
		20	Die 100er-Stelle von _CORR_MODE ist ungültig. Es sind nur die Werte 0 und 1 zulässig.
		21	Die 1000er-Stelle von _CORR_MODE ist ungültig. Es sind nur die Werte 0 und 1 zulässig.
		30	Bei der Korrektur eines Offsetvektors ist die Abweichung gegenüber dem aktuellen Wert in mindestens einer Koordinate größer als der durch das Settingdatum SD41612 $SN_CORR_TOCARR_LIN_MAX vorgegebene Maximalwert.
		31	Der Versuch, eine Systemvariable zu beschreiben, wurde wegen fehlender Schreibrechte abgewiesen.
<_Corr_Vect>:	Korrekturvektor Der Inhalt des Korrekturvektors wird durch die nachfolgenden Parameter <Corr_Index> und <_Corr_Mode> definiert. Ist <_Corr_Status> = 30, wird der Inhalt des Vektors überschrieben (s. o.).		
	Datentyp:	REAL	
<_Corr_Index>:	Bezeichnet den Abschnitt, für den der Richtungsvektor des Korrekturelements korrigiert werden soll.		
	Datentyp:	INT	

<_Corr_Mode>:	Korrekturmodus		
	Datentyp:	INT	
	Der Parameter <Corr_Mode> ist dezimalcodiert (1er- bis 1000er-Stelle):		
	1er-Stelle:	Reserviert	
	10er-Stelle:	Bestimmt, wie das Korrekturelement, auf das der Inhalt von <_Corr_Index> verweist, modifiziert werden soll.	
		xx**0**x	Der Korrekturvektor wird unmittelbar in das Korrekturelement geschrieben. Diese Variante kann dazu verwendet werden, das Korrekturelement unmittelbar zu beschreiben, ohne dass der Index <n> des betreffenden Systemdatums ($NK_OFF_DIR[<n>, ...]) bekannt sein muss.
		xx**1**x	Wie 0, jedoch mit dem Unterschied, dass der übergebene Korrekturwert in Weltkoordinaten interpretiert wird. Ein Unterschied zwischen den Varianten 0 und 1 kann immer dann entstehen, wenn die kinematische Kette in Grundstellung (Positionen aller Rundachsen gleich 0) weitere Drehungen enthält.
		xx**2**x	Wie 1, jedoch mit dem Unterschied, dass sich der Korrekturwert auf den gesamten Abschnitt bezieht, d. h., in das Korrekturelement wird ein solcher Wert eingetragen, dass der gesamte Abschnitt die durch den Korrekturwert definierte Länge erreicht.
	100er-Stelle:	Bestimmt, wie der Inhalt des Parameters <_Corr_Vect> zu interpretieren ist.	
		x**0**xx	Der übergebene Korrekturvektor <_Corr_Vect> enthält die gesamte neue Länge des Korrekturelements bzw. des Abschnitts, auf den <_Corr_Index> in Verbindung mit der 10er-Stelle von <_Corr_Mode> verweist (absolute Korrektur).
		x**1**xx	Der übergebene Korrekturvektor <_Corr_Vect> enthält nur die Differenz gegenüber der aktuellen Länge des Korrekturelements bzw. des Abschnitts, auf den <_Corr_Index> in Verbindung mit der 10er-Stelle von <_Corr_Mode> verweist (inkrementelle Korrektur).
	1000er-Stelle:	Bestimmt, ob die maximal zulässige Korrektur durch das Settingdatum $SN_CORR_TOCARR_LIN_MAX begrenzt werden soll oder nicht.	
		0xxx	Die Grenzwertüberwachung ist aktiv.
		1xxx	Die Grenzwertüberwachung wird unterdrückt.
<_No_Alarm>:	Verhalten im Fehlerfall (Rückgabewert > 0) (**optional**)		
	Datentyp:	BOOL	
	Wert:	FALSE (Default)	Im Fehlerfall wird die Programmabarbeitung angehalten und ein Alarm wird ausgegeben.
		TRUE	Im Fehlerfall wird die Programmabarbeitung nicht angehalten und es wird kein Alarm angezeigt. Anwendungsfall: Anwenderspezifische Reaktion entsprechend Rückgabewert

Weitere Informationen zu CORRTC

Die kinematische Struktur eines Toolcarriers wird durch eine (Typ T und Typ P) oder zwei (Typ M) kinematische Ketten (Teilketten) beschrieben, die vom dazugehörigen Bezugspunkt Maschinenbezugspunkt oder Werkzeugträgerbezugspunkt) ausgehen. Eine der beiden Ketten, die Werkzeug-Kette, endet am Bezugspunkt des Werkzeugs, die andere, die Werkstück-Kette im Nullpunkt des Basiskoordinatensystems.

Die Funktion CORRTC schreibt Hebelarmlängen und Achsrichtungen bei Maschinen mit Orientierungstransformation in spezielle Korrekturelemente. Eine kinematische Kette wird unter anderem mit Elementen vom Typ OFFSET beschrieben, die über $NK_TYPE definiert werden.

CORRTC arbeitet mit Abschnitten

Die beiden Teilketten können sich jeweils in maximal vier Abschnitte aufteilen:

- Abschnitt 1 beginnt am Startpunkt der Kette und endet an der ersten Orientierungsachse.
- Abschnitt 2 ist der Abschnitt zwischen Orientierungsachse 1 und Orientierungsachse 2.
- Abschnitt 3 ist der Abschnitt zwischen Orientierungsachse 2 und dem Ende der Kette.

Die folgende Grafik zeigt einen orientierbaren Werkzeugträger mit 2 Rundachsen.

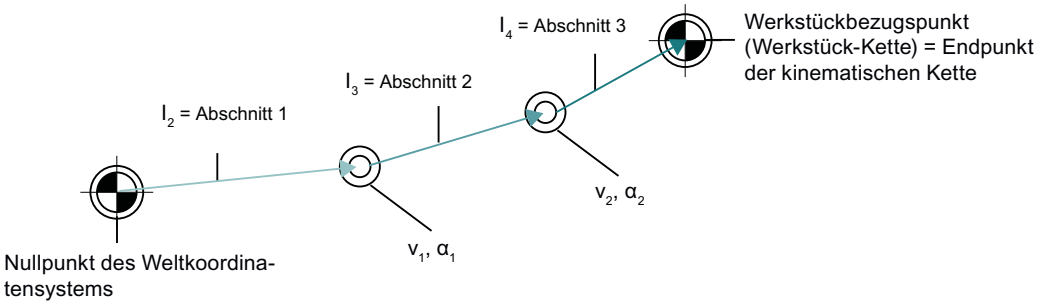


Bild 2-38 CORRTC Beispiel

Die Abschnitte sind eindeutig definiert: Durchläuft man die kinematische Teil-Kette von ihrem Startpunkt bis zum Endpunkt, so hat der erste Abschnitt den Index 1, der nächste den Index 2 usw.

Korrekturelemente

Auf jeweils ein konstantes kinematisches Kettenelement (Kettenelement des Typs $NK_TYPE[<n>] = "OFFSET") in jedem dieser Abschnitte kann mit der Systemvariablen $TC_CARR_CORR_ELEM [, 0 ... 3] verwiesen werden. In diese so bezeichneten Elemente werden mithilfe der Funktion CORRTC die Korrekturwerte geschrieben, die bei der Maschinenvermessung ermittelt wurden.

Die Reihenfolge der Verweise in $TC_CARR_CORR_ELEM[m, 0..3] muss mit den oben beschriebenen Abschnitten korrespondieren, d.h. in $TC_CARR_CORR_ELEM[m, 0] kann nur ein Kettenelement stehen, das zu dem Offsetvektor l1 gehört, usw..

Der Bezugswert ist dabei immer der entsprechende Wert, der in dem beim Aufruf von CORRTC aktiven Toolcarrier wirksam ist. Nach der Anwahl des Toolcarriers veränderte Inhalte der

Kinematikdaten in der Datenhaltung haben keinen Einfluss auf die Wirkungsweise der Funktion CORRTC.

2.8.5 Schrägbearbeitung mit 3 + 2 Achsen

Funktionale Beschreibung

Schrägbearbeitung mit 3 + 2 Achsen beschreibt die Erweiterung des Konzepts der orientierbaren Werkzeugträger sowie die Übertragung dieses Konzepts auf Maschinen mit drehbarem Werkzeugtisch auf solche Maschinen, bei denen die Orientierungen von Werkzeug und Tisch gleichzeitig veränderbar ist.

Die Funktion "Schrägbearbeitung mit 3 + 2 Achsen" dient zur Bearbeitung von Flächen im Raum, die gegenüber den Hauptebenen [X-Y-Ebene (`G17`), Z-X-Ebene (`G18`), Y-Z-Ebene (`G19`)] beliebig gedreht sind.

Eine entsprechende Orientierung des Werkzeugs relativ zum Werkstück kann erreicht werden, indem entweder das Werkzeug, das Werkstück oder sowohl Werkzeug als auch Werkstück gedreht wird.

Die Software berechnet dabei automatisch die sich aus den Werkzeuglängen, Hebelarmen und Drehachswinkeln erforderlichen Ausgleichsbewegungen. Dabei wird stets davon ausgegangen, dass zunächst die erforderliche Orientierung eingestellt und während eines Bearbeitungsganges, z. B. Fräsen einer Tasche auf einer schrägen Ebene, nicht mehr verändert wird.

Weiterhin werden folgende 3 Funktionen beschrieben, die bei der Schrägbearbeitung erforderlich sind:

- **Wegprogrammierung** in Richtung der Werkzeugorientierung unabhängig von einem aktiven Frame
- Definition einer **Framedrehung** durch Angabe der Raumwinkel
- Definition des **Rotationsanteils in Werkzeugrichtung** im programmierbaren Frame unter Erhaltung der übrigen Frameanteile

Abgrenzung zur 5-Achs-Tranformation

Wird die Funktionalität benötigt, dass die Werkzeugspitze (TCP-Tool Center Point) bei einer Umorientierung in Bezug auf das Werkstück auch während der Interpolation unverändert bleibt, ist die 5-Achs-Software erforderlich.

Weitere Erläuterungen zur 5-Achs-Transformation siehe:

Weitere Informationen
Funktionshandbuch Transformationen; Mehrachstransformation

Festlegung der orientierbaren Werkzeugträger

Der orientierbare Werkzeugträger wird durch eine allgemeine 5-Achs-Kinematik nachgebildet, die durch einen Datensatz im Werkzeugkorrekturspeicher mit insgesamt 33 REAL-Werten beschrieben wird. Bei einem Werkzeugträger (z. B. einem Fräskopf), der über zwei rotatorische Achsen zur Orientierungseinstellung verfügt, sind 31 dieser Werte konstant.

In der aktuellen SW-Version wird ein Datensatz im Werkzeugkorrekturspeicher mit insgesamt 47-Real-Werte beschrieben. Beim Werkzeugträger, der über zwei rotatorische Achsen zur Orientierungseinstellung verfügt, sind 45 dieser Werte konstant.

Die restlichen beiden Werte sind variabel und dienen zur Festlegung der Orientierung. Die konstanten Werte beschreiben Offsets und die Richtungen und Einstellmöglichkeiten der Drehachsen, die variablen Werte beschreiben die Winkel der Drehachsen.

2.8.6 Maschine mit drehbarem Werktisch

Systemvariablen

Bei der Berechnung der aktiven Werkzeuglänge mit `TCOABS` wurde bisher immer auf die in $TC_CARR13 bzw. $TC_CARR14 abgelegten Winkel zugegriffen. Das gilt nach wie vor, wenn $TC_CARR21 bzw. $TC_CARR22 nicht auf Rundachsen verweisen. Enthält $TC_CARR21 bzw. $TC_CARR22 einen Verweis auf eine Rundachse im Kanal, wird als Winkel anstelle des Eintrags in $TC_CARR13 bzw. $TC_CARR14 die Achsposition der betreffenden Achse am Beginn des aktuellen Satzes verwendet.

Eine gemischte Betriebsart ist zulässig, d. h. die Winkel können für eine Achse aus dem Eintrag der Systemvariablen $TC_CARR13 bzw. $TC_CARR14 und für die andere Achse aus der Position einer Kanalachse ermittelt werden.

Damit ist es möglich, bei Maschinen, bei denen die Achsen innerhalb der NC bekannt sind, die zur Einstellung des orientierbaren Werkzeugträgers dienen, direkt auf deren Positionen zuzugreifen, während dazu bisher beispielsweise die Systemvariable $AA_IM[Achse] gelesen und das Ergebnis der Leseoperation in $TC_CARR13/14 geschrieben werden musste. Damit entfällt insbesondere auch der implizite Vorlaufstopp beim Lesen der Achspositionen.

MD20180

Die Rundachsposition wird mit ihrem programmierten bzw. berechneten Wert verwendet wenn das Maschinendatum:

MD20180 $MC_TOCARR_ROT_ANGLE_INCR[i] = 0 (Rundachseninkrement des orientierbaren Werkzeugträgers)

Ist dieses Maschinendatum dagegen ungleich Null, so wird die Position des nächstliegenden Rasterpunktes verwendet, die sich bei geeignet gewähltem ganzzahligem n aus der folgenden Gleichung ergibt:

φ = $MC_TOCARR_ROT_ANGLE_OFFSET[i] + n * $MC_TOCARR_ROT_ANGLE_INCR[i]

Diese Funktionalität wird dann benötigt, wenn die Rundachsen indexiert werden müssen und deshalb nicht beliebige Positionen einnehmen können (z. B. bei Hirth-Verzahnungen). In der

Systemvariablen $P_TCANG[i] wird der gerasterte Wert, in der Systemvariablen $P_TCDIFF[i] die Differenz zwischen exaktem und gerastertem Wert geliefert.

Orientierung Frame TCOFR

Bei `TCOFR` (Bestimmung der Winkel aus der durch einen aktiven Frame definierten Orientierung) erfolgt die Rasterung nach der Bestimmung der Winkel aus der aktiven Framedrehung. Ist die verlangte Orientierung aufgrund der Maschinenkinematik nicht einstellbar, wird die Bearbeitung mit einem Alarm abgebrochen. Das gilt auch dann, wenn die Sollorientierung sehr nahe bei einer erreichbaren Orientierung liegt. Insbesondere kann in einer solchen Situation der Alarm durch die Winkelrasterung nicht verhindert werden.

TCARR Frameverschiebung

Eine Frameverschiebung als Folge eines Werkzeugträgerwechsels wird sofort bei der Anwahl mit `TCARR=...` wirksam. Dagegen wird eine Änderung der Werkzeuglänge nur dann sofort wirksam, wenn ein Werkzeug aktiv ist.

TCOFR / TCOABS Framedrehung

Eine Framedrehung wird mit der Aktivierung nicht bewirkt, bzw. eine bereits wirksame Drehung wird nicht verändert. Die Stellung der Rundachsen, die für die Berechnung verwendet wird, wird wie im Fall T (nur das Werkzeug ist drehbar) abhängig vom G-Befehl `TCOFR` / `TCOABS` aus dem Drehanteil eines aktiven Frames bzw. aus den Einträgen $TC_CARRn ermittelt.

Durch die Aktivierung eines Frames verändert sich die Position im Werkstückkoordinatensystem entsprechend, ohne dass es zu einer Ausgleichsbewegung der Maschine kommt. Die Verhältnisse sind im folgenden Bild dargestellt:

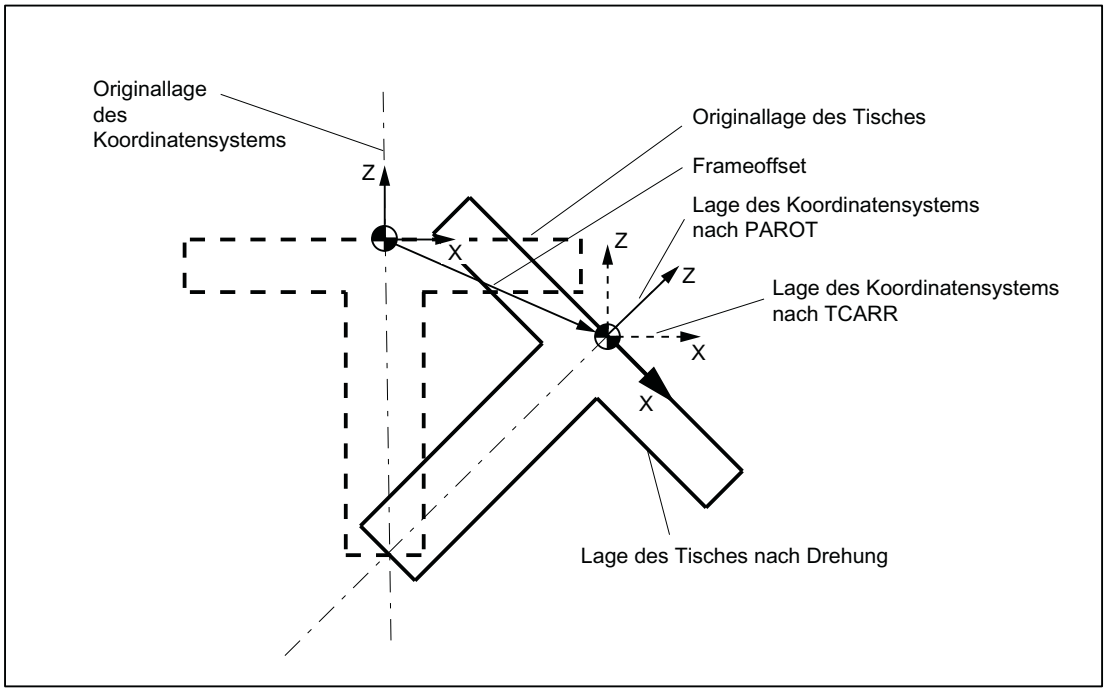


Bild 2-39 Nullpunktverschiebung bei Aktivierung eines drehbaren Werkzeugtisches mit `TCARR`

Beispiel

Bei der Maschine im Bild zeigt die Drehachse des Tisches in die positive Y-Richtung. Der Tisch ist um +45 Grad gedreht. Mit PAROT wird ein Frame definiert, der ebenfalls eine Drehung von 45 Grad um die Y-Achse beschreibt. Das gegenüber der Außenwelt nicht gedrehte Koordinatensystem (im Bild mit "Lage des Koordinatensystems nach TCARR" gekennzeichnet) ist gegenüber dem mitgeführten Koordinatensystem (Lage nach PAROT) aber um -45 Grad gedreht. Wird dieses Koordinatensystem z. B. mit ROT Y-45 definiert und anschließend der Werkzeugträger bei aktivem TCOFR angewählt, wird für die Drehachse des Werkzeugträgers ein Winkel von +45 Grad ermittelt.

Drehbarer Tisch

Bei drehbaren Tischen (Kinematiktypen P und M) hat die Aktivierung mit TCARR zunächst ebenfalls keine Drehung des Koordinatensystems zur Folge (siehe Bild), d.h. der Nullpunkt des Koordinatensystems verschiebt sich zwar in Bezug auf die Maschine und bleibt fest in Bezug auf den Nullpunkt des Werkstücks, die Orientierung bleibt jedoch unverändert raumfest.

Aktivierung Kinematiktyp P und M

Bei Kinematiken des Typs P und des Typs M wird bei der Anwahl eines Werkzeugträgers ein additiver Frame aktiviert (Tischoffset des orientierbaren Werkzeugträgers), der die Verschiebung des Nullpunktes als Folge der Drehung des Tisches berücksichtigt.

Die Nullpunktverschiebung kann in ein eigenes Systemframe $P_PARTFR geschrieben werden. Dazu muss das Bit 2 gesetzt werden im Maschinendatum:

MD28082 $MC_MM_SYSTEM_FRAME_MASK (Systemframes (SRAM))

Das durch das folgende Maschinendatum bezeichnete Basisframe wird dann für die Nullpunktverschiebung nicht mehr benötigt:

MD20184 $MC_TOCARR_BASE_FRAME_NUMBER (Nummer des Basisframes für Aufnahme des Tischoffsets)

Aktivierung Kinematiktyp M

Bei Kinematiken des Typs M (Werkzeug und Tisch sind jeweils um eine Achse drehbar), bewirkt die Aktivierung eines Werkzeugträgers mit TCARR gleichzeitig eine entsprechende Änderung der effektiven Werkzeuglänge (falls ein Werkzeug aktiv ist) und der Nullpunktverschiebung.

Drehungen

Je nach Bearbeitungsaufgabe muss beim Einsatz drehbarer Werkzeugträger bzw. Tische nicht nur eine Nullpunktverschiebung (sei es als Frame oder als Werkzeuglänge), sondern auch eine Drehung berücksichtigt werden. Die Aktivierung eines orientierbaren Werkzeugträgers führt jedoch in keinem Fall unmittelbar zu einer Drehung des Koordinatensystems.

TOROT

Ist nur das Werkzeug drehbar, kann ein Frame mittels TOFRAME bzw. TOROT definiert werden, dessen Z-Achse in Werkzeugrichtung zeigt.

PAROT

Wird ein Koordinatensystem benötigt, das in Bezug auf das Werkstück fest ist, d. h. gegenüber der Originallage nicht nur verschoben sondern auch entsprechend der Tischdrehung gedreht ist, so kann analog zur Situation bei drehbarem Werkzeug mit PAROT eine entsprechende Drehung aktiviert werden.

Mit PAROT bleiben Translationen, Skalierungen und Spiegelungen im aktiven Frame erhalten, der Rotationsanteil wird jedoch durch den Rotationsanteil eines orientierbaren Werkzeugträgers, der dem Tisch entspricht, gedreht.

PAROT und TOROT berücksichtigen in den Fällen, in denen der Tisch bzw. das Werkzeug mit zwei Drehachsen orientiert werden, die gesamte Orientierungsänderung. Bei gemischten Kinematiken wird jeweils nur der entsprechende, durch eine Drehachse verursachte Anteil berücksichtigt. Damit ist es z. B. bei der Verwendung von TOROT möglich, ein Werkstück so zu drehen, dass eine schräge Ebene parallel zur raumfesten X-Y-Ebene liegt, bei der Bearbeitung aber eine Drehung des Werkzeugs zu berücksichtigen ist, wenn beispielsweise Bohrungen angebracht werden müssen, die nicht senkrecht zu dieser Ebene sind.

Der Sprachbefehl PAROT wird nicht abgelehnt, wenn kein orientierbarer Werkzeugträger aktiv ist. Er bewirkt keine Änderung im programmierten Frame.

Hinweis

Für weitere Erläuterungen zu den Funktionen TCARR und TOROT sowie PAROT bezüglich kanalspezifischer Systemframes siehe Funktionshandbuch "Grundfunktionen", Kapitel "Achsen, Koordinatensysteme, Frames".

2.8.7 Vorgehen bei Benutzung der orientierbaren Werkzeugträger

Neuanlegen eines Werkzeugträgers

Einstellen der Anzahl verfügbarer Werkzeugträgerdatensätze

Die Anzahl der verfügbaren Werkzeugträgerdatensätze in der Steuerung wird festgelegt mit dem Maschinendatum:

MD18088 $MN_MM_NUM_TOOL_CARRIER (Maximale Anzahl definierbarer Werkzeugträger)

Die Anzahl ist nach folgender Regel einzustellen::

MD18088 $MN_NUM_TOOL_CARRIER =	
	<Größte benötigte Anzahl von Werkzeugträgern in **einem** Kanal> *
	<Anzahl der parametrierten Kanäle>

Die im Maschinendatum parametrierte Anzahl an Werkzeugträgern wird dann steuerungsintern gleichmäßig auf die parametrierten Kanäle bzw. TO-Einheiten aufgeteilt:

<Verfügbare Anzahl an Werkzeugträgern pro Kanal bzw. TO-Einheit> =	
	MD18088 $MN_NUM_TOOL_CARRIER /
	<Anzahl der parametrierten Kanäle>

Hinweis

Weitere Erläuterungen zur Definition und Zuordnung einer TO-Einheit zu einem Kanal durch das Maschinendatum MD28085 $MC_MM_LINK_TOA_UNIT finden sich in:

Weitere Informationen
Funktionshandbuch Basisfunktionen; Speicherkonfigurationen

Nullsetzen der Werkzeugträgerdaten:

Zum Nullsetzen aller Werte aller Datensätze kann der Befehl $TC_CARR1[0] = 0 verwendet werden.

Einzelne Werkzeugträgerdatensätze können selektiv mit dem Sprachbefehl `DELTC` bzw. über den PI-Dienst `_N_DELTCAR` gelöscht werden.

Zugriff auf die Daten eines Werkzeugträgers

- Teileprogramm
  - Schreiben: `$TC_CARR<n>[<m>] = <Wert>`
    Der Parameter <n> des Werkzeugträgers <m> wird mit dem Wert <Wert> beschrieben.
  - Lesen: `<Wert> = $TC_CARR<n>[<m>]`
    Der Parameter <n> des Werkzeugträgers <m> wird gelesen. Ist der referenzierte Werkzeugträger nicht definiert, wird ein Alarm angezeigt.
- BTSS-Schnittstelle
  Die Parameter eines orientierbaren Werkzeugträgers können über die Systemvariablen $P_TCANG[<n>] gelesen und geschrieben werden.

Datensicherung

Die genannten Systemvariablen werden im Rahmen der allgemeinen NC-Datensicherung gesichert.

Werkzeugträger anwählen

Ein Werkzeugträger mit der Nummer <m> wird mit dem NC-Programmbefehl `TCARR = <m>` angewählt (`TCARR`**T**ool **Carr**ier).

Mit `TCARR = 0` wird ein aktiver Werkzeugträger abgewählt.

Neues Werkzeug oder neuer Werkzeugträger

Ein neu aktiviertes Werkzeug wird so behandelt, als wäre es auf den aktiven Werkzeugträger montiert.

Ein neuer Werkzeugträger wird sofort bei seiner Programmierung wirksam, ein Wechsel oder eine Neuprogrammierung des aktiven Werkzeugs ist nicht notwendig. Werkzeugträger(-

nummer) und Werkzeug(-nummer) sind voneinander unabhängig und können beliebig miteinander kombiniert werden.

Werkzeugträger aus G-Befehl der Gruppe 42

Werkzeugorientierung absolut TCOABS (**T**ool **C**arrier **O**rientation **ABS**olute):

Die Werkzeugorientierung wird explizit bestimmt, wenn die entsprechenden Werte in die Systemvariablen $TC_CARR13 bzw. $TC_CARR14 eingetragen werden und in der G-Gruppe 42 der G-Befehl TCOABS aktiviert wird.

Werkzeugorientierung Frame TCOFR (**T**ool **C**arrier **O**rientation **FR**ame):

Ebenso kann die Werkzeugorientierung automatisch aus der aktuellen Orientierung eines aktiven Frames bei Werkzeuganwahl bestimmt werden, wenn bei der Werkzeugträgeranwahl in der G-Gruppe 42 einer der folgenden G-Befehle aktiv ist:

- TCOFR bzw. TCOFRZ
  Der orientierbare Werkzeugträger wird so eingestellt, dass das Werkzeug in Z-Richtung zeigt.
- TCOFRX
  Der orientierbare Werkzeugträger wird so eingestellt, dass das Werkzeug in X-Richtung zeigt.
- TCOFRY
  Der orientierbare Werkzeugträger wird so eingestellt, dass das Werkzeug in Y-Richtung zeigt.

TCOFR bringt zum Ausdruck, dass bei Bearbeitung auf einer schrägen Fläche implizit die Werkzeugkorrekturen so berücksichtigt werden, als ob das Werkzeug auf dieser Fläche senkrecht stünde.

Hinweis

Die Werkzeugorientierung ist nicht starr an die Frameorientierung gebunden. Bei aktivem Frame und bei aktivem G-Befehl TCOABS kann ein Werkzeug angewählt werden, bei dem die Orientierung des Werkzeugs von der Orientierung des aktiven Frames unabhängig ist.

Nach der Werkzeuganwahl kann der Anwender den Frame wechseln, welcher die Komponenten der Werkzeuglängenkorrektur nicht beeinflusst. Es ist dann auch nicht mehr sichergestellt, dass das Werkzeug senkrecht auf der Bearbeitungsebene steht. Der Anwender sollte deshalb die Einhaltung der gewollten Werkzeugorientierung auf einer schrägen Fläche vorher kontrollieren.

Die für die Werkzeuglängenberechnung verwendete Werkzeugorientierung wird bei aktivem TCOFR usw. bei einem Werkzeugträgerwechsel immer neu aus dem zu diesem Zeitpunkt aktiven Frame ermittelt.

Werkzeugträger aus G-Befehl der Gruppe 53

Die Werkzeugorientierung kann mit Hilfe der G-Befehle der Gruppe 53 (TOFRAME, TOROT usw.) einen Frame so zu definieren, dass eine Achsrichtung (Z, Y oder X) in diesem Frame gleich der aktuellen Werkzeugorientierung ist.

Für die Werkzeugorientierung ist zum Zeitpunkt des Aufrufs von TOFRAME der aktive G-Befehl der Gruppe 6 (G17 - G19) entscheidend.

Ist kein Werkzeugträger aktiv, oder ist ein Werkzeugträger aktiv, der keine Orientierungsänderung des Werkzeugs bewirkt, dann ist die Z-Richtung im neuen Frame:

- bei G17 gleich der alten Z-Richtung.
- bei G18 gleich der alten Y-Richtung.
- bei G19 gleich der alten X-Richtung.

Bei drehenden Werkzeugträgern werden diese Richtungen entsprechend modifiziert. Entsprechendes gilt für die neuen X- bzw. Y-Richtungen.

Statt TOFRAME bzw. TOROT kann einer der G-Befehle TOFRAMEX, TOFRAMEY, TOROTX oder TOROTY verwendet werden. Dann sind die Bedeutungen der Achsen entsprechend vertauscht.

Gruppenwechsel

Ein Wechsel des G-Befehls aus der Gruppe 42 (TCOABS, TCOFR usw.) bewirkt eine Neuberechnung der relevanten Werkzeuglängenkomponenten.

Die in den Werkzeugträgerdaten abgelegten (programmierten) Drehwinkel werden nicht beeinflusst, so dass bei einem Wechsel von TCOFR nach TCOABS wieder die ursprünglich in den Werkzeugträgerdaten abgelegten Winkel wirksam werden.

Drehwinkel (α_1 bzw. α_2) lesen:

Die aktuell zur Berechnung der Orientierung verwendeten Winkel können über die Systemvariable $P_TCANG[n] mit n = 1 oder n = 2 gelesen werden.

Gibt es zu einer bestimmten Orientierung zwei zulässige Lösungen (ein zweites gültiges Winkelpaar), so kann auf die entsprechenden Werte mit $P_TCANG[3] bzw. $P_TCANG[4] zugriffen werden. Die Anzahl der gültigen Lösungen 0 bis 2 kann mit $P_TCSOL gelesen werden.

Werkzeugradienkorrektur mit CUT2D oder CUT3DFS:

Bei der Werkzeugradiuskorrektur wird die aktuelle Werkzeugorientierung berücksichtigt, wenn in der G-Gruppe 22 (Werkzeugkorrekturtyp) entweder CUT2D oder CUT3DFS aktiv ist.

Bei nicht drehenden Werkzeugträgern ist das Verhalten nur von der aktiven Ebene vom G-Befehl der Gruppe 6 (G17 - G19) abhängig und somit identisch zum bisherigen Verhalten.

Alle anderen Werkzeugkorrektur Typen:

Das Verhalten bei allen anderen Werkzeugkorrekturtypen ist unverändert.

Insbesondere wird bei CUT2DF und bei CUT3DFF die für die Werkzeugradiuskorrektur verwendete Korrekturebene unabhängig von der aktuellen Werkzeugorientierung aus dem aktiven Frame ermittelt. Dabei wird die aktive Ebene (G17 - G19) berücksichtigt, so dass das Verhalten identisch zum bisherigen ist.

Die übrigen beiden G-Befehle der Gruppe 22 CUT3DC und CUT3DF sind durch die Werkzeugträgerfunktionalität nicht betroffen, da die Werkzeugorientierungsinformation in diesen Fällen von der aktiven kinematischen Transformation bereitgestellt wird.

Zwei Drehachsen

Für zwei Drehachsen existieren allgemein zwei Lösungen. Die Auswahl dieser beiden Lösungspaare wird von der Steuerung selbst so getroffen, dass die Orientierungswinkel, die sich aus dem Frame ergeben, möglichst nahe an den vorgegebenen Winkeln liegen.

Für die Vorgabe der Winkel gibt es zwei Möglichkeiten:

1. Verweist $TC_CARR21 bzw. $TC_CARR22 auf eine Rundachse, so wird als Vorgabe die Position dieser Achse am Anfang des Satzes verwendet, in dem der Werkzeugträger aktiviert wird.
2. Verweist $TC_CARR21 bzw. $TC_CARR22 nicht auf eine Rundachse, werden die $TC_CARR13 bzw. $TC_CARR14 enthaltenen Werte verwendet.

Beispiel

Die Steuerung berechnet zunächst einen Winkel 10 Grad für eine Achse. Der vorgegebene Winkel ist 750 Grad. Zum Ausgangswinkel werden deshalb 720 Grad (= 2 * 360 Grad) addiert, so dass der endgültige Winkel 730 Grad ist.

Rundachsenoffset

Mit Hilfe der beiden systemvariablen $TC_CARR24 und $TC_CARR25 kann ein Offset der Rundachsen vorgegeben werden. Ein Wert ungleich Null in einem dieser Parameter bedeutet, dass die zugehörige Rundachse in Grundstellung nicht die Position Null, sondern die durch den Parameter angegebene Position hat. Alle Winkelangaben beziehen sich dann auf das um diesen Wert verschobene Koordinatensystem.

Beim Wechsel der Bearbeitungsebene (`G17` - `G19`) werden nur die Werkzeuglängenkomponenten des aktiven Werkzeugs vertauscht. Die des Werkzeugträgers werden nicht getauscht. Der resultierende Werkzeuglängenvektor wird dann entsprechend dem aktuellen Werkzeugträger gedreht und gegebenenfalls mit den zum Werkzeugträger gehörenden Offsets modifiziert.

Die Bestimmung der beiden Werkzeugträgerwinkel α_1 und α_2 mittels eines Frames ist unabhängig von der aktuell angewählten aktiven Ebene (`G17` - `G19`).

Grenzwertlimits

Im Systemvariablensatz ($TC_CARR30 bis $TC_CARR33) zur Beschreibung des orientierbaren Werkzeugträgers können für jede Drehachse Grenzwinkel (Softwarelimits) angegeben werden. Sind sowohl der Minimal- als auch der Maximalwert Null, werden diese Grenzen nicht ausgewertet.

Ist mindestens einer der beiden Grenzwerte ungleich Null, wird überprüft, ob die zuvor ermittelte Lösung innerhalb der zulässigen Grenzen liegt. Ist das nicht der Fall, wird zunächst versucht, durch Addition oder Subtraktion von Vielfachen von 360 Grad zu der unzulässigen Achsposition eine gültige Einstellung zu erreichen. Ist dies nicht möglich und es existieren zwei verschiedene Lösungen, wird die erste Lösung verworfen und die zweite Lösung verwendet. Die zweite Lösung wird bezüglich der Achsgrenzen wie die erste Lösung behandelt.

Wird die erste Lösung verworfen und stattdessen die zweite Lösung verwendet, werden die Inhalte von $P_TCANG[1/2] und $P_TCANG[3/4] getauscht, so dass auch in diesem Fall die tatsächlich verwendete Lösung in $P_TCANG[1/2] steht.

Die Achsgrenzen werden auch dann überwacht, wenn die Achswinkel nicht berechnet, sondern vorgegeben werden. Dies ist der Fall, wenn bei der Aktivierung eines orientierbaren Werkzeugträgers `TCOABS` aktiv ist.

2.8.8 Programmierung

Werkzeugträger-Anwahl

Ein Werkzeugträger wird mit **m** Nummer des Werkzeug-Trägers angewählt mit:

```
TCARR = m
```

Zugriff auf die Werkzeugträger Datensätze

Aus dem Teileprogramm heraus sind folgende Zugriffe möglich:

Der bisherige Wert des Parameters **n** für den Werkzeugträger **m** wird mit dem neuen "Wert" **beschrieben** mit:

```
$TC_CARRn[m] = Wert
```

Die Parameter eines Werkzeugträgers m können, sofern der Werkzeugträger-Datensatz bereits definiert wurde, **gelesen** werden mit:

`wert = $TC_CARRn[m]` (Wert muss eine `REAL`-Variable sein)

Die Werkzeugträger-Datensatz-Nr muss in dem Bereich liegen der definiert wird durch das Maschinendatum:

MD18088 $MN_MM_NUM_TOOL_CARRIER (Gesamtzahl der definierbaren Werkzeugträger-Datensätze)

Für einen NC-Kanal können diese Anzahl geteilt durch die Anzahl der aktiven Kanäle Werkzeugträger-Datensätze definiert werden.

Ausnahme:

Wenn vom Standard abweichende Einstellungen gewählt werden über das Maschinendatum:

MD28085 $MC_MM_LINK_TOA_UNIT (Zuordnung einer TO-Einheit zu einem Kanal)

Nullsetzen aller Werkzeugträger-Datensätze

Aus dem Teileprogramm heraus können alle Werte aller Werkzeugträger-Datensätze mit einem Kommando gelöscht werden.

$TC_CARR1[0] = 0

Vom Anwender nicht gesetzte Werte sind mit Wert=0 vorbelegt.

Aktivierung

Ein Werkzeugträger wird wirksam, wenn sowohl ein **Werkzeugträger** als auch ein **Werkzeug** aktiviert wurden. Die Anwahl eines Werkzeugträgers allein hat keine Wirkung. Die Wirkung der Anwahl eines Werkzeugträgers ist abhängig vom G-Befehl `TCOABS` / `TCOFR` (modale G-Gruppe Werkzeugträger).

Ein Wechsel des G-Befehls in der Gruppe `TCOABS` / `TCOFR` bewirkt eine Neuberechnung der Werkzeuglängenkomponenten, wenn ein Werkzeugträger aktiv ist. Bei `TCOABS` werden die in den Werkzeugträgerdaten abgelegten Werte für die beiden Drehwinkel α1und α2für die Ermittlung der Werkzeugorientierung verwendet.

Bei `TCOFR` werden die beiden Winkel aus dem aktuellen Frame ermittelt. Die in den Werkzeugträgerdaten abgelegten Werte bleiben jedoch unverändert erhalten. Diese werden darüber hinaus auch dazu verwendet, die Mehrdeutigkeit, die sich bei der Berechnung der Drehwinkel aus einem Frame ergeben kann, aufzulösen. Dabei werden unter mehreren möglichen Winkeln die gewählt, die von den programmierten Winkeln am wenigsten abweichen.

Hinweis

Weitere Erläuterungen zur Programmierung von Werkzeugkorrekturen mit Werkzeugträgerkinematik und zu den Systemvariablen siehe:

Weitere Informationen
Programmierhandbuch Arbeitsvorbereitung

2.8.9 Randbedingungen und Steuerungsverhalten für Orientierungen

Volle Orientierung

Mit einem gegebenen Datensatz, der eine bestimmte Kinematik beschreibt, sind nur dann alle denkbaren räumlichen Orientierungen darstellbar, wenn:

- die beiden Vektoren v_1 und v_2, die Drehachsen beschreiben, auch definiert sind (d.h. beide Vektoren müssen ungleich Null sein).
- die beiden Vektoren v_1 und v_2 aufeinander senkrecht stehen.
- die Werkzeugorientierung senkrecht auf der zweiten Drehachse steht.

Nicht definierte Orientierung

Sind diese Bedingungen nicht erfüllt und wird mit `TCOFR` eine durch einen aktiven Frame bestimmte nicht erreichbare Orientierung angefordert, wird ein Alarm ausgegeben.

Abhängigkeiten Vektor / Drehwinkel

Ist ein die Richtung einer Drehachse beschreibender Vektor v_1 oder v_2 gleich Null, muss auch der zugehörige Drehwinkel α1bzw. α2 gleich Null sein. Andernfalls wird ein Alarm ausgegeben. Der Alarm wird erst ausgegeben, wenn der Werkzeugträger aktiv wird, d.h. bei einem Werkzeugträgerwechsel.

Werkzeugfeinkorrektur kombiniert mit Orientierung

Die Kombination von Werkzeugfeinkorrektur und Werkzeugträgern ist **nicht** zulässig. Die Aktivierung der Werkzeugfeinkorrektur bei aktivem Werkzeugträger und umgekehrt die Aktivierung eines Werkzeugträgers bei aktiver Werkzeugfeinkorrektur führt zu einem Alarm.

Automatische Werkzeugträger Anwahl, RESET

Bei `RESET` oder bei Programmstart kann automatisch ein Werkzeugträger angewählt werden über das Maschinendatum:

MD20126 $MC_TOOL_CARRIER_RESET_VALUE (Wirksamer Werkzeugträger bei RESET)

Er wird analog zu der gesteuerten Anwahl eines Werkzeugs behandelt über das Maschinendatum:

MD20120 $MC_TOOL_RESET_VALUE (Werkzeuglängenkorr. Hochlauf (RESET/TP-Ende))

Das Verhalten bei `RESET` oder Programmstart wird wie bei Werkzeuganwahl gesteuert über das gleiche Bit 6 in den Maschinendaten:

MD20110 $MC_RESET_MODE_MASK (Festlegung der Steuerungsgrundstellung nach RESET/TP-Ende)

Oder:

MD20112 $MC_START_MODE_MASK (Festlegung der Steuerungsgrundstellung bei NC-START)

Für weitere Informationen siehe Funktionshandbuch "Grundfunktionen", Kapitel "BAG, Kanal, Programmbetrieb, Reset-Verhalten".

Ab SW 6.3

War bei der letzten Anwahl vor Reset `TCOABS` aktiv, ist das Verhalten gegenüber den Vorgängerständen unverändert. Ein anderer aktiver G-Codes führt dazu, dass der orientierbare Werkzeugträger mit dem Frame aktiviert wird, der zum letzten Anwahlzeitpunkt vor Reset aktiv war. Dabei werden auch geänderte Werkzeugträgerdaten ($TC_CARR...) berücksichtigt. Sind diese Daten unverändert, wird der Werkzeugträger wieder im exakt gleichen Zustand wie vor Reset aktiviert. Wurden nach der letzten Werkzeugträgeranwahl vor Reset Werkzeugträgerdaten verändert, ist eine Anwahl in Übereinstimmung mit dem alten Frame nicht immer möglich. In diesem Fall wird der orientierbare Werkzeugträger entsprechend den zu diesem Zeitpunkt gültigen Werten des G-Codes der Gruppe 42 und der aktiven Frames angewählt.

MD22530 Hilfsfunktionsausgabe an PLC

Dass bei Anwahl des Werkzeugträgers wahlweise ein konstanter oder ein M-Code, dessen Nummer von der Werkzeugträgernummer abgeleitet wird, ausgegeben wird kann eingestellt werden mit dem Maschinendatum:

MD22530 $MC_TOCARR_CHANGE_M_CODE (M-Code bei Werkzeugträgerwechsel)

Für weitere Informationen siehe Funktionshandbuch "Grundfunktionen", Kapitel "BAG, Kanal, Programmbetrieb, Reset-Verhalten".

Werkzeugträgerkinematik

Für die Werkzeugträgerkinematik sind folgende Randbedingungen zu beachten:

- Die Werkzeugorientierung in Grundstellung, beide Winkel α_1 und α_2 sind Null, ist wie im Standardfall auch bei:
  - `G17` parallel zu Z
  - `G18` parallel zu Y
  - `G19` parallel zu X
- Bezüglich der Achsgrenzwerte muss eine zulässige Stellung erreichbar sein.
- Nur wenn die beiden Rundachsen aufeinander senkrecht stehen sind auch beliebige Orientierungen einstellbar.
  Zusätzlich muss bei Maschinen, bei denen beide Achsen den Tisch drehen, die Werkzeugorientierung senkrecht auf der ersten Drehachse stehen.
  Bei Maschinen mit gemischter Kinematik muss die Werkzeugorientierung senkrecht auf der Achse stehen, die das Werkzeug dreht, d.h. ebenfalls auf der ersten Drehachse.

Für eine **in einem Frame** festgelegte Orientierung gilt:

- Die in einem Frame festgelegte Orientierung muss mit der definierten Werkzeugträgerkinematik erreichbar sein, sonst wird ein Alarm ausgegeben.
  Dieser Fall kann immer dann auftreten, wenn die beiden Drehachsen, die zur Definition der Kinematik notwendig sind, nicht aufeinander senkrecht stehen.
  Dies gilt, wenn weniger als zwei Drehachsen definiert sind und ist der Fall:
  - beim **Kinematiktyp T mit drehbarem Werkzeug**, wenn die Werkzeugachse, die die Werkzeugrichtung definiert, nicht senkrecht auf der **zweiten** steht.
  - beim **Kinematiktyp M und P mit drehbarem Werkstück**, wenn die Werkzeugachse, die die Werkzeugrichtung definiert, nicht senkrecht auf der **ersten** steht.
- Drehachsen, die zum Erreichen einer bestimmten Position einen Frame mit vorgegebener Werkzeugorientierung erfordern, sind nur im Falle einer Drehachse eindeutig bestimmt. Für zwei Drehachsen existieren allgemein zwei Lösungen.
- In allen Fällen, in denen eine Mehrdeutigkeit zu berücksichtigen ist, ist es besonders wichtig, dass in den Werkzeugdaten die ungefähr aus den Framedaten zu erwartenden Winkel abgelegt werden, bzw. dass die Drehachsen in der Nähe der zu erwartenden Positionen stehen.

Verhalten bei ASUP, REPOS

Der Werkzeugträger kann in einem asynchronen Unterprogramm ASUP gewechselt werden. Bei Fortsetzung des unterbrochenen Programms mit `REPOS` wird bei der Anfahrbewegung der neue Werkzeugträger berücksichtigt und das Programm mit diesem fortgesetzt. Auch hier ist die Behandlung analog zu der des Werkzeugwechsels in einem ASUP.

Für weitere Informationen siehe Funktionshandbuch "Grundfunktionen", Kapitel "BAG, Kanal, Programmbetrieb, Reset-Verhalten".

2.8.10 Beispiele

2.8.10.1 Beispiel: Orientierbare Werkzeugträger

Anforderung

Im nachfolgenden Beispiel wird ein Werkzeugträger verwendet, der durch eine Drehung um die Y-Achse vollständig beschrieben wird. Es reicht deshalb aus, mit dem Eintrag eines Wertes die Drehachse zu definieren (Satz `N20`).

In den Sätzen `N50` bis `N70` wird ein Schaftfräser mit der Länge 20 mm und dem Radius 5 mm beschrieben.

Im Satz `N90` wird eine Drehung von 37 Grad um die Y-Achse definiert.

Im Satz `N120` wird die Werkzeugradiuskorrektur aktiviert, und es werden alle Einstellungen vorgenommen um die in den nachfolgenden Sätzen beschriebene Korrektur unter einer Drehung von 37 Grad um die y-Achse bearbeiten zu können.

```
N10                                     ; Definition des Werkzeugtraegers 1
N20 $TC_CARR8[1] = 1                    ; Komponente der ersten Drehachse in Y-
                                        Richtung
N30
N40                                     ; Definition des Werkzeugkorrekturspei-
                                        chers T1/D1
N50 $TC_DP1[1,1] = 120                  ; Schaftfraeser
N60 $TC_DP3[1,1] = 20                   ; Laenge 1
N70 $TC_DP6[1,1] = 5                    ; Radius
N80
N90 ROT Y37                             ; 37 Grad-Drehung um y-Achse
N100
N110 X0 Y0 Z0 F10000
N120 G42 CUT2DF TCOFR TCARR = 1 T1 D1 X10
N130 X40
N140 Y40
N150 X0
N160 Y0
N170 M30
```

2.8.10.2 Beispiel Orientierbare Werkzeugträger mit drehbarem Tisch

Verwendung des MOVT-Befehls

Für die Verwendung des `MOVT`-Befehls wird angenommen, dass das Programm auf einer 5-Achs-Maschine abläuft, bei der eine Drehung der B-Achse das Werkzeug um die Y-Achse dreht:

```
N10 TRAORI()
```

```
N20 X0 X0 Z0 B45 F2000                    ; Einstellen der Werkzeugorientierung
N30 MOVT=-10                              ; Zustellbewegung 10mm in Werkzeugrich-
                                          tung
                                          ; (unter 45 Grad in der Y-Z-Ebene)
N40 MOVT=AC(20)                           ; Rueckzug in Werkzeugrichtung auf Ab-
                                          stand
                                          ; 20mm vom Nullpunkt
```

Maschine mit drehbarem Tisch

Vollständige Definition für die Verwendung eines orientierbaren Werkzeugträgers mit drehbarem Tisch:

```
N10 $TC_DP1[1,1]= 120
N20 $TC_DP3[1,1]= 13                      ; Werkzeuglänge 13mm

; Definition des Werkzeugträgers 1:

N30 $TC_CARR1[1] = 0                      ; X-Komponente des 1. Offsetvektors
N40 $TC_CARR2[1] = 0                      ; Y-Komponente des 1. Offsetvektors
N50 $TC_CARR3[1] = 0                      ; Z-Komponente des 1. Offsetvektors

N60 $TC_CARR4[1] = 0                      ; X-Komponente des 2. Offsetvektors
N70 $TC_CARR5[1] = 0                      ; Y-Komponente des 2. Offsetvektors
N80 $TC_CARR6[1] = -15                    ; Z-Komponente des 2. Offsetvektors

N90 $TC_CARR7[1] = 1                      ; X-Komponente der 1. Achse
N100 $TC_CARR8[1] = 0                     ; Y-Komponente der 1. Achse
N110 $TC_CARR9[1] = 0                     ; Z-Komponente der 1. Achse

N120 $TC_CARR10[1] = 0                    ; X-Komponente der 2. Achse
N130 $TC_CARR11[1] = 1                    ; Y-Komponente der 2. Achse
N140 $TC_CARR12[1] = 0                    ; Z-Komponente der 2. Achse

N150 $TC_CARR13[1] = 30                   ; Drehwinkel der 1. Achse
N160 $TC_CARR14[1] =-30                   ; Drehwinkel der 2. Achse

N170 $TC_CARR15[1] = 0                    ; X-Komponente des 3. Offsetvektors
N180 $TC_CARR16[1] = 0                    ; Y-Komponente des 3. Offsetvektors
N190 $TC_CARR17[1] = 0                    ; Z-Komponente des 3. Offsetvektors

N200 $TC_CARR18[1] = 0                    ; X-Komponente des 4. Offsetvektors
N210 $TC_CARR19[1] = 0                    ; Y-Komponente des 4. Offsetvektors
N220 $TC_CARR20[1] = 15                   ; Z-Komponente des 4. Offsetvektors

N230 $TC_CARR21[1] = A                    ; Bezug fuer 1. Achse
```

```
N240 $TC_CARR22[1] = B                    ; Bezug fuer 2. Achse
N250 $TC_CARR23[1] = "P"                  ; Typ des Werkzeugtraegers

N260 X0 Y0 Z0 A0 B45 F2000
N270 TCARR=1 X0 Y10 Z0 T1 TCOABS
N280 PAROT
N290 X0 Y0 Z0
N300 G18 MOVT=AC(20)
N310 G17 X10 Y0 Z0
N320 MOVT=-10
N330 PAROTOF
N340 TCOFR
N350 X10 Y10 Z-13 A0 B0
N360 ROTS X-45 Y45
N370 X20 Y0 Z0 D0
N380 Y20
N390 X0 Y0 Z20
N400 M30
```

Die Definition des orientierbaren Werkzeugträgers ist vollständig angegeben. Die Komponenten, die den Wert 0 enthalten, müssten nicht angegeben werden, da sie ohnehin mit Null vorbesetzt sind.

In `N270` wird der Werkzeugträger aktiviert.

Da in *$TC_CARR21* bzw. in *$TC_CARR22* auf die Maschinenachsen `A` und `B` verwiesen wird und `TCOABS` aktiv ist, werden die Einträge in *$TC_CARR13* und *$TC_CARR14* ignoriert, d.h. für die Drehung werden die Achsposition `A0 B45` verwendet.

Die Drehung des 4. Offsetvektors (Länge 15 mm in Z-Richtung) um die B-Achse bewirkt eine Verschiebung des Nullpunktes um X10.607 [= 15 * sin(45)] und Z-4.393 [= -15 * (1. - cos(45))]. Diese Nullpunktverschiebung wird durch einen automatisch beschriebenen Basis- bzw. Systemframe berücksichtigt, so dass die Position X10.607 Y10.000 Z8.607 angefahren wird. In Z-Richtung ergibt sich dabei durch die Werkzeuganwahl eine zusätzliche Verschiebung von 13 mm, die Y-Komponente wird durch die Tischdrehung nicht beeinflusst.

In `N280` wird eine Rotation entsprechend der Tischdrehung des orientierbaren Werkzeugträgers definiert. Die neue X-Richtung zeigt deshalb in Richtung der Winkelhalbierenden im 4. Quadranten, die neue Z-Achse in Richtung der Winkelhalbierenden im 1. Quadranten.

In `N290` wird der Nullpunkt angefahren, d.h. die Maschinenpostion X10.607 Y0 Z-4.393, da die Lage des Nullpunkts durch die Drehung nicht verändert wird.

In `N300` wird in Y auf die Position Y33.000 verfahren, da `G18` aktiv ist und die Y-Komponenten durch den aktiven Frame nicht beeinflusst werden. X- und Z-Position bleiben unverändert.

In `N310` wird die Position X17.678 Y0 Z1.536 angefahren.

In `N320` verändert sich als Folge des `MOVT`-Befehls nur die Z-Position auf den Wert -8.464. Da nur der Tisch drehbar ist, bleibt die Werkzeugorientierung unverändert parallel zur Maschinen-Z-Richtung, auch wenn die Z-Richtung des aktiven Frames um 45 Grad gedreht ist.

N330 löscht den Basis- bzw. Systemframe, die Framedefinition aus N280 wird damit wieder rückgängig gemacht.

In N340 wird mit TCOFR angegeben, dass der orientierbare Werkzeugträger entsprechend dem aktiven Frame ausgerichtet werden soll. Da wegen des PAROTOF-Befehls in N330 keine Drehung mehr aktiv ist, ergibt sich als Resultat die Grundstellung. Die Frameverschiebung wird 0.

In N350 wird deshalb die Position X10 X10 Z0 (= Z-13 + Werkzeuglänge) angefahren. Achtung: Durch die gleichzeitige Programmierung der beiden Rundachsen A und B wird die tatsächliche Lage des orientierbaren Werkzeugträgers mit der in N340 verwendeten in Übereinstimmung gebracht. Die Position, die von den drei Linearachsen angefahren wird ist davon jedoch unabhängig.

In N360 wird mit Hilfe von Raumwinkeln eine Ebene definiert deren Schnittgeraden in der X-Z- und in der Y-Z-Ebene jeweils einen Winkel von +45 Grad bzw. -45 Grad mit der X- bzw. der Y-Achse bilden. Die so definierte Ebene hat damit folgende Lage: die Flächennormale zeigt in Richtung der Raumdiagonalen.

In N370 wird im neuen Koordinatensystem auf die Position X20 Y0 Z0 verfahren. Da gleichzeitig das Werkzeug mit D0 abgewählt wird, gibt es in Z keinen zusätzlichen Offset mehr. Da die neue X-Achse in der alten X-Z-Ebene liegt, wird in diesem Satz die Maschinenposition X14.142 Y0 Z-14.142 erreicht.

In N380 verfährt im gedrehten Koordinatensystem nur die Y-Achse. Das führt zu einer Bewegung aller drei Maschinenachsen. Die Maschinenposition ist X5.977 Y16.330 Z-22.307.

In N390 wird ein Punkt auf der neuen Z-Achse angefahren. Bezogen auf die Maschinenachsen liegt er deshalb auf der Raumdiagonalen. Alle drei Achsen erreichen deshalb die Position 11.547.

2.8.10.3 Korrekturen einsatzort- und werkstückspezifisch einrechnen

Werkzeug mit Adapter

Im nachfolgenden Programmbeispiel ist ein Werkzeug mit Adapter und orientierbarem Werkzeugträger definiert. Um die Verhältnisse überschaubar zu halten, ist beim Werkzeug selbst, bei den Summen- und Einsatzkorreketuren sowie beim Adapter jeweils nur die Länge L1 von Null verschieden. Die Offsetvektoren des orientierbaren Werkzeugträgers sind alle Null.

Programmcode	Kommentar
N10 $TC_TP2[1]="MillingTool"	; Bezeichner
N20 $TC_TP7[1]=9	; Platztype
N30 $TC_TP8[1]=2	; Status : Freigeben und nicht gesperrt
; D corr. D=1	
N40 $TC_DP1[1,1]=120	; WZ-Typ - Fräsen
N50 $TC_DP3[1,1]=	; Längenkorrekturvektor
N60 $TC_DP12[1,1]=	; Verschleiß

Programmcode	Kommentar
N70 $TC_SCP13[1,1]=0.1	; Summenkoorektur DL=1
N80 $TC_ECP13[1,1]=0.01	; Einsatzkorrektur DL=1
N90 $TC_ADPTT[1]=5	; Adaptertransformation
N100 $TC_ADPT1[1]=0.001	; Adaptermaß
; Magazindaten	
N110 $TC_MAP1[1]=3	; Art des Magazins: Revolver
N120 $TC_MAP2[1]="Revolver"	; Name eines Magazins
N130 $TC_MAP3[1]=17	; Zustand des Magazins
N140 $TC_MAP6[1]=1	; Dimension - Zeile
N150 $TC_MAP7[1]=2	; Dimension - Spalte -> 2 Plätze
N160 $TC_MPP1[1,1]=1	; Platzart
N170 $TC_MPP2[1,1]=9	; Platztype
N180 $TC_MPP4[1,1]=2	; Platzzustand
N190 $TC_MPP7[1,1]=1	; Adapter auf Platz bringen
N200 $TC_MPP6[1,1]=1	; T-Nummer "MillingTool"
N210 $TC_MAP1[9999]= 7	; Art des Magazins: Zw.speicher
N220 $TC_MAP2[9999]="Zw.speicher"	; Name eines Magazins
N230 $TC_MAP3[9999]=17	; Zustand des Magazins
N240 $TC_MAP6[9999]=1	; Dimension - Zeile
N250 $TC_MAP7[9999]=1	; Dimension - Spalte -> 1 Platz
N260 $TC_MPP1[9999,1]=2	; Platzart
N270 $TC_MPP2[9999,1]=9	; Platztype
N280 $TC_MPP4[9999,1]=2	; Platzzustand
N290 $TC_MPP5[9999,1]=1	; Spindelnr. 1
N300 $TC_MDP2[1,1]=0	; Distanz Spindel zu Mag.1
; Definition des Werkzeugträgers 1	
N310 $TC_CARR10[1]=1	; Komponente 2. Drehachse in X-Richtung
N320 $TC_CARR14[1]=45	; Drehwinkel der 2. Achse
N330 $TC_CARR23[1]="T"	; Tool Mode
N340 Stopre	
N350 $SC_WEAR_TRANSFORM='B101'	
N360 T0 D0 DL=0	
N370 ROT X30	
N380 G90 G1 G17 F10000 X0 Y0 Z0	
N390 T="MillingTool" X0 Y0 Z0 TOWSTD	; X 0.000 Y11.110 Z 0.001
N400 T="MillingTool" X0 Y0 Z0 TOWMCS	; X 0.000 Y10.100 Z 1.011
N410 T="MillingTool" X0 Y0 Z0 TOWWCS	; X 0.000 Y 9.595 Z 0.876
N420 TCARR=1 X0 Y0 Z0	; X 0.000 Y 6.636 Z 8.017
N430 G18 X0 Y0 Z0	; X10.100 Y-0.504 Z 0.876
N440 M30	

Erläuterungen zum Beispiel oben

Beginnend mit Satz N390 wird in unterschiedlichen Varianten jeweils die Position X0 Y0 Z0 angefahren. Die erreichten Maschinenpostionen sind in den Sätzen jeweils im Kommentar angegeben. Im Anschluss an das Programm wird erläutert, wie die Postionen zu Stande kommen.

N390: Die Adaptertransformtion 5 (Satz N90) transformiert die Länge L1 in die Länge L2. Dieser Transformation ist lediglich das Adaptermaß selbst nicht unterworfen. Damit ergibt sich der Y-Wert (L2 bei G17) aus der Summe aus Werkzeuglänge (10), Werkzeugverschleiß (1), Summenkorrektur (0.1) und Einsatzkorrektur (0.01). Das Adaptermaß (0.001) steht in Z (L1).

N400: In Satz N350 wurden die Bits 0 und 2 gesetzt im Settingdatum:

SD42935 $SC_WEAR_TRANSFORM (Transformationen für Werkzeugkomponenten)

Dies bedeutet, dass wegen TOWMCS in Satz N400 der Werkzeugverschleiß und die Einsatzkorrektur der Adaptertransformation nicht unterworfen werden. Die Summe dieser beiden Korrekturen ist 1.01. Um dieses Maß erhöht sich deshalb die Z-Position und vermindert sich entsprechend die Y-Position gegenüber Satz N390.

In N410 ist TOWWCS aktiv. Die Summe aus Werkzeugverschleiß und Einsatzkorrektur wirkt deshalb im aktiven Werkstückkoordinatensystem. In Satz N370 wurde eine Drehung um die X-Achse um 30 Grad aktiviert. Der ursprüngliche Korrekturwert von 1.01 in Z-Richtung ergibt deshalb eine neue Z-Komponente von 0.875 (= 1.01 * cos(30)) und eine neue Y-Komponente von -0.505 (= 1.01 * sin(30)). Zu der, wie in Satz N390, gebildeten Summe aus Werkzeuglänge, Summenkorrektur und Adaptermaß addiert, ergibt sich das im Programm-Kommentar angegebene Maß.

In N420 wird zusätzlich ein orientierbarer Werkzeugträger aktiviert. Dieser führt eine Drehung um 45 Grad um die X-Achse aus (s. N310 - N330). Da alle Offsetvektoren des Werkzeugträgers Null sind, ergibt sich keine zusätzliche Nullpunktverschiebung. Der orientierbare Werkzeuträger wirkt auf die die Summe aus Werkzeuglänge, Summenkorrektur und Adaptermaß. Der resultierende Teilvektor ist X0 Y7.141 Z7.142. Darauf wird wie in Satz N410 die im WKS bewertete Summe aus Werkzeugverschleiß und Einsatzkorrektur addiert.

In N430 wird G18 aktiviert. Die Komponenten der Summe aus Werkzeuglänge, Summenkorrektur und Adaptermaß werden entsprechend getauscht. Auf diesen neuen Vektor wirkt der orientierbare Werkzeugträger unverändert (Drehung um X-Achse um 45 Grad). Dabei ergibt sich der resultierende Teilvektor X10.100 Y0.0071 Z0.0071. Der aus Werkzeugverschleiß und Einsatzkorrektur gebildete Vektor (X0 Y-0.505 Z0.875) wird durch den Wechsel der Ebene nicht beeinflusst. Die Summe beider Vektoren ergibt das im Kommentar von N430 angegebene Maß.

2.8.10.4 Beispiel Orientierbare Werkzeugträger über kinematische Kette

Beispiel für einen Werkzeugträger mit gemischter Kinematik (drehbares Werkzeug und drehbarer Tisch)

Das Beispiel zeigt einen Werkzeugträger, der über eine kinematische Kette definiert wird. Die Kette wird automatisch an den Endpunkten der beiden Einzelketten (Werkzeug und Tisch) geschlossen.

```
Programmcode                                                    Kommentar
                                                                ; Definition Werkzeugträger 1
N1070 _MIXED_1_S:
N1080 IF (DELOBJ("KIN_CHAIN_ELEM") <> 0)
N1080 IF (DELOBJ("KIN_CHAIN_ELEM") <> 0)
N1090 SETAL(62000)
N1100 ENDIF
N1110 _KIE_CNT = 0
N1120 _CARR_CNT = 1
N1130 TCARR=0                                                   ; Aktuellen Werkzeugträger deaktivieren.
N1140 $TC_CARR1[0] = 0                                          ; Alle Werkzeugträgerdaten löschen.
N1150 $TC_CARR_KIN_CNTRL[_CARR_CNT]=7                           ; Werkzeugträger über kinematische Kette
                                                                ; + Tool- und Part-Kette schließen.
N1160 $TC_CARR_KIN_TOOL_START[_CARR_CNT]="Y_AXIS"               ; Element der Tool-Kette starten.
N1170 $TC_CARR_KIN_TOOL_END[_CARR_CNT]="B_OFFSET_CORR"          ; Element der Tool-Kette beenden.
N1180 $TC_CARR_KIN_PART_START[_CARR_CNT]="X_AXIS"               ; Element der Part-Kette starten.
N1190 $TC_CARR_KIN_PART_END[_CARR_CNT]="END_PART_CHAIN"         ; Element der Part-Kette beenden.
N1200 $TC_CARR21[_CARR_CNT]=B
N1210 $TC_CARR22[_CARR_CNT]=C
N1220 $TC_CARR23[_CARR_CNT]="M"
N1230 $NK_NAME[_KIE_CNT] = "ROOT"
N1240 $NK_PARALLEL[_KIE_CNT]="X_AXIS"
N1250 $NK_TYPE[_KIE_CNT] = "OFFSET"
N1260 $NK_NEXT[_KIE_CNT] = "Y_AXIS"
N1270 _KIE_CNT=_KIE_CNT+1
                                                                ;***** Tool-Kette starten *****
N1280 $NK_NAME[_KIE_CNT] = "Y_AXIS"
N1290 $NK_TYPE[_KIE_CNT] = "AXIS_LIN"
N1300 $NK_OFF_DIR[_KIE_CNT,1] = 1
N1310 $NK_AXIS[_KIE_CNT] = "Y1"
N1320 $NK_NEXT[_KIE_CNT] = "Z_AXIS"
N1330 _KIE_CNT=_KIE_CNT+1
N1340 $NK_NAME[_KIE_CNT] = "Z_AXIS"
N1350 $NK_TYPE[_KIE_CNT] = "AXIS_LIN"
N1360 $NK_OFF_DIR[_KIE_CNT,2] = 1
N1370 $NK_AXIS[_KIE_CNT] = "Z1"
N1380 $NK_NEXT[_KIE_CNT] = "CLOSE_HEAD"
N1390 _KIE_CNT=_KIE_CNT+1
```

Programmcode	Kommentar
N1400 $NK_NAME[_KIE_CNT] = "CLOSE_HEAD"	
N1410 $NK_TYPE[_KIE_CNT] = "OFFSET"	
N1420 $NK_OFF_DIR[_KIE_CNT,0] = 0	
N1430 $NK_OFF_DIR[_KIE_CNT,1] = 0	
N1440 $NK_OFF_DIR[_KIE_CNT,2] = 0	
N1450 $NK_NEXT[_KIE_CNT] = "B_AXIS"	
N1460 _KIE_CNT=_KIE_CNT+1	
N1470 $NK_NAME[_KIE_CNT] = "B_AXIS"	
N1480 $NK_TYPE[_KIE_CNT] = "AXIS_ROT"	
N1490 $NK_OFF_DIR[_KIE_CNT,1] = 1	
N1500 $NK_OFF_DIR[_KIE_CNT,2] = 1	
N1520 $NK_A_OFF[_KIE_CNT] = -27	
N1540 $NK_AXIS[_KIE_CNT] = "B1"	
N1550 $NK_NEXT[_KIE_CNT] = "B_OFFSET"	
N1560 _KIE_CNT=_KIE_CNT+1	
N1570 $NK_NAME[_KIE_CNT] = "B_OFFSET"	
N1580 $NK_TYPE[_KIE_CNT] = "OFFSET"	
N1590 $NK_OFF_DIR[_KIE_CNT,2] = -30.6	
N1600 $NK_NEXT[_KIE_CNT] = "B_OFFSET_CORR"	
N1610 _KIE_CNT=_KIE_CNT+1	
N1620 $NK_NAME[_KIE_CNT] = "B_OFFSET_CORR"	
N1630 $NK_TYPE[_KIE_CNT] = "OFFSET"	
N1640 $NK_NEXT[_KIE_CNT] = ""	
N1650 _KIE_CNT=_KIE_CNT+1	
	;***** Tool-Kette beenden *****
	;***** Part-Kette starten *****
N1660 $NK_NAME[_KIE_CNT] = "X_AXIS"	
N1670 $NK_TYPE[_KIE_CNT] = "AXIS_LIN"	
N1680 $NK_OFF_DIR[_KIE_CNT,0] = 1	
N1690 $NK_AXIS[_KIE_CNT] = "X1"	
N1700 $NK_NEXT[_KIE_CNT] = "C_OFFSET"	
N1710 _KIE_CNT=_KIE_CNT+1	
N1720 $NK_NAME[_KIE_CNT] = "C_OFFSET"	
N1730 $NK_TYPE[_KIE_CNT] = "OFFSET"	
N1740 $NK_OFF_DIR[_KIE_CNT,0] = 300	
N1750 $NK_OFF_DIR[_KIE_CNT,1] = 150	
N1760 $NK_NEXT[_KIE_CNT] = "C_OFFSET_CORR"	
N1770 _KIE_CNT=_KIE_CNT+1N1780 $NK_NAME[_KIE_CNT] = "C_OFF-SET_CORR"	
N1790 $NK_TYPE[_KIE_CNT] = "OFFSET"	
N1800 $NK_OFF_DIR[_KIE_CNT,0] = 0	
N1810 $NK_OFF_DIR[_KIE_CNT,1] = 0	
N1820 $NK_OFF_DIR[_KIE_CNT,2] = 0	
N1830 $NK_NEXT[_KIE_CNT] = "C_AXIS"	

Programmcode	Kommentar
N1840 _KIE_CNT=_KIE_CNT+1	
N1850 $NK_NAME[_KIE_CNT] = "C_AXIS"	
N1860 $NK_TYPE[_KIE_CNT] = "AXIS_ROT"	
N1870 $NK_OFF_DIR[_KIE_CNT,2] = -1	
N1890 $NK_A_OFF[_KIE_CNT] = -58	
N1910 $NK_AXIS[_KIE_CNT] = "C1"	
N1920 $NK_NEXT[_KIE_CNT] = "CLOSE_PART"	
N1930 _KIE_CNT=_KIE_CNT+1	
N1940 $NK_NAME[_KIE_CNT] = "CLOSE_PART"	
N1950 $NK_TYPE[_KIE_CNT] = "OFFSET"	
N1960 $NK_OFF_DIR[_KIE_CNT,0] = 0	
N1970 $NK_OFF_DIR[_KIE_CNT,1] = 0	
N1980 $NK_OFF_DIR[_KIE_CNT,2] = 0	
N1990 $NK_NEXT[_KIE_CNT] = "END_PART_CHAIN"	
N2000 _KIE_CNT=_KIE_CNT+1	
N2010 $NK_NAME[_KIE_CNT] = "END_PART_CHAIN"	
N2020 $NK_TYPE[_KIE_CNT] = "OFFSET"	
	;***** Part-Kette beenden *****

2.9 Modifikation der Korrekturdaten bei drehbaren Werkzeugen

2.9.1 Einführung

Funktion

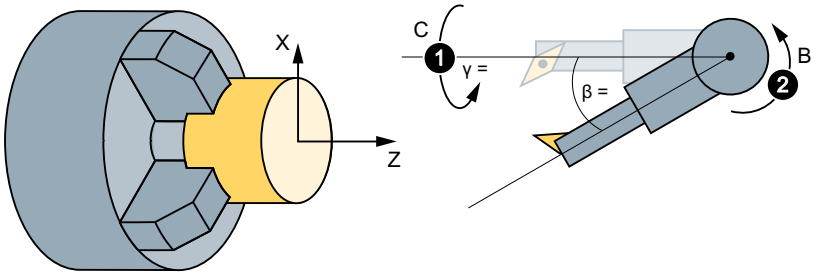
Mit der Funktion "Modifikation der Korrekturdaten bei drehbaren Werkzeugen" können die veränderten geometrischen Verhältnisse, die sich bei der Drehung von Werkzeugen (vorwiegend Drehwerkzeuge, aber auch Bohr- und Fräswerkzeuge) relativ zum bearbeiteten Werkstück ergeben, berücksichtigt werden.

Die aktuelle Drehung des Werkzeugs wird dabei immer aus einem aktuell aktiven orientierbaren Werkzeugträger (siehe Kapitel "Orientierbare Werkzeugträger (Seite 94)") bzw. aus einer mit kinematischen Ketten definierten Orientierungstransformation (siehe Funktionshandbuch Sonderfunktionen; Kapitel "Transformationsdefinition mit kinematischen Ketten") ermittelt.

Die Definition der Drehwinkel des orientierbaren Werkzeugträgers erfolgt dabei in der Regel (aber nicht notwendigerweise) mit dem Befehl TCOFR aus einem aktiven Frame. Durch diese Vorgehensweise ist es möglich, die Werkzeugorientierung unabhängig von der tatsächlichen Kinematik, mit der das Werkzeug gedreht wird, immer auf gleiche Art mit Hilfe zweier Winkel zu definieren.

Bei einer aktiven kinematischen Transformation können die erforderlichen Winkel mit der Funktion ORISOLH ermittelt werden (siehe "Orientierungen berechnen (ORISOLH) (Seite 154)").

Zur Definition der Werkzeugdrehung werden die beiden maschinenunabhängigen Ausrichtwinkel β (Beta) und γ (Gamma) verwendet. β ist dabei der Drehwinkel um die Applikate (typischerweise eine B-Achse bei `G18`) und γ eine Drehung um die Abszisse (typischerweise eine C-Achse bei G18). Zuerst wird die Drehung um γ ausgeführt, anschließend die um β. D. h., die γ-Achse wird von der β-Achse gedreht:



① Drehung um Abszisse
② Drehung um Applikate

Bild 2-40 Werkzeugdrehung

Inbetriebnahme

Die Inbetriebnahme der Funktion erfolgt mittels Maschinen- und Settingdaten.

Siehe Kapitel "Inbetriebnahme (Seite 151)".

Aktivierung

Die Modifikation der Korrekturdaten bei drehbaren Werkzeugen wird im NC-Programm über die Sprachbefehle CUTMOD (in Verbindung mit orientierbaren Werkzeugträgern) bzw. CUTMODK (für Orientierungstransformationen, die mittels kinematischer Ketten definiert wurden) aktiviert.

Siehe Kapitel "Modifikation der Korrekturdaten bei drehbaren Werkzeugen aktivieren (CUTMOD, CUTMODK) (Seite 165)".

Ergebnisse

Bei aktiver Funktion und Werkzeugdrehung werden die modifizierten Daten in Systemvariablen und BTSS-Variablen zur Verfügung gestellt.

Siehe Kapitel "Modifikation der Korrekturdaten bei drehbaren Werkzeugen aktivieren (CUTMOD, CUTMODK) (Seite 165)".

2.9.2 Drehung von Drehwerkzeugen

2.9.2.1 Schneidenlage, Schnittrichtung und Winkel bei Drehwerkzeugen

Drehwerkzeuge

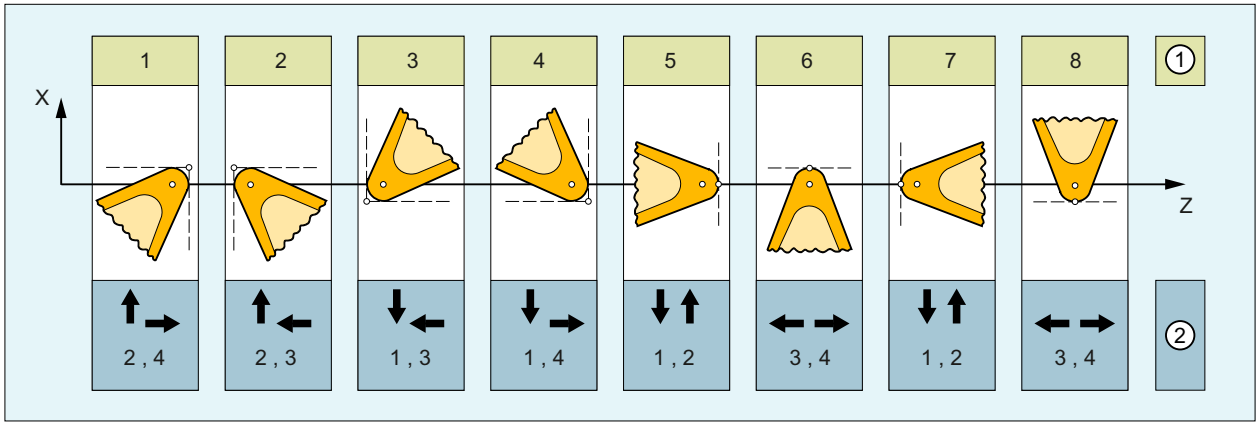
Unter Drehwerkzeugen werden im Folgenden Werkzeuge verstanden, deren Werkzeugtyp ($TC_DP1) Werte im Bereich 500 bis 599 hat. Schleifwerkzeuge (Werkzeugtypen 400 bis 499) sind den Drehwerkzeugen gleichgestellt.

Werkzeuge werden unabhängig vom Werkzeugtyp wie Drehwerkzeuge behandelt, wenn:

SD42950 $SC_TOOL_LENGTH_TYPE = 2

Schneidenlage und Schnittrichtung

Drehwerkzeuge sind durch ihre Haupt- und Nebenschneiden begrenzt. Die Schneidenlage wird durch die Lage von Haupt- und Nebenschneide relativ zu den Koordinatenachsen definiert. Jeder Schneidenlage können zwei verschiedene Schnittrichtungen zugeordnet werden.



① Schneidenlage (1 - 8)

② Der Schneidenlage zugeordnete Schnittrichtungen (1 - 4)

Bild 2-41 Schneidenlage und Schnittrichtung

Schneidenlage

Bei Schneidenlage 1 - 4 liegen Haupt- und Nebenschneide im gleichen Quadranten.

Bei Schneidenlage 5 - 8 liegen Haupt- und Nebenschneide in benachbarten Quadranten bzw. es liegt eine Koordinatenachse zwischen beiden Schneiden.

Die Schneidenlage wird im Werkzeug-Parameter $TC_DP2 abgelegt.

Schnittrichtung

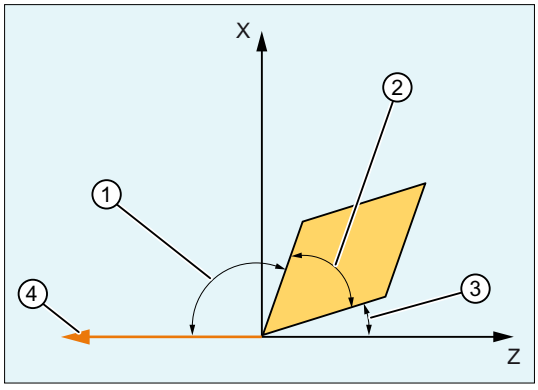
Die Schnittrichtungen 1 - 4 und bezeichnen eine positive oder negative Richtung der Koordinatenachsen:

- 1: Ordinate **-**
- 2: Ordinate **+**
- 3: Abszisse **-**
- 4: Abszisse **+**

Die Schnittrichtung wird im Werkzeug-Parameter $TC_DP11 abgelegt.

Halterwinkel und Freiwinkel

Im folgenden Bild sind für ein Drehwerkzeug mit Schneidenlage 3 Halterwinkel und Freiwinkel dargestellt. Die Arbeitsebene ist G18 (Z/X). Die Schnittrichtung ist 3 (negative Z- bzw. Abszissenrichtung).



① Halterwinkel
② Keil- oder Plattenwinkel = 180° - Halterwinkel - Freiwinkel
③ Freiwinkel
④ Schnittrichtung

Bild 2-42 Winkel und Schnittrichtung bei einem Drehwerkzeug

Schnittrichtung

Die Schnittrichtung gibt die Bezugsrichtung für den Halterwinkel an. Der Freiwinkel ist der zwischen der inversen Schnittrichtung und der benachbarten Schneide (positiv) gemessene Winkel.

Halterwinkel und Freiwinkel werden in den Werkzeug-Parametern $TC_DP10 bzw. $TC_DP24 abgelegt.

Hinweis

Schnittrichtung und Werkzeugwinkel sind nur bei Schneidenlagen 1 - 8 von Bedeutung.

2.9.2.2 Modifikationen bei der Drehung von Drehwerkzeugen

Beschreibung der Werkzeugorientierung

Drehwerkzeuge sind im Gegensatz zu Fräswerkzeugen nicht rotationssymmetrisch. Das bedeutet, dass zur Beschreibung der Werkzeugorientierung in der Regel 3 Freiheitsgrade bzw. drei rotatorische Achsen notwendig sind. Die Unabhängigkeit von der konkreten Kinematik an der Maschine ist deshalb nur insoweit gegeben, wie die gewünschte Orientierung überhaupt einstellbar ist. Gegebenenfalls kann der dritte Freiheitsgrad durch eine Drehung des Werkstückkoordinatensystems ersetzt werden.

Hinweis

Die Aufteilung der Orientierung in einen Anteil, der durch den orientierbaren Werkzeugträger erzeugt wird, und einen weiteren Anteil, der durch eine Drehung des Koordinatensystems erreicht wird, liegt in der Verantwortung der Applikation. Die Steuerung stellt dafür keine weitere Funktionalität zur Verfügung.

Wird das Werkzeug mit Hilfe einer kinematischen Transformation orientiert, steht dem Anwender zur Orientierungsberechnung die Funktion ORISOLH zur Verfügung (siehe Kapitel "Orientierungen berechnen (ORISOLH) (Seite 154)").

Winkel zwischen Schneidplatte und Bearbeitungsebene

Wird ein Drehwerkzeug gegenüber der Bearbeitungsebene (d. h. um eine Achse in der Bearbeitungsebene, typischerweise eine C-Achse) um einen Winkel gedreht, der nicht ein Vielfaches von 180° ist, wird die Projektion der (kreisförmigen) Werkzeugschneide in die Bearbeitungsebene zu einer Ellipse. Es wird davon ausgegangen, dass die durch derartige Drehungen entstehenden Abweichungen von der Kreisform so gering sind, dass sie vernachlässigt werden können (Kippwinkel < 5°). D. h., die Steuerung geht unabhängig von der Werkzeugorientierung immer von einer kreisförmigen Schneide aus.

Das bedeutet letztlich, dass die Steuerung bezüglich der aktiven Ebene als einzige von der Grundstellung abweichende Einstellung eine Drehung um 180° akzeptiert. Diese Einschränkung gilt nur für die Schneidenform. Die Werkzeuglängen werden bei beliebigen räumlichen Drehungen immer korrekt berücksichtigt.

Eine Drehung von 180° um eine Achse in der Bearbeitungsebene bedeutet, dass beim Einsatz des Werkzeugs am gleichen Einsatzort die Spindeldrehrichtung gegenüber dem Einsatz des nicht gedrehten Werkzeugs invertiert werden muss.

Mit dem Settingdatum SD42998 $SC_CUTMOD_PLANE_TOL kann ein Maximalbetrag für den Winkel vorgegeben werden, den die Schneidplatte gegenüber der Bearbeitungsebene einnehmen darf (siehe auch Kapitel "Parametrierung (Seite 151)").

Schneidenlage, Schnittrichtung und Werkzeugwinkel

Schneidenlage und Schnittrichtung werden ebenso wie der Schneidenbezugspunkt (siehe unten) nicht modifiziert, wenn das Werkzeug um ± 90° (mit einer Toleranz von etwa 1°) aus der Ebene gedreht ist, weil dann die Projektion der Schneide in die aktuelle Ebene nicht definiert ist.

Bei einer aktiven Transformation geht der aktuelle Gesamtframe immer in die Berechnung der aktuellen Werkzeugdrehung relativ zum Werkstück ein. Bei aktivem orientierbarem Werkzeugträger wird der Frame nur dann berücksichtigt, wenn im Maschinendatum MD20360 $MC_TOOL_PARAMETER_DEF_MASK (Seite 151) das Bit 21 gesetzt ist. In diesem Fall wird ein eventuell vorhandener Tischanteil des Werkzeugträgers nicht mehr getrennt berücksichtigt, weil davon auszugehen ist, dass dieser Frameanteil bereits im Gesamtframe mit enthalten ist (Systemframe PAROT).

Mit Bit 17 des Maschinendatums MD20360 $MC_TOOL_PARAMETER_DEF_MASK besteht bei aktiver Orientierungstransformation die Möglichkeit, in dem Fall, dass die Schneide nicht ohnehin in der aktuellen Bearbeitungsebene liegt, für die Berechnung der modifizierten Schneidendaten die Schneide entweder in die Bearbeitungsebene zu projizieren (Bit 17 = 0) oder sie alternativ in die Bearbeitungsebene zu drehen (Bit 17 = 1).

Hinweis

MD20360, Bit 17

Ist dieses Bit gesetzt, liegt es in der Verantwortung des Anwenders, mit einem zusätzlichen Frame oder einer Änderung der Orientierung dafür zu sorgen, dass die geometrischen Verhältnisse bei der Bearbeitung mit denen bei der Berechnung übereinstimmen.

Ist Bit 20 des Maschinendatums MD20360 $MC_TOOL_PARAMETER_DEF_MASK auf den Wert "0" gesetzt, können für den Fall, dass die Schneidenparameter für Halterwinkel und Schnittrichtung ($TC_DP10 bzw. $TC_DP24) den Wert 0 haben, für die Berechnung der modifizierten Schneidenlage und der modifizierten Schnittrichtung von null abweichende Standardwerte verwendet werden. Dabei gelten die folgenden Regeln:

- Ein Halterwinkel von 0° wirkt bei den Schneidenlagen 1 bis 4 wie ein Winkel von 112,5°.
- Ein Halterwinkel von 0° wirkt bei den Schneidenlagen 5 bis 8 wie ein Winkel von 67,.5°.
- Ein Freiwinkel von 0° wirkt bei den Schneidenlagen 1 bis 4 wie ein Winkel von 22,5°.
- Ein Freiwinkel von 0° wirkt bei den Schneidenlagen 5 bis 8 wie ein Winkel von 67,5°.

Halterwinkel und Freiwinkel werden gegebenenfalls nur temporär auf die oben genannten Standardwerte gesetzt, um die modifizierte Schneidenlage und die modifizierte Schnittrichtung berechnen zu können. D. h., Leseoperationen auf die Systemvariablen $TC_DP10 bzw. $TC_DP24 liefern weiterhin den Wert "0".

Wird ein Werkzeug in der Ebene gedreht (Drehung um eine Achse senkrecht zur Bearbeitungsebene bzw. um die Y-Achse bei G18), wird die Schneidenlage aus den resultierenden Winkeln ermittelt, die sich für Frei- und Halterwinkel ergeben. Sind diese beiden Winkel für das Werkzeug nicht angegeben (d. h. $TC_DP10 und $TC_DP24 sind beide null), wird die neue Schneidenlage alleine aus dem Drehwinkel bestimmt. Dabei gilt die Besonderheit, dass sich die Schneidenlage nur in 90°-Schritten ändert. D. h., die Schneidenlage bleibt abhängig vom Grundzustand immer entweder im Wertebereich 1 bis 4 oder im Wertebereich 5 bis 8. Die neue Schneidenlage wird auch dann ausschließlich aus dem Drehwinkel ermittelt, wenn die vorgegebenen Werte für Halterwinkel und Freiwinkel unzulässig sind (negative Werte, resultierender Plattenwinkel negativ oder größer als 90°). Unzulässige Winkel führen wahlweise zu einem Alarm (siehe MD20125 $MC_CUTMOD_ERR (Seite 151)). In allen diesen Fällen werden Freiwinkel und Halterwinkel nicht modifiziert.

Die Schnittrichtung wird abhängig von der Drehung so modifiziert, dass der resultierende Freiwinkel kleiner 90° bleibt. Passen originale Schnittrichtung und originale Schneidenlage

nicht zusammen, wird die Schnittrichtung bei Drehung des Werkzeugs nicht modifiziert. Diese Situation führt wahlweise zu einem Alarm (siehe MD20125 $MC_CUTMOD_ERR).

Der Drehwinkel in der Ebene, wie er aus dem orientierbaren Werkzeugträger bzw. aus der aktiven kinematischen Transformation ermittelt wurde, steht in den BTSS-Variablen pTCutMod bzw. ptCut- ModS sowie in den Systemvariablen $P_CUTMOD_ANG bzw. $AC_CUTMOD_ANG zur Verfügung. Dieser Winkel ist der Originalwinkel ohne eventuell anschließende Rundungen auf Vielfache von 45° oder 90°.

Grenzfälle

Haben bei einem Drehwerkzeug Schneidenlage, Schnittrichtung, Frei- und Halterwinkel gültige Werte, sodass sich durch geeignete Drehungen in der Ebene alle Schneidenlagen (1 bis 8) ergeben können, werden in den Fällen, in denen eine der beiden Schneiden (Haupt- bzw. Nebenschneide) um weniger als ein halbes Eingabe-Inkrement (0,0005° bei einer Eingabe-Feinheit von 3 Nachkommastellen) von einer Koordinatenachse entfernt ist, die Schneidenlagen 1 bis 4 gegenüber den Schneidenlagen 5 bis 8 bevorzugt.

In allen anderen Fällen (Fräswerkzeuge bzw. Drehwerkzeuge ohne gültige Schneidenparameter), in denen Drehungen nur in 90°-Schritten möglich sind, gilt: Ist der Betrag des Drehwinkels kleiner als 45° + 0,5 Eingabe-Inkremente (entspricht 45,0005° bei einer Eingabe-Feinheit von 3 Nachkommastellen), werden Schneidenlage und Schnittrichtung nicht verändert. D. h., diese Fälle werden wie Drehungen behandelt, die kleiner 45° sind. Analog dazu werden Drehungen, deren Betrag um weniger als 45° + 0,5 Eingabe-Inkremente von 180° abweicht, wie Drehungen im Bereich zwischen 135° und 225° behandelt.

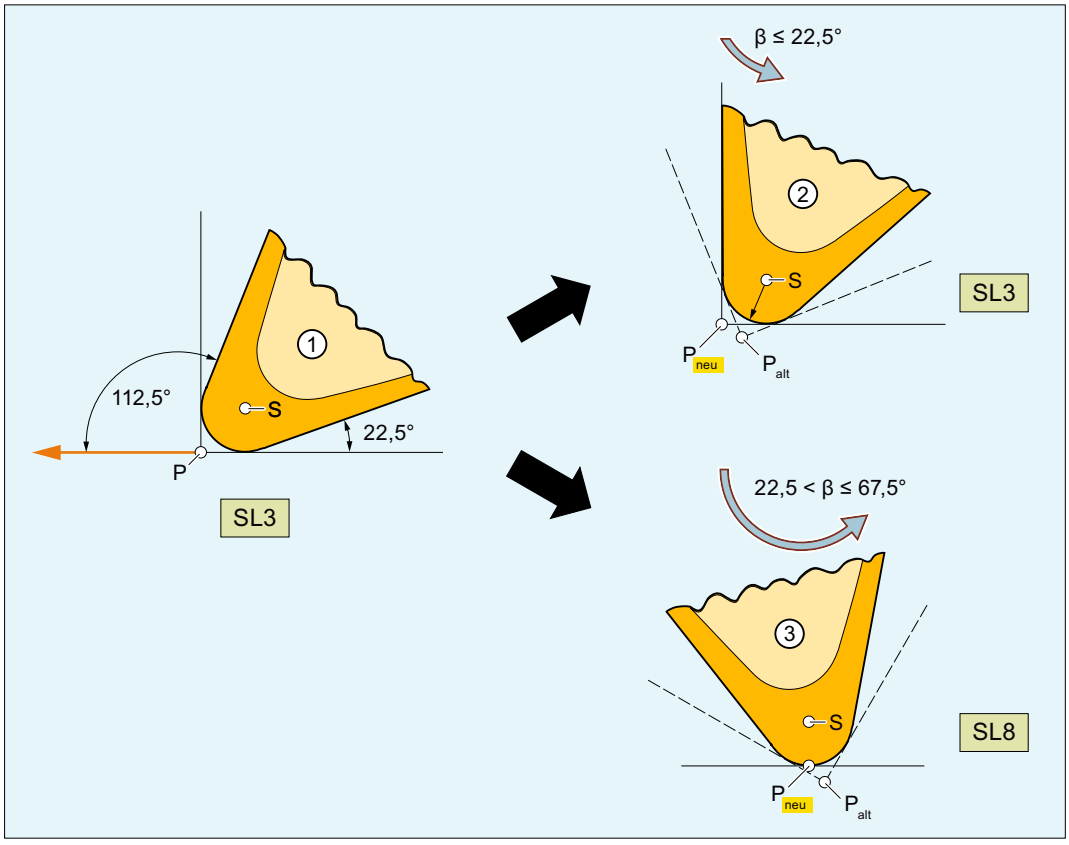
Schneidenbezugspunkt

Bei Drehwerkzeugen sind einerseits der Schneidenmittelpunkt und andererseits der Schneidenbezugspunkt definiert. Die Lage dieser beiden Punkte zueinander wird dabei durch die Schneidenlage bestimmt.

Der Abstand der beiden Punkte ist für die Schneidenlagen 1 bis 4 gleich dem √2-fachen des Schneidenradius, für die Schneidenlagen 5 bis 8 gleich dem 1-fachen Schneidenradius. Im ersten Fall liegt der Schneidenbezugspunkt relativ zum Schneidenmittelpunkt in der Bearbeitungsebene auf einer Winkelhalbierenden, im zweiten Fall auf einer Koordinatenachse.

Dreht man das Werkzeug um einen beliebigen Winkel um eine Achse senkrecht zur Bearbeitungsebene, würde sich der Schneidenbezugspunkt mitdrehen, wenn er relativ zum Werkzeug eine feste Position hätte. Die oben genannte Bedingung (Lage auf einer Achse bzw. einer Winkelhalbierenden) wäre dann nicht mehr erfüllt. Dies ist nicht erwünscht. Vielmehr soll der Schneidenbezugspunkt immer so modifiziert werden, dass der Abstandsvektor zwischen Schneidenbezugspunkt und Schneidenmittelpunkt eine der 8 genannten Richtungen hat. Die Schneidenlage muss dazu gegebenenfalls modifiziert werden.

Die Verhältnisse sind beispielhaft im folgenden Bild dargestellt:



S: Schneidenmittelpunkt

P: Schneidenbezugspunkt

SL: Schneidenlage

① Werkzeug mit der Schneidenlage 3, dem Freiwinkel 22,5° und dem Halterwinkel 112,5°

② Bei Drehungen des Werkzeugs bis 22,5° bleibt die Schneidenlage erhalten, die Lage des Schneidenbezugspunkts relativ zum Werkzeug wird jedoch so korrigiert, dass die relative Lage beider Punkte in der Bearbeitungsebene erhalten bleibt.

③ Bei größeren Werkzeugdrehungen (bis 67,5°) ändert sich die Schneidenlage auf den Wert 8.

Bild 2-43 Schneidenbezugspunkt und Schneidenlage bei Werkzeugdrehung

Hinweis

Da der Schneidenbezugspunkt durch den Werkzeuglängenvektor definiert ist, ändert sich durch die Modifikation des Schneidenbezugspunkts die effektive Werkzeuglänge.

Werkzeugrotation

Die Drehung der Werkzeugschneide in der Werkzeugaufnahme wird beschrieben durch den Werkzeugparameter $TC_DPROT. Dieser Parameter (Winkel) ist nur sinnvoll für Werkzeuge, die nicht rotationssymmetrisch sind (z. B. Drehwerkzeuge, Bohrstangen). Die Drehung erfolgt um die Richtung, die durch das Kreuzprodukt aus dem Werkzeugnormalenvektor und dem Werkzeugorientierungsvektor gebildet werden kann. Dieser Vektor kann für ein aktives Werkzeug mit der Systemvariablen $P_TOOLBIN gelesen werden. Für Standarddrehwerkzeuge zeigt dieser Vektor bei G18 in Richtung der Z-Achse. D. h., er dreht um die Achse, um die auch der Winkel γ dreht.

2.9.3 Drehung von Fräs- und Bohrwerkzeugen

2.9.3.1 Schneidenlage bei Fräs- und Bohrwerkzeugen

Fräs- und Bohrwerkzeuge

Unter Fräs- und Bohrwerkzeugen werden im Folgenden Werkzeuge verstanden, deren Werkzeugtyp ($TC_DP1) Werte im Bereich 100 bis 299 hat.

Werkzeuge werden unabhängig vom Werkzeugtyp wie Fräs- und Bohrwerkzeuge behandelt, wenn:

SD42950 $SC_TOOL_LENGTH_TYPE = 1

Schneidenlage

Für die so definierten Fräs- und Bohrwerkzeuge wird ebenfalls eine Schneidenlage eingeführt, die bei Drehungen entsprechend modifiziert wird (siehe Kapitel "Modifikationen bei der Drehung von Fräs- und Bohrwerkzeugen (Seite 151)").

Hinweis

Eine eventuell angegebene Schneidenlage bei Werkzeugen, die keine Fräs- und Bohrwerkzeuge bzw. keine Drehwerkzeuge entsprechend den genannten Definitionen sind, wird nicht ausgewertet.

Die Schneidenlage der Bohr- und Fräswerkzeuge ist wie bei Drehwerkzeugen im Werkzeugparameter $TC_DP2 hinterlegt. In Anlehnung an die Definition der Schneidenlage bei Drehwerkzeugen kann dieser Parameter die Werte 5 bis 8 annehmen. Die Schneidenlage gibt dabei die Orientierung (die Richtung der Rotationsachse) des Werkzeugs an:

Schneidenlage	Richtung der Rotationsachse des Werkzeugs
5	Abszisse **+**
6	Ordinate **+**
7	Abszisse **-**
8	Ordinate **-**

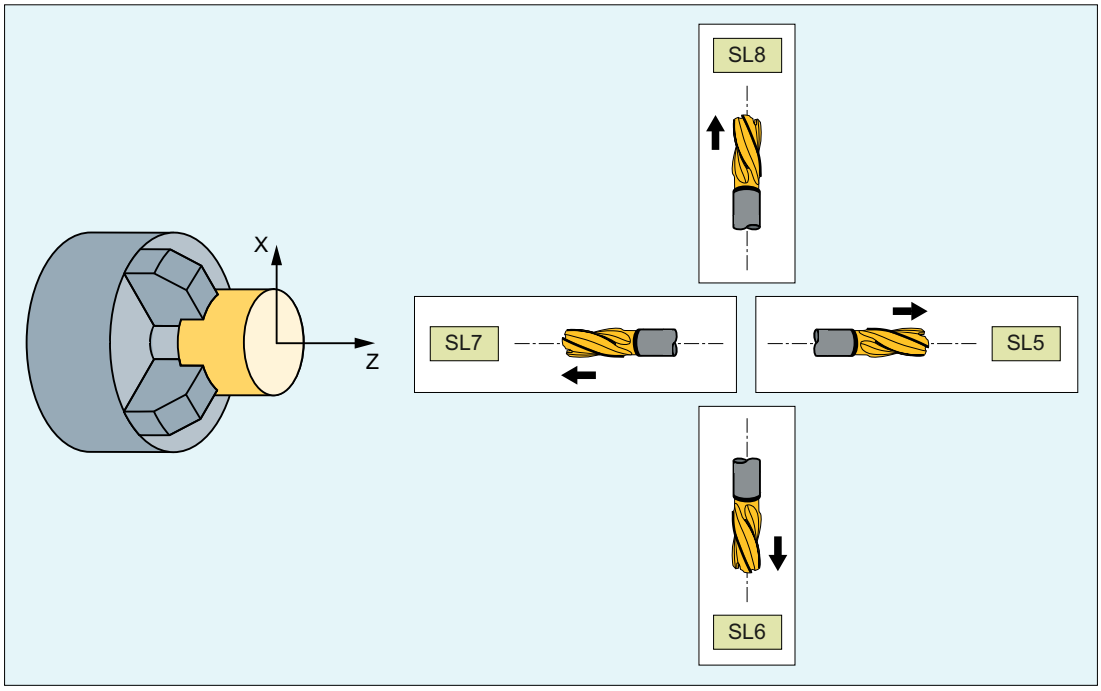


Bild 2-44 Schneidenlage (SL) 5 - 8 bei einem Fräswerkzeug

2.9.3.2 Modifikationen bei der Drehung von Fräs- und Bohrwerkzeugen

Bei einer Drehung eines Fräs- oder Bohrwerkzeugs wird die Schneidenlage entsprechend umgerechnet. Schnittrichtung und Werkzeugwinkel (Freiwinkel bzw. Halterwinkel) sind für Fräs- und Bohrwerkzeuge nicht definiert, sodass die Veränderung der Schneidenlage ausschließlich aus der Drehung abgeleitet wird. Daraus folgt weiter, dass sich bei Fräs- und Bohrwerkzeugen die Schneidenlage immer dann ändert, wenn der Betrag der Drehung gegenüber der Nullstellung größer als 45° wird.

2.9.4 Inbetriebnahme

2.9.4.1 Parametrierung

Verhalten im Fehlerfall

Bei Wirksamwerden der Funktion "Modifikation der Korrekturdaten bei drehbaren Werkzeugen" (durch expliziten Aufruf im Teileprogramm oder durch eine Werkzeuganwahl) können verschiedene Fehlerzustände auftreten. Für Fehler mit einer Fehlernummer < 100 (siehe Diagnosehandbuch) kann eingestellt werden, ob der Fehler zu einer Alarmausgabe führen soll, und falls ja, ob der Alarm nur angezeigt werden oder auch einen Programmstopp auslösen soll. Die Einstellungen erfolgen über das Maschinendatum:

MD20125 $MC_CUTMOD_ERR

Jedem Fehlerzustand sind zwei Bit des Maschinendatums zugeordnet:

Fehlerzustand	Bit	Bedeutung
Für das aktive Werkzeug ist keine gültige Schnittrichtung definiert.	0	Alarmausgabe
	1	Programmstopp
Die Schneidenwinkel (Freiwinkel und Halterwinkel) des aktiven Werkzeugs sind beide null.	2	Alarmausgabe
	3	Programmstopp
Der Freiwinkel des aktiven Werkzeugs hat einen unzulässigen Wert (< 0° oder > 180°).	4	Alarmausgabe
	5	Programmstopp
Der Halterwinkel des aktiven Werkzeugs hat einen unzulässigen Wert (< 0° oder > 90°).	6	Alarmausgabe
	7	Programmstopp
Der Plattenwinkel des aktiven Werkzeugs hat einen unzulässigen Wert (< 0° oder > 90°).	8	Alarmausgabe
	9	Programmstopp
Die Kombination Schneidenlage - Halterwinkel des aktiven Werkzeugs ist unzulässig (bei den Schneidenlagen 1 bis 4 muss der Halterwinkel ≤ 90° sein, bei den Schneidenlagen 5 bis 8 muss er ≥ 90° sein).	10	Alarmausgabe
	11	Programmstopp
Die Schneidplatte liegt nicht in der Bearbeitungsebene, und der Winkel γ zwischen Schneidplatte und Bearbeitungsebene überschreitet die mit dem Settingdatum SD42998 $SC_CUTMOD_PLANE_TOL vorgegebene Obergrenze.	12	Alarmausgabe
	13	Programmstopp
Die Schneidplatte liegt nicht in der Bearbeitungsebene. Der Betrag des Winkels α ist größer als 1°. Der Winkel α ist der Drehwinkel um die Koordinatenachse, die sowohl auf der Drehachse des Winkels β als auch auf der Drehachse des Winkels γ senkrecht steht (bei G18 die X-Achse).	14	Alarmausgabe
	15	Programmstopp

Hinweis

Das jeweils 2. Bit (Programmstopp) wird nur wirksam, wenn auch das zugehörige 1. Bit (Alarmausgabe) gesetzt ist.

Hinweis

Fehler mit einer Fehlernummer ≥ 100 haben immer eine Alarmausgabe und einen Programmstopp zur Folge.

Initialisierungswert für CUTMOD

Die mit dem Sprachbefehl CUTMOD (Seite 154) programmierbare Funktion wird nach einem Warmstart automatisch mit folgendem Wert initialisiert:

MD20127 $MC_CUTMOD_INIT = <Initialisierungswert für CUTMOD>

Standardwert: 0

Bei MD20127 = -2 wird als Initialisierungswert der Wert aus folgendem Maschinendatum wirksam:

MD20126 $MC_TOOL_CARRIER_RESET_VALUE (Wirksamer Werkzeugträger bei Reset)

Funktionsspezifische Werkzeugeinstellungen

Das folgende Maschinendatum ist eine Bitleiste, mit der verschiedene Funktionen mit Bezug zu Werkzeugen gesteuert werden können:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK

Für die Funktion "Modifikation der Korrekturdaten bei drehbaren Werkzeugen" sind folgende Bits relevant:

Bit	Bedeutung	
17	Mit Bit 17 wird eingestellt, ob bei der Modifikation der Korrekturdaten für Dreh- und Schleifwerkzeuge die Schneidenebene zur Berechnung der Korrekturwerte in die Bearbeitungsebene projiziert oder gedreht werden soll.	
	= 0	Die Schneidenebene wird in die Bearbeitungsebene projiziert.
	= 1	Die Schneidenebene wird in die Bearbeitungsebene gedreht.
	Hinweis: Bit 17 wird nur ausgewertet, wenn die Funktion in Verbindung mit einer kinematischen Orientierungstransformation aktiv ist.	
18	Mit Bit 18 wird für Dreh- und Schleifwerkzeuge die Bestimmung der Ebene bei der Modifikation der Korrekturdaten festgelegt.	
	= 0	Bestimmung der Ebene durch das Settingdatum SD42940 $SC_TOOL_LENGTH_CONST hat Vorrang vor der Ebenenbestimmung durch die G-Befehle der G-Gruppe 6 (Ebenenanwahl, G17 - G19).
	= 1	Es wird immer die aktive Ebene verwendet (G17 - G19).
20	Für den Fall, dass die Werkzeugparameter $TC_DP10 (Halterwinkel) und/oder $TC_DP24 (Freiwinkel) den Wert "0" haben, können zur Berechnung der modifizierten Schneidenlage und der modifizierten Schnittrichtung von "0" abweichende Standardwerte verwendet werden.	
	= 0	Es werden folgende Standardwerte verwendet: • Halterwinkel 112,5° bei Schneidenlagen 1 - 4 • Halterwinkel 67,5° bei Schneidenlagen 5 - 8 • Freiwinkel 22,5° bei Schneidenlagen 1 - 4 • Freiwinkel 67,5° bei Schneidenlagen 5 – 8
	= 1	Es wird ein Alarm ausgegeben.
	Hinweis: Bit 20 dient zur Herstellung der Kompatibilität mit älteren Softwareständen.	
21	Mit Bit 21 wird eingestellt, ob bei aktivem orientierbarem Werkzeugträger der aktive Gesamtframe bei der Modifikation der Korrekturdaten berücksichtigt werden soll.	
	= 0	Eine eventuell vorhandene Drehung im Tischanteil des Werkzeugträgers wird berücksichtigt. Frames werden ignoriert.
	= 1	Anstelle des Tischanteils des Werkzeugträgers wird der aktive Gesamtframe berücksichtigt. Der Gesamtframe kann dabei auch einen Tischanteil des Werkzeugträgers enthalten.

Abweichung Schneidplatten-/Bearbeitungsebene bei CUTMOD bzw. CUTMODK

Der maximal zulässige Winkel, um den die Schneidplatte beim Aufruf der Funktion CUTMOD bzw. CUTMODK (Seite 165) aus der Bearbeitungsebene (im Standardfall G18) gedreht sein darf (≙ maximal zulässige Abweichung des Winkels γ von einer der beiden Standardpositionen 0° bzw. 180°), wird festgelegt im Settingdatum:

SD42998 $SC_CUTMOD_PLANE_TOL

Beispiel:

SD42998 = 5.0 ⇒ Die Schneidplatte darf um maximal 5° aus der Bearbeitungsebene gedreht sein.

Hinweis

SD42998 = 0

Ist SD42998 $SC_CUTMOD_PLANE_TOL auf "0" gesetzt, sind Abweichungen bis 89° zulässig!

Abweichung Schneidplatten-/Bearbeitungsebene bei ORISOLH

Der maximal zulässige Winkel, um den die Schneidplatte beim Aufruf der Funktion ORISOLH (Seite 165) aus der Bearbeitungsebene gedreht sein darf, wird festgelegt im Settingdatum:

SD42999 $SC_ORISOLH_INCLINE_TOL

2.9.5 Programmierung

2.9.5.1 Orientierungen berechnen (ORISOLH)

Die vordefinierte Funktion ORISOLH unterstützt den Anwender dabei, die Rundachspositionen einer Maschine so einzustellen, dass ein Drehwerkzeug in eine definierte, kinematikunabhängige Lage relativ zum Werkstück gebracht wird. Voraussetzung ist, dass eine 6-Achstransformation aktiv ist, die mit kinematischen Ketten parametriert ist.

Es stehen zwei Grundfunktionen zur Verfügung:

- Werkzeug ausrichten
  Die beiden Winkel β und γ werden vorgegeben. Die Funktion berechnet die dazu nötigen Winkel der drei Orientierungsachsen.
- Werkzeug ausrichten direkt
  Die Winkel der zweiten und dritten Orientierungsachse werden vorgegeben. Die Funktion berechnet daraus die zugehörigen Winkel β und γ, sowie den Winkel der noch fehlenden ersten Orientierungsachse.

Hinweis

Reihenfolge der Orientierungsachsen

Durchläuft man die kinematische Kette, die den Aufbau der Maschine beschreibt, vom Werkstück bis zum Werkzeug, dann gelten für die Reihenfolge der drei Orientierungsachsen einer 6-Achstransformation folgende Festlegungen:

- Die Orientierungsachse, die am nächsten am Werk**stück** liegt, ist die **erste** Orientierungsachse.
- Die Orientierungsachse, die am nächsten am Werk**zeug** liegt, ist die **dritte** Orientierungsachse.

Die erste Orientierungsachse wird in der Regel eine Spindel sein, und die entsprechende Drehung wird deshalb in diesen Fällen durch einen drehenden Frame realisiert.

Syntax

```
<RetVal> = ORISOLH(<Cntrl>,<W1>,<W2>)
```

Bedeutung

ORISOLH:	Funktionsaufruf		
<RetVal>:	Rückgabewert der Funktion		
	Datentyp:	INT	
	Wertebereich:	0, -2, -3, ..., -17	
	Werte:	0	Funktion wurde ohne Fehler beendet.
		-2	Es ist keine gültige Transformation (6-Achs-Orientierungstransformation) aktiv.
		-3	Der erste Parameter (<Cntrl>) ist negativ.
		-4	Die 1er-Stelle des ersten Parameters (<Cntrl>) ist ungültig. Es sind nur die Werte 0 und 1 zulässig.
		-5	Die 10er-Stelle des ersten Parameters (<Cntrl>) ist ungültig. Es sind nur die Werte 0 bis 3 zulässig.
		-6	Die 100er-Stelle des ersten Parameters (<Cntrl>) ist ungültig. Es sind nur die Werte 0 und 1 zulässig.
		-7	Die 1000er-Stelle des ersten Parameters (<Cntrl>) ist ungültig. Es sind nur die Werte 0 bis 3 zulässig.
		-8	Bei der Funktion "Werkzeug ausrichten direkt" ist der Winkel γ zu groß.
		-9	Bei der Funktion "Werkzeug ausrichten direkt" verletzt mindestens eine der vorgegebenen Achspositionen eine Achsgrenze.
		-10	Es ist kein Werkzeug aktiv.
		-11	Die verlangte Orientierung ist nicht einstellbar.
		-12	Die Anpassung der freien Achswinkel bei Hirthverzahnung ist für die erste oder einzige Lösung nicht möglich.
		-13	Die Anpassung der freien Achswinkel bei Hirthverzahnung ist für die zweite Lösung nicht möglich.
		-14	Die Anpassung der freien Achswinkel bei Hirthverzahnung ist für keine der beiden Lösungen möglich.
		-15	Die erste Orientierungsachse ist als Hirthachse parametriert.
		-16	Sowohl die zweite als auch die dritte Rundachse ist als Hirthachse parametriert. Es darf nur maximal eine der beiden Achsen Hirthachse sein.
		-17	Bei der Funktion "Schwenken direkt" ist mindestens eine der vorgegebenen Achspositionen nicht mit der zugehörigen Hirthverzahnung kompatibel.

<Cntrl>:	Steuert das Verhalten der Funktion		
	Datentyp:	INT	
	Der Parameter <Cntrl> ist dezimalcodiert (1er- bis 1000er-Stelle):		
	1er-Stelle:	Die 1er-Stelle steuert das Verhalten bei Fehlern.	
		xxx**0**	Im Fehlerfall (Rückgabewert < 0) wird die Programmbearbeitung abgebrochen und der Alarm 14106 ausgegeben. **Hinweis:** Der Alarm wird unabhängig vom Wert der 1er-Stelle auch dann ausgegeben, wenn der Parameter <Cntrl> negativ ist.
		xxx**1**	Im Fehlerfall (Rückgabewert < 0) wird **kein** Alarm ausgegeben. Der Anwender hat im Programm die Möglichkeit, selbst geeignet zu reagieren.
	10er-Stelle:	Steuert das Verhalten bei Vorhandensein einer Orientierungsachse mit Hirthverzahnung. **Hinweis:** Dieser Parameter wird nur bei der Funktion "Werkzeug ausrichten" ausgewertet (d. h., wenn die 100er-Stelle den Wert "0" hat).	
		xx**0**x	Die Achsposition wird auf die nächstgelegene Position gerundet.
		xx**1**x	Die Achspositionen werden so gerundet, dass die Abweichung des Winkels β von seinem programmierten Wert minimal wird.
		xx**2**x	Die Achspositionen werden so gerundet, dass der Winkel β gleich dem größten möglichen Wert wird, der kleiner als der programmierte Wert ist (β wird abgerundet).
		xx**3**x	Die Achspositionen werden so gerundet, dass der Winkel β gleich dem kleinsten möglichen Wert wird, der größer als der programmierte Wert ist (β wird aufgerundet).
	100er-Stelle:	Gibt an, welche Funktion ausgeführt werden soll, bzw. welche Bedeutung die beiden nachfolgenden Parameter <W1> und <W2> haben.	
		x**0**xx	Funktion "Werkzeug ausrichten" Parameter <W1> und <W2> haben folgende Bedeutung: • <W1> = β • <W2> = γ Es werden die zugehörigen Winkel der Orientierungsachsen berechnet.
		x**1**xx	Funktion "Werkzeug ausrichten direkt" <W1> ist die Positionsvorgabe für die zweite Orientierungsachse, <W2> die Positionsvorgabe für die dritte Orientierungsachse einer 6-Achstransformation. Es werden die Position der ersten Orientierungsachse und die Winkel β und γ bestimmt, die mit den beiden Positionsvorgaben kompatibel sind. Falls kein Fehler auftritt, werden in der Systemvariablen $P_ORI_POS[<n>, <m>] immer zwei Lösungen ausgegeben. Dabei verweist der erste Index <n> (0 oder 1) auf die Lösung und der zweite Index <m> (0 ... 2) auf die Orientierungsachse:

			• $P_ORI_POS[0/1, **0**]: Position der ersten Orientierungsachse • $P_ORI_POS[0/1, **1**]: Winkel β • $P_ORI_POS[0/1, **2**]: Winkel γ Es wird überprüft, ob die Positionsvorgaben <W1> und <W2> mit eventuell aktiven Hirthverzahnungen oder aktiven Softwarelimits kompatibel sind. Ist das nicht der Fall, wird eine entsprechende Fehlernummer zurückgegeben (siehe Parameter <RetVal>). Bei einer beliebigen Wahl der beiden Winkel <W1> und <W2> liegt die Schneide des Werkzeugs im Allgemeinen nicht in der Bearbeitungsebene. Der Winkel γ, um den die Schneide aus der Bearbeitungsebene gedreht ist, darf nicht größer sein als der Grenzwert, der durch das Settingdatum SD42999 $SC_ORISOLH_INCLINE_TOL bestimmt wird.
	1000er-Stelle:	Gibt an, wie die Positionen der Lösungen gegebenenfalls modifiziert werden, wenn die 100er-Stelle den Wert "0" hat, d. h. bei der Funktion "Werkzeug ausrichten".	
		0xxx	Die berechneten Achspositionen sollen möglichst nahe bei den aktuellen Maschinenachspositionen liegen.
		1xxx	Die berechneten Achspositionen sollen bei Moduloachsen möglichst nahe bei der Mitte des Modulobereichs liegen, bei anderen Achsen möglichst nahe bei 0. Für Nicht-Moduloachsen bedeutet das, dass die Achspositionen auf den Bereich -180° … +180° reduziert werden.
		2xxx	Die berechneten Achspositionen sollen unabhängig vom Achstyp auf den Bereich -180° … +180° reduziert werden.
<W1>:	Erster Winkel Die Bedeutung ergibt sich aus der 100er-Stelle des Parameters <Cntrl>.		
	Datentyp:	REAL	
<W1>:	Zweiter Winkel Die Bedeutung ergibt sich aus der 100er-Stelle des Parameters <Cntrl>.		
	Datentyp:	REAL	

Hinweis

Nicht programmierte Parameter haben den Standardwert "0".

Weitere Informationen

Die Anzahl gefundener Lösungen zusammen mit weiteren Statusinformationen beim Ausführen der Funktion ORISOLH können über folgende Systemvariablen gelesen werden:

Systemvariable	Bedeutung		
$P_ORI_POS [<n>, <m>]	Liefert die Winkel der Orientierungsachsen, die sich bei Orientierungsprogrammierung ergeben.		
	<n>:	Index der Lösung	
		Wertebereich:	0, 1
	<m>:	Index der Orientierungsachse	
		Wertebereich:	0 ... 2
			Die Reihenfolge der Orientierungsachsen (1 ... 3) bezieht sich auf die Definition der Achsen in $NT_ROT_AX_NAME.
	Beim Aufruf der Funktion ORISOLH im Modus "Werkzeug ausrichten direkt" enthalten die Variablen $P_ORI_POS[0/1, 1] und P_ORI_POS[0/1, 2] die zu den beiden Lösungen gehörenden Werte der beiden Winkel β und γ. Die erste in $P_ORI_POS[<n>, <m>] eingetragene Lösung, d. h. mit dem Index <n> = 0, ist immer die, die von der Steuerung gewählt wird, wenn man die verlangte Orientierung direkt anfährt. Der zweite Index <m> bezieht sich dabei auf die Orientierungsachse, d. h. auf $NT_ROT_AX_NAME. Die in $P_ORI_POS[<n>, <m>] eingetragenen Achspositionen berücksichtigen die Offsets, die in $NK_OFF und $NK_OFF_FINE eingetragen sind, d. h., diese Achswinkel können in nachfolgenden Sätzen ohne weitere Modifikation dazu verwendet werden, die gewünschte Orientierung einzustellen. Ist eine Rundachse eine Hirthachse, werden die Lösungspositionen auf die nächstgelegene Raststellung der Hirthverzahnung gerundet. In der Systemvariablen $P_ORI_DIFF können für Hirth-verzahnte Rundachsen die Differenzen der Achspositionen für die exakten Lösungen und die an die Hirth-Rasterung angepassten Lösungen gelesen werden.		
$P_ORI_DIFF [<n>, <m>]	Liefert die Differenz zwischen den exakten und den in $P_ORI_POS zur Verfügung gestellten Positionen der Orientierungsachsen, die sich bei Orientierungsprogrammierung ergeben.		
	<n>:	Index der Lösung	
		Wertebereich:	0, 1
	<m>:	Index der Orientierungsachse	
		Wertebereich:	0 ... 2
			Die Reihenfolge der Orientierungsachsen (1 ... 3) bezieht sich auf die Definition der Achsen in $NT_ROT_AX_NAME.
	Der Inhalt kann nur dann ungleich Null sein, wenn die Positionen gerastert werden (Hirth-Verzahnung), d. h. wenn das Systemdatum $NT_HIRTH_INCR der betreffenden Achse ungleich Null ist und wenn diese Achse eine manuelle Rundachse ist.		

Systemvariable	Bedeutung		
$P_ORI_SOL	Werden bei einer Orientierungstransformation mit mehr als einer Orientierungsachse die Achswinkel berechnet, die zu einer vorgegebenen Orientierung führen sollen, gibt es im Allgemeinen mehr als eine Lösung. In der Systemvariablen $P_ORI_SOL ist die Anzahl der gültigen Lösungen zusammen mit zusätzlichen Statusinformationen enthalten. Der Inhalt von $P_ORI_SOL ist folgendermaßen codiert:		
	Werte < 0	Allgemeine Fehlerzustände	
		-1	Für die aktive Transformation wurden noch keine Lösungen berechnet (fehlender Aufruf von ORISOLH).
		-2	Es ist keine Transformation aktiv, oder die aktive Transformation ist keine Orientierungstransformation (6-Achs-Transformation), die Positionen zu einer vorgegebenen Orientierungsprogrammierung liefern kann.
		-4	Die gewünschte Orientierung ist mit der gegebenen Kinematik nicht einstellbar.
		-5	Beim Aufruf der Funktion ORISOLH im Modus "Werkzeug ausrichten direkt" wurde keine Lösung gefunden.
		-6	Beim Aufruf der Funktion ORISOLH im Modus "Werkzeug ausrichten direkt" ist der Winkel γ zu groß.
		-7	Beim Aufruf der Funktion ORISOLH im Modus "Werkzeug ausrichten direkt" wurde ein Winkel vorgegeben, der aufgrund der Hirthverzahnung nicht einstellbar ist.
		-8	Die erste Orientierungsachse (Frameachse) darf nicht als Hirthachse parametriert sein.
		-9	Sowohl die zweite als auch die dritte Rundachse ist als Hirthachse parametriert. Es darf nur max. eine der beiden Achsen Hirthachse sein.
		-10	Es wurde keine Anpassung der Lösung(en) an die Hirthachsverzahnung gefunden.
	Werte > 0 1er-Stelle	Anzahl der mathematisch möglichen Lösungen ohne Berücksichtigung von Achsgrenzen und evtl. Fehlerbedingungen.	
		0	Es existiert keine Lösung, d. h., die verlangte Orientierung ist nicht einstellbar. Dieser Fall kann drei verschiedene Ursachen haben: • Die verlangte Orientierung ist aufgrund der Maschinenkinematik (nicht rechtwinkelig angeordnete Orientierungsachsen) auch bei beliebigem Verfahrbereich der Orientierungsachsen prinzipiell nicht erreichbar. In diesem Fall sind die 10er- und 100er-Stelle von $P_ORI_SOL beide null, die den Orientierungsachsen zugeordneten Statusvariablen $P_ORI_STAT enthalten den Wert "-4". • Die berechneten Lösungen sind nicht erreichbar, weil diese die Achsbegrenzungen verletzen würden. Die Positionen der Orientierungsachsen, die sich ohne Achsbegrenzungen ergeben würden, können in $P_ORI_POS gelesen werden. • Beim Aufruf der Funktion ORISOLH im Modus "Werkzeug ausrichten direkt" wurden Achspositionen derart

Systemvariable	Bedeutung		
			vorgegeben, dass entweder der Orientierungsvektor oder der Orientierungsnormalenvektor des Werkzeugs parallel zur ersten Orientierungsachse, deren Position berechnet werden soll, ausgerichtet ist. Die Position dieser Achse ist in diesen Fällen nicht definiert.
		1	Es existiert eine Lösung. Dieser Fall kann drei verschiedene Ursachen haben: • Es gibt aufgrund der vorgegebenen Orientierung und der Maschinenkinematik auch ohne Berücksichtigung der Achsgrenzen nur eine Lösung (mathematisch gesehen handelt es sich um zwei zusammenfallende Lösungen). Dieser Fall tritt bei nicht rechtwinkeligen Kinematiken am Rand des Orientierungsbereichs auf. $P_ORI_POS enthält beide (identischen) Lösungen. • Es gibt nur eine Lösung, weil eine zweite Lösung aufgrund verletzter Achsgrenzen ungültig ist. Die gültige Lösung ist dann immer die erste Lösung in $P_ORI_POS. Die zweite Lösung, die sich ohne Berücksichtigung der Achsgrenzen ergeben würde, ist in $P_ORI_POS ebenfalls lesbar. • Beim Aufruf der Funktion ORISOLH im Modus "Werkzeug ausrichten direkt" ist das der Normalfall. Zu den vorgegebenen Achspositionen zweier Orientierungsachsen gibt es im Allgemeinen nur eine gültige Position der zu berechnenden fehlenden Orientierungsachse.
		2	Es existieren zwei Lösungen.
		8	Es existieren unendlich viele Lösungen d. h. die Position einer Orientierungsachse (der Polachse) ist beliebig. Von den beiden möglichen Stellungen der anderen Achse(n) ist jedoch eine aufgrund verletzter Achsgrenzen ausgeschlossen.
		9	Es existieren unendlich viele Lösungen, d. h. die Position einer Orientierungsachse (der Polachse) ist unbestimmt. Die unbestimmte Achse kann aus der 100er-Stelle oder aus der Systemvariablen $P_ORI_STAT ermittelt werden.
	Werte > 0 10er-Stelle	Bitcodierte Anzeige für verletzte Achsgrenzen. Die genaue Fehlerursache kann aus der Systemvariablen $P_ORI_STAT ermittelt werden.	
		Bit 0 (Wert 10):	Für mindestens eine Lösung ist mindestens eine Achsgrenze der 1. Orientierungsachse verletzt.
		Bit 1 (Wert 20):	Für mindestens eine Lösung ist mindestens eine Achsgrenze der 2. Orientierungsachse verletzt.
		Bit 2 (Wert 40):	Für mindestens eine Lösung ist mindestens eine Achsgrenze der 3. Orientierungsachse verletzt.
	Werte > 0 100er-Stelle	Bitcodierte Anzeige für nicht-definierte Achspositionen (kann nur auftreten, wenn es unendlich viele Lösungen gibt, d. h., wenn die 1er-Stelle gleich "9" ist).	
		Bit 0 (Wert 100):	Die Position der 1. Orientierungsachse ist nicht definiert.

Systemvariable	Bedeutung		
		Bit 1 (Wert 200):	Die Position der 2. Orientierungsachse ist nicht definiert.
		Bit 2 (Wert 400):	Die Position der 3. Orientierungsachse ist nicht definiert.
	Die Bezeichnungen 1., 2. und 3. Orientierungsachse beziehen sich auf die Definition der Achsen in $NT_ROT_AX_NAME.		

Systemvariable	Bedeutung		
$P_ORI_STAT [<n>]	Liefert für jede der maximal 3 Orientierungsachsen den Status nach Aufruf von ORISOLH.		
	<n>:	Index der Orientierungsachse (entspricht dem Index der betreffenden Orientierungsachse in $NT_ROT_AX_NAME)	
		Wertebereich:	0 ... 2
			Die Reihenfolge der Orientierungsachsen (1 ... 3) bezieht sich auf die Definition der Achsen in $NT_ROT_AX_NAME.
	Der Inhalt von $P_ORI_STAT ist folgendermaßen codiert:		
	Werte < 0	Allgemeine Fehlerzustände	
		-1	Der Status ist nicht definiert (fehlender Aufruf von ORISOLH).
		-2	Es ist keine Transformation aktiv, oder die aktive Transformation ist keine Orientierungstransformation (6-Achs-Transformation), die Positionen zu einer vorgegebenen Orientierungsprogrammierung liefern kann.
		-3	Die Achse ist in der aktiven Transformation nicht enthalten.
		-4	Die Position der Achse kann nicht berechnet werden, weil mit der gegebenen Kinematik die verlangte Orientierung auch bei beliebig angenommenem Verfahrbereich der Achse nicht erreichbar ist.
		-5	Beim Aufruf der Funktion ORISOLH im Modus "Werkzeug ausrichten direkt" wurden Achspositionen derart vorgegeben, dass entweder der Orientierungsvektor oder der Orientierungsnormalenvektor des Werkzeugs parallel zur ersten Orientierungsachse, deren Position berechnet werden soll, ausgerichtet ist. Die Position dieser Achse ist in diesen Fällen nicht definiert.
		-6	Beim Aufruf der Funktion ORISOLH im Modus "Werkzeug ausrichten direkt" ist der Winkel γ zu groß.
		-7	Beim Aufruf der Funktion ORISOLH im Modus "Werkzeug ausrichten direkt" wurde ein Winkel vorgegeben, der aufgrund der Hirthverzahnung nicht einstellbar ist.
		-8	Die erste Orientierungsachse (Frameachse) darf nicht als Hirthachse parametriert sein.
		-9	Sowohl die zweite als auch die dritte Rundachse ist als Hirthachse parametriert. Es darf nur max. eine der beiden Achsen Hirthachse sein.
		-10	Es wurde keine Anpassung der Lösung(en) an die Hirthachsverzahnung gefunden.
	Werte > 0	Bitcodierte Anzeige für verletzte Achsgrenzen der ersten Lösung.	
	1er-Stelle	Bit 0 (Wert 1):	Die erste Lösung verletzt die untere Achsgrenze.
		Bit 1 (Wert 2):	Die erste Lösung verletzt die obere Achsgrenze.
	Werte > 0	Bitcodierte Anzeige für verletzte Achsgrenzen der zweiten Lösung.	
	10er-Stelle	Bit 0 (Wert 10):	Die zweite Lösung verletzt die untere Achsgrenze.
		Bit 1 (Wert 20):	Die zweite Lösung verletzt die obere Achsgrenze.

Systemvariable	Bedeutung		
	Werte > 0	Anzeige einer nicht-definierten Achsposition.	
	100er-Stelle	Bit 0 (Wert 100):	Die Position der Orientierungsachse ist nicht definiert, d. h., die verlangte Orientierung wird mit jeder beliebigen Einstellung der Rundachse erreicht (Polstellung). Diese Information ist auch in der Systemvariablen $P_ORI_SOL enthalten.
	Von den Fehlernummern, die eine Verletzung der Achsgrenzen anzeigen, können mehrere gleichzeitig auftreten. Da bei der Verletzung einer Achsgrenze versucht wird, durch Addition bzw. Subtraktion von Vielfachen von 360° eine Position innerhalb der erlaubten Achsgrenzen zu erreichen, ist – falls dies nicht möglich ist – nicht eindeutig definiert, ob die untere oder die obere Achsgrenze verletzt wurde. Gibt es für die verlangte Orientierung keine Lösung ($P_ORI_SOL = 0), ist der Status der in der Transformation enthaltenen Orientierungsachsen "0".		

Hinweis

$NT_ROT_AX_NAME

Mit dieser Systemvariablen wird auf maximal 3 Achsen verwiesen, die zur Einstellung der Orientierung dienen. Sie enthält die Namen der Kettenglieder ($NK_NAME), die diejenigen Maschinenachsen (Rundachsen) definieren, die die Orientierungsbewegungen ausführen müssen, die sich aus einer kinematischen Transformation ergeben. Die Reihenfolge, in der die maximal drei Rundachsen in dieser Systemvariablen enthalten sind, ist für die Maschinenkinematik bedeutungslos, da diese aus der Struktur der kinematischen Ketten abgeleitet wird. Da sie aber die Reihenfolge definiert, mit der andere Variablen auf die Rundachsen zugreifen, muss die Reihenfolge der Orientierungsachsen in $NT_ROT_AX_NAME mit der Kinematikbeschreibung übereinstimmen.

Hinweis

Statusinformationen

Die Statusinformationen, die z. B. anzeigen, dass eine Orientierung nicht erreichbar ist oder nur unter Verletzung der relevanten Achsgrenzen erreichbar wäre, führen nicht zu einem NC-Alarm. Es liegt in der Verantwortung des Anwenders, auf die genannten Bedingungen geeignet zu reagieren.

2.9.5.2 Modifikation der Korrekturdaten bei drehbaren Werkzeugen aktivieren (CUTMOD, CUTMODK)

Die Modifikation der Korrekturdaten bei drehbaren Werkzeugen wird im NC-Programm über den Sprachbefehl CUTMOD (in Verbindung mit orientierbaren Werkzeugträgern) bzw. CUTMODK (für Orientierungstransformationen, die mittels kinematischer Ketten definiert wurden) aktiviert.

Hinweis

Da orientierbare Werkzugträger und Orientierungstransformationen, die mittels kinematischer Ketten definiert wurden, nicht gleichzeitig aktiv sein können, kommt es nicht zu Konflikten zwischen beiden Varianten.

Syntax

```
CUTMOD = <Value>
```

bzw.

```
CUTMODK = <Command>
```

Bedeutung

CUTMOD:	Funktionsaufruf in Verbindung mit orientierbaren Werkzeugträgern		
<Value>:	Zugewiesener Wert		
	Datentyp:	INT	
	Wert:	0	Die Funktion ist deaktiviert. Die von den Systemvariablen $P_AD... gelieferten Werte sind gleich den korrespondierenden Werkzeugparametern.
		> 0	Die Funktion wird aktiviert, falls ein orientierbarer Werkzeugträger mit der angegebenen Nummer aktiv ist, d. h., die Aktivierung ist an einen bestimmten orientierbaren Werkzeugträger gebunden. Die von den Systemvariablen $P_AD... gelieferten Werte sind gegenüber den korrespondierenden Werkzeugparametern abhängig von der aktiven Drehung gegebenenfalls modifiziert. Die Deaktivierung des bezeichneten orientierbaren Werkzeugträgers deaktiviert die Funktion temporär, die Aktivierung eines anderen orientierbaren Werkzeugträgers deaktiviert sie permanent. Im ersten Fall wird die Funktion deshalb bei erneuter Anwahl des gleichen orientierbaren Werkzeugträgers wieder aktiviert, im zweiten Fall ist eine erneute Anwahl notwendig, auch dann, wenn zu einem späteren Zeitpunkt der orientierbare Werkzeugträger mit der angegebenen Nummer erneut aktiviert wird. Die Funktion wird durch Reset nicht beeinflusst.
		-1	Die Funktion wird immer aktiviert, falls ein orientierbarer Werkzeugträger aktiv ist. Beim Wechsel des Werkzeugträgers oder bei dessen Abwahl und einer späteren erneuten Anwahl muss CUTMOD nicht erneut gesetzt werden.
		-2	Die Funktion wird immer aktiviert, falls ein orientierbarer Werkzeugträger aktiv ist, dessen Nummer gleich der des aktuell aktiven orientierbaren Werkzeugträgers ist. Ist kein orientierbarer Werkzeugträger aktiv, ist das gleichbedeutend mit `CUTMOD=0`. Ist ein orientierbarer Werkzeugträger aktiv, ist das gleichbedeutend mit der unmittelbaren Angabe der aktuellen Werkzeugträgernummer.
		< -2	Werte kleiner 2 werden ignoriert, d. h., dieser Fall wird so behandelt, als wäre CUTMOD nicht programmiert. **Hinweis:** Dieser Wertebereich sollte nicht verwendet werden, weil er für eventuelle spätere Erweiterungen reserviert ist.
CUTMODK:	Funktionsaufruf in Verbindung mit Orientierungstransformationen, die mittels kinematischer Ketten definiert wurden		

<Command>:	Zugewiesenes Kommando		
	Datentyp:	STRING	
	Wert:	"NEW"	Es werden die für die "Modifikation der Korrekturdaten" relevanten Zustände einer aktiven mit kinematischen Ketten definierten Transformation, der Name der Transformation und der aktuelle Konturframe abgespeichert. **Hinweis:** Dieses Kommando ist nur zulässig, wenn eine geeignete Transformation (TRAORI_DYN, TRAORI_STAT oder TRAANG_K) aktiv ist.
		"OFF"	Schaltet die aktive "Modifikation der Korrekturdaten" aus. Die zuvor mit "NEW“ abgelegten Daten bleiben erhalten. **Hinweis:** Dieses Kommando ist auch zulässig, wenn CUTMODK nicht aktiv ist. Es bleibt dann ohne Auswirkung. Ein eventuell vorhandener Datensatz zur "Modifikation der Korrekturdaten" bleibt erhalten.
		"ON"	Mit diesem Kommando wird die "Modifikation der Korrekturdaten" mit einem zuvor mit dem Kommando "NEW" gespeicherten Datensatz erneut aktiviert. Ist bei Ausführung dieses Kommandos eine Transformation mit dem Namen des gespeicherten Datensatzes aktiv, wird die "Modifikation der Korrekturdaten" sofort wirksam. Andernfalls wird die Aktivierung verzögert, bis eine aktive Transformation aktiviert wird.
		"CLEAR"	Schaltet wie das Kommando "OFF" die "Modifikation der Korrekturdaten" aus und löscht zusätzlich den abgespeicherten Datensatz. **Hinweis:** Dieses Kommando ist auch zulässig, wenn CUTMODK nicht aktiv ist.

Hinweis

SD42984 $SC_CUTDIRMOD

Die über den Befehl CUTMOD bzw. CUTMODK aktivierbare Funktion ersetzt die über das Settingdatum SD42984 $SC_CUTDIRMOD aktivierbare Funktion. Diese Funktion steht jedoch weiterhin unverändert zur Verfügung. Da es aber nicht sinnvoll ist, beide Funktionen parallel zu nutzen, kann sie nur aktiviert werden, wenn CUTMOD gleich Null und CUTMODK gleich dem Nullstring ist.

Weitere Informationen

Modifizierte Korrekturdaten lesen

Die modifizierten Korrekturdaten werden in den folgenden Systemvariablen und BTSS-Variablen zur Verfügung gestellt:

Bedeutung	Systemvariable	BTSS-Variable
Schneidenlage	$P_AD[2]	cuttEdgeParam2
Halterwinkel	$P_AD[10]	cuttEdgeParam10
Schnittrichtung	$P_AD[11]	cuttEdgeParam11
Freiwinkel	$P_AD[24]	cuttEdgeParam24

Die Daten sind gegenüber den korrespondierenden Werkzeugparametern ($TC_DP2[..., ...], ...) immer dann modifiziert, wenn die Funktion "Modifikation der Korrekturdaten bei drehbaren Werkzeugen" mit dem Befehl `CUTMOD` bzw. `CUTMODK` aktiviert wurde und das Werkzeug durch einen orientierbaren Werkzeugträger bzw. eine geeignete Orientierungstransformation gedreht wurde.

Weitere funktionsrelevante Systemvariablen

Systemvariable	Bedeutung
$P_CUTMOD_ANG / $AC_CUTMOD_ANG	Liefert den Winkel, um den ein Werkzeug in der aktiven Bearbeitungsebene gedreht wurde und der der Ermittlung modifizierter Schneidendaten bei den Funktionen CUTMOD bzw. CUTMODK zugrunde liegt.
$P_CUTMOD / $AC_CUTMOD	Liest den aktuell gültigen Wert, der zuletzt mit dem Befehl CUTMOD programmiert wurde (Nummer des Werkzeugträgers, für den die Modifikation der Korrekturdaten aktiviert werden soll). War der letzte programmierte Wert `CUTMOD=-2` (Aktivierung mit dem aktuell aktiven orientierbaren Werkzeugträger), wird in der Systemvariablen nicht der Wert "-2", sondern die Nummer des zum Zeitpunkt der Programmierung aktiven orientierbaren Werkzeugträgers zurückgeliefert.
$P_CUTMODK / $AC_CUTMODK	Liest den Namen der Transformation, unter dem der aktuell gültige Datensatz zur "Modifikation der Korrekturdaten" angelegt wurde.

Systemvariable	Bedeutung
$P_CUT_INV / $AC_CUT_INV	Liefert den Wert TRUE, wenn das Werkzeug so gedreht ist, dass die Spindeldrehrichtung invertiert werden muss. Dazu müssen in dem Satz, auf den sich die jeweilige Leseoperation bezieht, die folgenden vier Bedingungen erfüllt sein: 1. Es ist ein Dreh- oder Schleifwerkzeug aktiv (Werkzeugtypen 400 bis 599 und / oder SD42950 $SC_TOOL_LENGTH_TYPE = 2). 2. Die Modifikation der Korrekturdaten wurde mit dem Befehl `CUTMOD` bzw. `CUTMODK` aktiviert. 3. Es ist ein orientierbarer Werkzeugträger bzw. eine mit kinematischen Ketten definierte Orientierungstransformation aktiv, der/die durch den Befehl CUTMOD bzw. CUTMODK selektiert wurde. 4. Das Werkzeug wird durch den orientierbaren Werkzeugträger bzw. durch die kinematische Orientierungstransformation so gedreht, dass die resultierende Normale der Werkzeugschneide gegenüber der Ausgangslage um mehr als 90° (typischerweise 180°) gedreht ist. Ist mindestens eine der genannten vier Bedingungen nicht erfüllt, liefert die Variable den Wert FALSE. Für Werkzeuge, deren Schneidenlage nicht definiert ist, ist der Wert der Variablen immer FALSE.

Systemvariable	Bedeutung	
$P_CUTMOD_ERR	Fehlerzustand nach dem letzten Aufruf der CUTMOD-Funktion Die CUTMOD-Funktion kann auch implizit bei Werkzeugwechsel aufgerufen werden. Die Variable wird bei Reset auf null zurückgesetzt. Sie wird bei jedem Werkzeugwechsel zunächst zurückgesetzt und gegebenenfalls neu beschrieben. Die Variable ist Bit-codiert. Die Bits haben folgenden Bedeutungen:	
	Bit 0:	Für das aktive Werkzeug ist keine gültige Schnittrichtung definiert.
	Bit 1:	Die Schneidenwinkel (Freiwinkel und Halterwinkel) des aktiven Werkzeugs sind beide null.
	Bit 2:	Der Freiwinkel des aktiven Werkzeugs hat einen unzulässigen Wert (< 0° oder > 180°).
	Bit 3:	Der Halterwinkel des aktiven Werkzeugs hat einen unzulässigen Wert (< 0° oder > 90°).
	Bit 4:	Der Plattenwinkel des aktiven Werkzeugs hat einen unzulässigen Wert (< 0° oder > 90°).
	Bit 5:	Die Kombination Schneidenlage - Halterwinkel des aktiven Werkzeugs ist unzulässig (bei den Schneidenlagen 1 bis 4 muss der Halterwinkel ≤ 90° sein, bei den Schneidenlagen 5 bis 8 muss er ≥ 90° sein).
	Bit 6:	Unzulässige Drehung des aktiven Werkzeugs. Das Werkzeug wurde um ± 90° (mit einer Toleranz von etwa 1°) aus der aktiven Bearbeitungsebene herausgedreht. Dadurch ist die Schneidenlage in der Bearbeitungsebene nicht mehr definiert.
	Bit 7:	Die Schneidplatte liegt nicht in der Bearbeitungsebene, und der Winkel zwischen Schneidplatte und Bearbeitungsebene überschreitet die mit dem Settingdatum SD42998 $SC_CUTMOD_PLANE_TOL vorgegebene Obergrenze.
	Bit 8:	Die Schneidplatte liegt nicht in der Bearbeitungsebene. Der Betrag des Winkels α ist größer als 1°. Der Winkel α ist der Drehwinkel um die Koordinatenachse, die sowohl auf der Drehachse des Winkels β als auch auf der Drehachse des Winkels γ senkrecht steht (bei G18 die X-Achse).

$P_...: Vorlaufvariablen

$AC_...: Hauptlaufvariablen

Alle Hauptlaufvariablen können in Synchronaktionen gelesen werden. Ein Lesezugriff aus dem Vorlauf generiert einen Vorlaufstopp.

Ebenenwechsel

Für die Bestimmung von modifizierter Schneidenlage, Schnittrichtung und Halter- bzw. Freiwinkel ist die Betrachtung der Schneide in der jeweils aktiven Ebene (G17 - G19) maßgebend.

Enthält jedoch das Settingdatum SD42940 $SC_TOOL_LENGTH_CONST (Wechsel der Werkzeuglängenkomponenten bei Ebenenwechsel) einen gültigen Wert ungleich Null (plus oder minus 17, 18 oder 19), so bestimmt dessen Inhalt die Ebene, in der die relevanten Größen betrachtet werden.

Diese Vorrangregel der Settingdaten über den G-Code kann durch Setzten von Bit 18 des Maschinendatums $MC_TOOL_PARAMETER_DEF_MASK außer Kraft gesetzt werden. D. h., wenn dieses Bit gesetzt ist, gilt immer die mit dem G-Befehl der Gruppe 6 definierte Ebene.

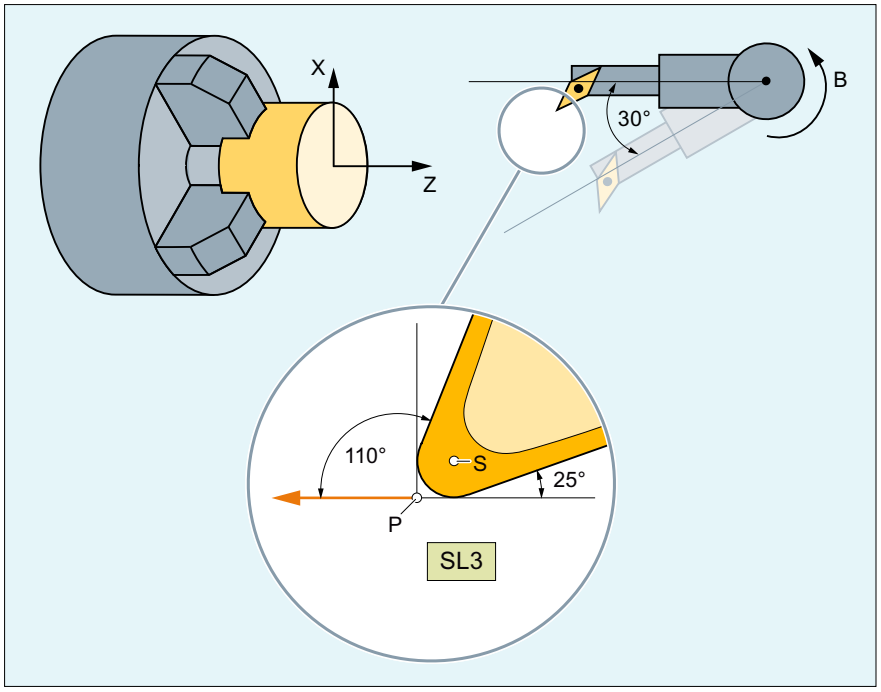
Wirksamkeit der modifizierten Schneidendaten

Die modifizierte Schneidenlage und der modifizierte Schneidenbezugspunkt werden bei Programmierung auch für ein bereits aktives Werkzeug sofort wirksam. Eine Werkzeugneuanwahl ist dazu nicht notwendig.

2.9.6 Beispiel

Beispiel

Bei einem Werkzeug mit der Schneidenlage 3 und einem orientierbaren Werkzeugträger, der das Werkzeug um die B-Achse drehen kann, soll mit Hilfe des CUTMOD-Befehls die Schneidenlage nach einer Werkzeugdrehung modifiziert werden.



S: Schneidenmittelpunkt
P: Schneidenbezugspunkt
SL: Schneidenlage

Programmcode	Kommentar
N10 $TC_DP1[1,1]=500	
N20 $TC_DP2[1,1]=3	; Schneidenlage
N30 $TC_DP3[1,1]=12	
N40 $TC_DP4[1,1]=1	
N50 $TC_DP6[1,1]=6	
N60 $TC_DP10[1,1]=110	; Halterwinkel
N70 $TC_DP11[1,1]=3	; Schnittrichtung

```
Programmcode                                          Kommentar
N80 $TC_DP24[1,1]=25                                  ; Freiwinkel

N90 $TC_CARR7[2]=0 $TC_CARR8[2]=1 $TC_CARR9[2]=0      ; B-Achse
N100 $TC_CARR10[2]=0 $TC_CARR11[2]=0                  ; C-Achse
$TC_CARR12[2]=1
N110 $TC_CARR13[2]=0
N120 $TC_CARR14[2]=0
N130 $TC_CARR21[2]=X
N140 $TC_CARR22[2]=X
N150 $TC_CARR23[2]="M"

N160 TCOABS CUTMOD=0
N170 G18 T1 D1 TCARR=2                                ; X        Y         Z
N180 X0 Y0 Z0 F10000                                  ; 12.000   0.000     1.000

N190 $TC_CARR13[2]=30
N200 TCARR=2
N210 X0 Y0 Z0                                         ; 10.892   0.000    -5.134
N220 G42 Z-10                                         ;  8.696   0.000   -17.330
N230 Z-20                                             ;  8.696   0.000   -21.330
N240 X10                                              ; 12.696   0.000   -21.330
N250 G40 X20 Z0                                       ; 30.892   0.000    -5.134

N260 CUTMOD=2 X0 Y0 Z0                                ;  8.696   0.000    -7.330
N270 G42 Z-10                                         ;  8.696   0.000   -17.330
N280 Z-20                                             ;  8.696   0.000   -21.330
N290 X10                                              ; 12.696   0.000   -21.330
N300 G40 X20 Z0                                       ; 28.696   0.000    -7.330

N310 M30
```

Die Zahlenwerte in den Kommentaren geben jeweils die Satzendpositionen in Maschinenkoordinaten (MKS) in der Reihenfolge X → Y → Z an.

Erläuterungen

In Satz `N180` wird zunächst das Werkzeug bei `CUTMOD=0` und nicht gedrehtem orientierbaren Werkzeugträger angewählt. Da alle Offsetvektoren des orientierbaren Werkzeugträgers 0 sind, wird die Position angefahren, die den in `$TC_DP3[1,1]` und `$TC_DP4[1,1]` angegebenen Werkzeuglängen entspricht.

In Satz `N200` wird der orientierbare Werkzeugträger mit einer Drehung von 30° um die B-Achse aktiviert. Da die Schneidenlage wegen `CUTMOD=0` nicht modifiziert wird, ist nach wie vor der alte Schneidenbezugspunkt maßgebend. Deshalb wird in Satz `N210` die Position angefahren, die den alten Schneidenbezugspunkt im Nullpunkt beibehält (d. h. der Vektor (1, 12) wird in der Z/X-Ebene um 30° gedreht).

In Satz N260 ist im Unterschied zu Satz N200 CUTMOD=2 wirksam. Aufgrund der Drehung des orientierbaren Werkzeugträgers wird die modifizierte Schneidenlage 8. Daraus folgen auch abweichende Achspositionen.

In den Sätzen N220 bzw. N270 wird jeweils die Werkzeugradiuskorrektur (WRK) aktiviert. Die unterschiedliche Schneidenlage in beiden Programmstücken hat auf die Endpositionen der Sätze, in denen die WRK aktiv ist, keinen Einfluss, die entsprechenden Positionen sind deshalb identisch. Erst in den Abwahlsätzen N260 bzw. N300 wirken sich die unterschiedlichen Schneidenlagen wieder aus.

2.10 Inkrementell programmierte Korrekturwerte

2.10.1 G91 Erweiterung

Voraussetzungen

Kettenmaßprogrammierung mit G91 ist so definiert, dass bei Anwahl einer Werkzeugkorrektur der Korrekturwert additiv zum inkrementell programmierten Wert gefahren wird.

Anwendungen

Für Anwendungen wie Ankratzen ist es erforderlich, im Kettenmaß nur den programmierten Weg zu fahren. Die aktivierte Werkzeugkorrektur wird nicht gefahren.

Ablauf

Anwahl einer Werkzeugkorrektur mit Kettenmaßprogrammierung

- Mit Werkzeugspitze an Werkstück ankratzen.
- Die aktuelle Istposition reduziert um Werkzeugkorrektur in Basisframe übernehmen (Istwertsetzen).
- Ab der Nullposition inkrementell fahren.

Aktivierung

Ob bei FRAME und inkrementeller Programmierung einer Achse eine geänderte Werkzeuglänge herausgefahren wird oder nur der programmierte Weg gefahren wird ist einstellbar mit dem Settingdatum:

SD42442 $SC_TOOL_OFFSET_INCR_PROG (Werkzeuglängenkorrekturen)

Nullpunktverschiebung / Frames G91

Ob beim Wert = 1 standardmäßig Nullpunktverschiebung bei FRAME und inkrementeller Programmierung einer Achse herausgefahren wird oder beim Wert = 0 nur der programmierte Wert gefahren wird ist einstellbar mit dem Settingdatum:

SD42440 $SC_FRAME_OFFSET_INCR_PROG (Nullpunktverschiebung in Frames)

Für weitere Informationen siehe Funktionshandbuch "Grundfunktionen", Kapitel "Achsen, Koordinatensysteme, Frames".

Randbedingung

Ist das Verhalten so eingestellt, dass die Verschiebung über Programmende und `RESET` hinweg aktiv bleibt
MD20110 $MC_RESET_MODE_MASK, Bit6=1 (Festlegung der Steuerungs-Grundstellung nach Reset/TP-Ende)
und wird im 1. Teileprogrammsatz ein inkrementeller Weg programmiert, so wird die Korrektur additiv zum programmierten Weg immer herausgefahren.

Hinweis

Bei dieser Konfiguration sollten Teileprogramme immer mit Absolut-Programmierung beginnen.

2.10.2 Verfahren in Richtung der Werkzeugorientierung (MOVT)

Typische Anwendung)

An Maschinen mit orientierbarem Werkzeugträger kann in Werkzeugrichtung verfahren werden (typischerweise beim Bohren), ohne dass ein Frame aktiviert wird (z.B. mittels `TOFRAME` oder `TOROT`), bei dem eine der Achsen in Richtung des Werkzeugs zeigt.

Ebenso an Maschinen, bei denen bei einer Schrägbearbeitung ein Frame aktiv ist, der die schräge Ebene definiert, das Werkzeug aber nicht exakt senkrecht eingestellt werden kann, weil wegen eines indexierten Werkzeugträgers (Hirth-Verzahnung) die Werkzeugorientierung nicht beliebig einstellbar ist.

In diesen Fällen muss dann - abweichend von der eigentlich verlangten Bewegung senkrecht zur Ebene - in Werkzeugrichtung gebohrt werden, da andernfalls der Bohrer nicht in Richtung seiner Längsachse geführt würde, was im Allgemeinen den Bruch des Bohrers zur Folge haben würde.

MOVT

Der Endpunkt einer solchen Bewegung wird mit `MOVT=...` programmiert. Der programmierte Wert wirkt standardmäßig inkrementell in Werkzeugrichtung. Die positive Richtung ist dabei von der Werkzeugspitze zur Werkzeugaufnahme definiert. Der Inhalt von `MOVT` ist bei der Zustellbewegung (beim Bohren) deshalb in der Regel negativ, bei der Rückzugsbewegung dagegen positiv. Das entspricht den Verhältnissen bei normaler achsparalleler Bearbeitung z.B. mit `G91Z...`

Wird die Bewegung in der Form `MOVT=AC(...)` programmiert, wirkt `MOVT` absolut. Dazu wird eine Ebene definiert, die durch den aktuellen Nullpunkt verläuft, und deren Flächennormalenvektor parallel zur Werkzeugorientierung ist. `MOVT` gibt dann die Lage bezüglich dieser Ebene an:

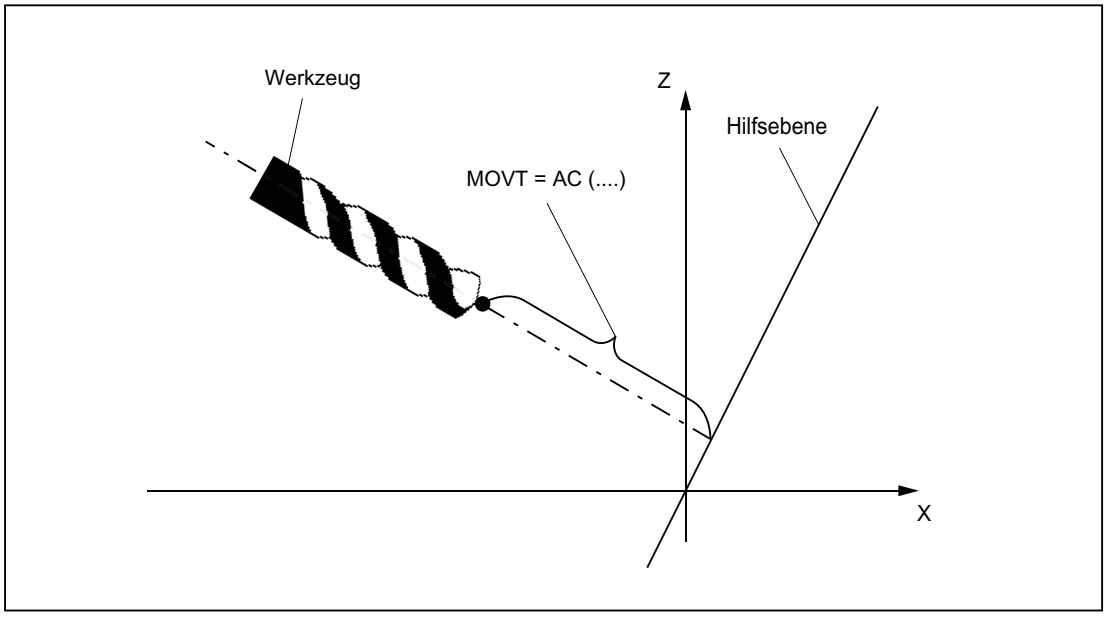


Bild 2-45 Definition der Position bei Absolutprogrammierung einer Bewegung in Werkzeugrichtung

Der Bezug zu dieser Hilfsebene dient nur zur Berechnung der Endposition. Aktive Frames werden durch diese interne Berechnung nicht beeinflusst.

Anstelle von `MOVT=...` kann auch `MOVT=IC(...)` geschrieben werden, wenn deutlich sichtbar zum Ausdruck gebracht werden soll, dass `MOVT` inkrementell wirkt. Ein funktioneller Unterschied zwischen beiden Schreibweisen besteht nicht.

Randbedingungen

Für die Programmierung mit `MOVT` gibt es folgende Randbedingungen:

- Sie ist unabhängig vom Vorhandensein eines orientierbaren Werkzeugträgers. Die Richtung der Bewegung ist von der aktiven Ebene abhängig. Sie verläuft in Richtung der Applikate, d.h. bei `G17` in Z-Richtung, bei `G18` in Y-Richtung und bei `G19` in X-Richtung. Dies gilt sowohl für den Fall, dass keine orientierbarer Werkzeugträger aktiv ist als auch für den Fall eines orientierbaren Werkzeugträgers ohne drehbares Werkzeug oder mit drehbarem Werkzeug in Grundstellung.
- `MOVT` wirkt bei aktiver Orientierungstransformation (3-4-5-Achs-Transformation) in gleicher Weise.
- Wird in einem Satz mit `MOVT` gleichzeitig die Werkzeugorientierung verändert (z.B. bei aktiver 5-Achstransformation durch gleichzeitige Interpolation der Rundachsen), ist die Orientierung am Satzanfang für die Bewegungsrichtung von `MOVT` maßgebend. Die Bahn der Werkzeugspitze (TCP-Tool Center Point) wird durch die Orientierungsänderung nicht beeinflusst.
- Linear- oder Spline-Interpolation (`G0`, `G1`, `ASPLINE`, `BSPLINE`, `CSPLINE`) muss aktiv sein. Andernfalls wird ein Alarm ausgegeben. Ist eine Spline-Interpolationsart aktiv, ist die resultierende Bahn im Allgemeinen keine Gerade, da der von `MOVT` ermittelte Endpunkt so behandelt wird, als sei er explizit mit X, Y, Z programmiert worden.
- In einem Satz mit `MOVT` dürfen keine Geometrieachsen programmiert sein (Alarm 14157).

2.11 Zuordnung der Werkzeuglängenkomponenten zu den Geometrieachsen

2.11.1 Zuordnung gemäß Werkzeugtyp und Arbeitsebene

Die Werte der Werkzeugparameter Länge 1 ... 3 sind in den Systemvariablen $TC_DP3 ... $TC_DP5 abgelegt (siehe Kapitel "Werkzeugparameter (Seite 37)"). Die Zuordnung zu den Geometrieachsen und damit die Wirkrichtung der Werkzeuglängenkomponenten ist abhängig vom Werkzeugtyp ($TC_DP1) und der aktiven Bearbeitungsebene (G17/G18/G19).

Tabelle 2-1 Dreh-/Schleifwerkzeuge ($TC_DP1 = 400 … 599)

Arbeitsebene	Länge 1	Länge 2	Länge 3
G17 (X/Y)	Y	X	Z
G18 (Z/X)	X	Z	Y
G19 (Y/Z)	Z	Y	X

Tabelle 2-2 Fräswerkzeuge / Spezialwerkzeuge ($TC_DP1 <> 400 … 599)

Arbeitsebene	Länge 1	Länge 2	Länge 3
G17 (X/Y)	Z	Y	X
G18 (Z/X)	Y	X	Z
G19 (Y/Z)	X	Z	Y

2.11.2 Zuordnung bei Ebenenwechsel

Die Zuordnung der Werkzeuglängenkomponenten (Länge, Verschleiß und Basismaß) zu den Geometrieachsen wird bei einem Wechsel der Bearbeitungsebene **nicht** verändert, wenn das folgende Settingdatum ungleich null gesetzt ist:

SD42940 $SC_TOOL_LENGTH_CONST <> 0

Die Zuordnung der Werkzeuglängenkomponenten zu den Geometrieachsen ergibt sich dann aus der Einer- und Zehnerstelle des Settingdatums gemäß den folgenden Tabellen.

Tabelle 2-3 Dreh- / Schleifwerkzeuge ($TC_DP1 = 400 … 599)

SD42940	Zuordnung der Werkzeuglängenkomponenten zu den Geometrieachsen		
	Länge L1	Länge L2	Länge L3
= x17	Y	X	Z
= x18	X	Z	Y
= x19	Z	Y	X
= -x17	X	Y	Z

SD42940	Zuordnung der Werkzeuglängenkomponenten zu den Geometrieachsen		
	Länge L1	Länge L2	Länge L3
= -x18	Z	X	Y
= -x19	Y	Z	X

Tabelle 2-4 Fräs- / Spezialwerkzeuge ($TC_DP1 <> 400 … 599)

SD42940	Zuordnung der Werkzeuglängenkomponenten zu den Geometrieachsen		
	Länge L1	Länge L2	Länge L3
= x17	Z	Y	X
= x18	Y	X	Z
= x19	X	Z	Y
= -x17	Z	X	Y
= -x18	Y	Z	X
= -x19	X	Y	Z

Jeder Wert ungleich 0, der nicht gleich einem der sechs aufgeführten Werte ist, wird wie der Wert "17" (für Fräs-/Spezialwerkzeuge) bzw. "18" (für Dreh-/Schleifwerkzeuge) bewertet.

Hinweis

Zuordnung der Werkzeugorientierungskomponenten

Die Zuordnung der Werkzeugorientierungskomponenten wird durch SD42940 $SC_TOOL_LENGTH_CONST nicht beeinflusst.

Gegebenenfalls müssen folgende Settingdaten angepasst werden:

- SD42954 $SC_TOOL_ORI_CONST_M
- SD42956 $SC_TOOL_ORI_CONST_T

Siehe Kapitel "Werkzeugorientierung bei Ebenenwechsel (Seite 178)".

2.11.3 Zuordnung unabhängig vom Werkzeugtyp

Mit dem folgenden Settingdatum kann eine vom tatsächlichen Werkzeugtyp ($TC_DP1) unabhängige Zuordnung der Werkzeuglängenkomponenten (Länge, Verschleiß und Basismaß) zu den Geometrieachsen festgelegt werden:

SD42950 $SC_TOOL_LENGTH_TYPE = <Wert>

<Wert>	Zuordnung der Werkzeuglängenkomponenten
0 (Standardwert)	Entsprechend der Angabe in der Systemvariablen $TC_DP1[...]. Es werden folgende Werkzeugtypen unterschieden: • $TC_DP1[...] == 400 ... 599 ⇒ Dreh-/Schleifwerkzeuge • $TC_DP1[...] <> 400 ... 599 ⇒ Fräs-/Spezialwerkzeuge
1	Immer wie bei Fräs-/Spezialwerkzeugen.
2	Immer wie bei Dreh-/Schleifwerkzeugen.

<Wert>	Zuordnung der Werkzeuglängenkomponenten	
3	Aktiviert das Settingdatum SD42942 $SC_TOOL_LENGTH_CONST_T. Mit dieser Einstellung ist es möglich, die bei einem Ebenenwechsel wirksame Zuordnung der Werkzeuglängenkomponenten (Seite 176) getrennt für Fräs-/Spezialwerkzeuge und Dreh-/Schleifwerkzeuge festzulegen:	
	SD42940 $SC_TOOL_LENGTH_CONST	Zuordnung für Fräs-/Spezialwerkzeuge
	SD42942 $SC_TOOL_LENGTH_CONST_T	Zuordnung für Dreh-/Schleifwerkzeuge
	Beispiel: SD42940 $SC_TOOL_LENGTH_CONST = 17 SD42942 $SC_TOOL_LENGTH_CONST_T = 18 Längenkorrekturen für Fräs-/Spezialwerkzeuge wirken in der Ebene G17, Längenkorrekturen für Dreh-/Schleifwerkzeuge wirken in der Ebene G18.	
> 3	Wie bei SD42950 = 0.	

2.12 Achsparallele Werkzeugorientierung

2.12.1 Werkzeuggrundorientierung

Die Werkzeuggrundorientierung ergibt sich aus der aktiven Bearbeitungsebene:

Arbeitsebene	Werkzeugorientierung (Grundstellung)
G17 (X/Y)	parallel zu Z
G18 (Z/X)	parallel zu Y
G19 (Y/Z)	parallel zu X

2.12.2 Werkzeugorientierung bei Ebenenwechsel

Wie sich die Werkzeugorientierung bei einem Wechsel der Bearbeitungsebene verändert, ist abhängig von der Einstellung der Settingdaten:

- SD42954 $SC_TOOL_ORI_CONST_M (für Fräs-/Spezialwerkzeuge)
- SD42956 $SC_TOOL_ORI_CONST_T (für Dreh-/Schleifwerkzeuge)

Standardverhalten

In der Standardeinstellung (SD42954 und SD42956 = 0) ändert sich die Werkzeugorientierung bei einem Ebenenwechsel wie folgt:

Ebenenwechsel	Änderung der Werkzeugorientierung
G17 → G18	1. Drehung um -90° um die Z-Koordinate 2. Drehung um -90° um die X-Koordinate
G18 → G19	
G19 → G17	
G18 → G17	1. Drehung um 90° um die X-Koordinate 2. Drehung um 90° um die Z-Koordinate
G19 → G18	
G17 → G19	

Konstante Werkzeugorientierung bei einem Ebenenwechsel

Wird SD42954 bzw. SD42956 ungleich null gesetzt, so wird ein rechtshändiges, orthogonales Werkzeugkoordinatensystem definiert, das bei einem Wechsel der Bearbeitungsebene (G17-G19) unverändert bleibt. Das Orientierungskoordinatensystem wird bestimmt durch den Orientierungsvektor und einen darauf senkrecht stehenden Orientierungsnormalenvektor. Die Grundorientierung wird dabei durch die Einer- und Zehnerstelle des Settingsdatums bestimmt:

Tabelle 2-5 Fräs- / Spezialwerkzeuge ($TC_DP1 <> 400 … 599)

SD42954	Koordinatensystem der Werkzeugorientierung	
	Orientierungsvektor	Orientierungsnormalenvektor
= xx17	(0, 0, 1)	(0, 1, 0)
= xx18	(0, 1, 0)	(1, 0, 0)
= xx19	(1, 0, 0)	(0, 0, 1)

Tabelle 2-6 Dreh- / Schleifwerkzeuge ($TC_DP1 = 400 … 599)

SD42956	Koordinatensystem der Werkzeugorientierung	
	Orientierungsvektor	Orientierungsnormalenvektor
= xx17	(0, 0, 1)	(0, 1, 0)
= xx18	(0, 1, 0)	(1, 0, 0)
= xx19	(1, 0, 0)	(0, 0, 1)

Jeder Wert ungleich 0, der nicht gleich einem der drei aufgeführten Werte ist, wird wie der Wert "17" (für Fräs-/Spezialwerkzeuge) bzw. "18" (für Dreh-/Schleifwerkzeuge) bewertet.

Weitere Informationen
Eine ausführliche Beschreibung von SD42954 und SD42956 findet sich im Listenhandbuch Systemvariablen.

Hinweis

Bei Werkzeugen, bei denen die Orientierung mittels Schneidendaten ($TC_DPV…) definiert ist, wird SD42954 bzw. SD42956 normalerweise ignoriert (siehe Kapitel "Parametrierbare Werkzeuggrundorientierung (Seite 180)").

2.13 Parametrierbare Werkzeuggrundorientierung

2.13.1 Funktion

Hinweis
Werkzeug T und Schneide D

Im Folgenden steht im Zusammenhang mit den Systemvariablen der Werkzeuggrundorientierung die Schreibweise [...] für [<t>, <d>]. Dabei bezeichnet <t> die Nummer des Werkzeugs T=<t> und <d> die Nummer der Werkzeugschneide D=<d>.

Durch die Funktion "Parametrierbare Werkzeuggrundorientierung" kann mit den folgenden Systemvariablen jeder Werkzeugschneide eine eigene Grundorientierung zugeordnet werden:

- Auswahl eines vordefinierten Orientierungsvektors
  - $TC_DPV[...]: Orientierungsvektor (Werte 1 ... 6)
- Definition eines Orientierungsvektors
  - $TC_DPV3[...]: L1-Komponente des Orientierungsvektors
  - $TC_DPV4[...]: L2-Komponente des Orientierungsvektors
  - $TC_DPV5[...]: L3-Komponente des Orientierungsvektors
- Definition eines Normalenvektors zum Orientierungsvektor
  - $TC_DPVN3[...]: L1-Komponente des Normalenvektors
  - $TC_DPVN4[...]: L2-Komponente der Normalenvektors
  - $TC_DPVN5[...]: L3-Komponente der Normalenvektors

Ist die Systemvariable $TC_DPV[...] gleich Null, definieren die drei Systemvariablen $TC_DPV3 - 5[...] den Richtungsvektor der Werkzeuggrundorientierung. Zusätzlich können die drei Systemvariablen $TC_DPVN3 - 5[...] die Orientierung des Werkzeugs in der Ebene senkrecht zum Orientierungsvektor definieren. Die Beträge der Orientierungsvektoren sind dabei unerheblich.

2.13.2 Inbetriebnahme

2.13.2.1 Aktivierung

Die "Parametrierbare Werkzeuggrundorientierung" wird über folgendes Maschinendatum aktiviert:

MD18114 $MN_MM_ENABLE_TOOL_ORIENTATION = <Wert>

<Wert>	Bedeutung
0	Die Funktion "Parametrierbare Werkzeuggrundorientierung" ist nicht aktiv.
1	Aktivierung der Systemvariablen: • Funktionsanwahl: $TC_DPV[...] Mit der Systemvariablen $TC_DPV[...] = 1, 2, ... 6 kann jeder Werkzeugschneide D=<d> eines Werkzeugs T=<t> eine von sechs vordefinierten Werkzeugrundorientierungen zugeordnet werden.
2	Aktivierung der Systemvariablen: • Funktionsanwahl: $TC_DPV[...] • Orientierungsvektor: $TC_DPV3[...], $TC_DPV4[...] und $TC_DPV5[...] Mit den Systemvariablen $TC_DPV[...] = 0 und $TC_DPV3[...], $TC_DPV4[...] und $TC_DPV5[...] kann jeder Werkzeugschneide D=<d> eines Werkzeugs T=<t> ein beliebiger Orientierungsvektor zugeordnet werden.
3	Aktivierung der Systemvariablen: • Funktionsanwahl: $TC_DPV[...] • Orientierungsvektor: $TC_DPV3[...], $TC_DPV4[...] und $TC_DPV5[...] • Normalenvektor zum Orientierungsvektor: $TC_DPVN3[...], $TC_DPVN4[...] und $TC_DPVN5[...] Mit der Einstellung $TC_DPV[...] = 0 kann mit den Systemvariablen $TC_DPV3[...], $TC_DPV4[...] und $TC_DPV5[...] jeder Werkzeugschneide D=<d> eines Werkzeugs T=<t> ein beliebiger Orientierungsvektor zugeordnet werden. Zusätzlich kann mit den Systemvariablen $TC_DPVN3[...], $TC_DPVN4[...] und $TC_DPVN5[...] ein Normalenvektor zum Orientierungsvektor definiert werden.

2.13.2.2 Parametrierung

Zuordnung der Systemvariablen $TC_DPVx[...]

Entsprechend der Angabe in der Systemvariablen $TC_DP1[...] werden folgende Werkzeugtypen unterschieden:

- $TC_DP1[...] == 400 ... 599 ⇒ Dreh-/Schleifwerkzeuge
- $TC_DP1[...] <> 400 ... 599 ⇒ Fräs-/Spezialwerkzeuge

Die in $TC_DPVx[...] programmierte Werkzeuggrundorientierung wird abhängig vom Werkzeugtyp den Koordinatenachsen zugeordnet.

Durch die Einstellung des Settingdatums SD42950 $SC_TOOL_LENGTH_TYPE wird der Werkzeugtyp möglicherweise umgeschaltet (wie im Kapitel "Zuordnung unabhängig vom Werkzeugtyp (Seite 177)" für Werkzeuglängenkomponenten beschrieben).

Wenn in den Settingdaten SD42954 $SC_TOOL_ORI_CONST_M und SD42956 $SC_TOOL_ORI_CONST_T (Seite 178) keine abweichende Orientierung definiert ist, dann gilt für die Zuordnung der Systemvariablen $TC_DPVx[...] der Standardfall (G17 für Fräs-/Spezialwerkzeuge, G18 für Dreh-/Schleifwerkzeuge):

Werkzeugtyp	TC_DPV3[…]	TC_DPV4[…]	TC_DPV5[…]
Dreh-/Schleifwerkzeuge	X	Z	Y
Fräs-/Spezialwerkzeuge	Z	Y	X

Hinweis

SD42954 / SD42956

Die Zuordnung der Systemvariablen $TC_DPVx[...] kann durch SD42954 $SC_TOOL_ORI_CONST_M und SD42956 $SC_TOOL_ORI_CONST_T nur dann geändert werden, wenn die 1000er-Stelle gleich "1" ist.

Unabhängig vom Wert des Settingdatums SD42950 $SC_TOOL_LENGTH_TYPE wird SD42954 nur für ein Werkzeug wirksam, für den der Parameter $TC_DP1[...] ein Fräswerkzeug definiert. Analog wird SD42956 nur für ein Werkzeug wirksam, für den der Parameter $TC_DP1[...] ein Dreh-/Schleifwerkzeuge definiert.

2.13.3 Programmierung

Mit den Systemvariablen $TC_DPV3 - 5[...] bzw. $TC_DPV[...] kann jeder Werkzeugschneide eine eigene Grundorientierung zugeordnet werden.

Einstellmöglichkeiten

Grundsätzlich gibt es folgende Einstellmöglichkeiten:

- **$TC_DPV[...] == 0 UND $TC_DPV3 - 5[...] == 0**
  Der Vektor für die Werkzeuggrundorientierung ergibt sich aus der aktiven Bearbeitungsebene:
  - G17: Z-Koordinate
  - G18: Y-Koordinate
  - G19: X-Koordinate

  Siehe auch "Achsparallele Werkzeugorientierung (Seite 178)".
- **$TC_DPV[...] == 0 UND $TC_DPV3 - 5[...] <> 0**
  Der Vektor für die Werkzeuggrundorientierung ist durch $TC_DPV3 - 5[...] vorgegeben:
  - `$TC_DPV3[...] = <Wert in L1-Richtung>`
  - `$TC_DPV4[...] = <Wert in L2-Richtung>`
  - `$TC_DPV5[...] = <Wert in L3-Richtung>`

  Beispiel:
  Werkzeuggrundorientierung in Richtung der Winkelhalbierenden in der L1/L3-Ebene, d. h., bei einem Fräswerkzeug und aktiver Ebene G17 in Richtung der Winkelhalbierenden der ZX-Ebene.
  ```
  $TC_DPV[1,1]  = 0
  $TC_DPV3[1,1] = 1.0
  $TC_DPV4[1,1] = 0.0
  $TC_DPV5[1,1] = 1.0
  ```
- **$TC_DPV[...] == 1, 2, ... 6**
  Der Vektor für die Werkzeuggrundorientierung ist durch $TC_DPV[...] vorgegeben. Welche Werkzeuggrundorientierungen vordefiniert und über $TC_DPV[...] auswählbar sind, zeigen folgende Tabellen.

Auswahl eines vordefinierten Orientierungsvektors

```
$TC_DPV[...] = <Wert>
```

Tabelle 2-7 Dreh- / Schleifwerkzeuge ($TC_DP1 = 400 … 599)

<Wert>	Bedeutung		
0	$TC_DPV3[...]	$TC_DPV5[...]	$TC_DPV4[...]
1	0	1	0
2	1	0	0
3	0	0	1
4	0	-1	0
5	-1	0	0
6	0	0	-1

Tabelle 2-8 Fräs- / Spezialwerkzeuge ($TC_DP1 <> 400 … 599)

<Wert>	Bedeutung		
0	$TC_DPV5[...]	$TC_DPV4[...]	$TC_DPV3[...]
1	0	0	1
2	0	1	0
3	1	0	0
4	0	0	-1
5	0	-1	0
6	-1	0	0

Beispiele

Dreh-/Schleifwerkzeuge:

```
$TC_DPV[...] = 2    entspricht    $TC_DPV3[...] = 1
                                  $TC_DPV4[...] = 0
                                  $TC_DPV5[...] = 0
```

Fräs-/Spezialwerkzeuge:

```
$TC_DPV[...] = 3    entspricht    $TC_DPV3[...] = 0
                                  $TC_DPV4[...] = 0
                                  $TC_DPV5[...] = 1
```

2.13.4 Beispiele

Es wird ein Fräswerkzeug mit der Länge L1=10 definiert, dessen Werkzeuggrundorientierung in der Winkelhalbierenden der X-Z-Ebene liegt.

Beispiel 1

Programmcode	Kommentar
$SC_TOOL_LENGTH_TYPE=2	; Zuordnung zu den Koordinatenachsen wie bei Dreh-/ Schleifwerkzeugen.
$SC_TOOL_ORI_CONST_M=1019	; Arbeitsebene: G19
N10 $TC_DP1[1,1]=120	; Werkzeugtyp: Fräswerkzeug
N20 $TC_DP3[1,1]=10	; Längenkorrekturvektor: L1=10
N30 $TC_DPV[1,1]=0	; Werkzeuggrundorientierung über $TC_DPV3-5
N40 $TC_DPV3[1,1]=1	; Z-Koordinate
N50 $TC_DPV4[1,1]=0	; Y-Koordinate
N60 $TC_DPV5[1,1]=1	; X-Koordinate
N70 TRAFOON(...)	; Transformation aktivieren.
N80 G17 F1000 X0 Y0 Z0 T1 D1	; Werkzeug aktivieren und Nullposition anfahren.
	; Die Ebenenanwahl G18 im Satz N80 ist nur für die Längenkorrekturen relevant.

Programmcode	Kommentar
...	

Beispiel 2

Programmcode	Kommentar
N10 $TC_DP1[1,1]=120	; Werkzeugtyp: Fräswerkzeug
N20 $TC_DP3[1,1]=10	; Längenkorrekturvektor: L1=10
N30 $TC_DPV[1,1]=0	; Werkzeuggrundorientierung über $TC_DPV3-5
N40 $TC_DPV3[1,1]=1	; Z-Koordinate
N50 $TC_DPV4[1,1]=0	; Y-Koordinate
N60 $TC_DPV5[1,1]=1	; X-Koordinate
N70 TRAFOON(...)	; Transformation aktivieren.
N80 G17 F1000 X0 Y0 Z0 T1 D1	; Werkzeug aktivieren und Nullposition anfahren.
	; Wegen Werkzeuglänge L1=10 =>
	; Maschinenpositionen: X=0, Y=0, Z=10
N90 MOVT=10	; Inkr. Verfahrbewegung in Werkzeugrichtung.
	; Resultierende Achspositionen: X=7.071 Y=0 Z=17.071
N100 M30	

2.14 Werkzeugkorrektur-Sonderbehandlungen

2.14.1 Relevante Settingdaten

SD42900- 42960

Über die Settingdaten SD42900 - SD42940 lassen sich bezüglich der Werkzeugkorrektur folgende Einstellungen vornehmen:

- Vorzeichen der Werkzeuglänge
- Vorzeichen des Verschleiß
- Verhalten der Verschleißkomponenten beim Spiegeln von Geometrieachsen
- Verhalten der Verschleißkomponenten beim Wechsel der Bearbeitungsebene über Settingdaten

- Zuordnung der Werkzeuglängenkomponenten unabhängig vom tatsächlichen Werkzeugtyp
- Transformation der Verschleißkomponenten in ein geeignetes Koordinatensystem zur Beeinflussung der wirksamen Werkzeuglänge

Hinweis

In der nachfolgenden Beschreibung ist unter Verschleiß die Summe der Werte folgender Komponenten zu verstehen:

- Verschleißwerte: $TC_DP12 bis $TC_DP20
- Summenkorrektur, bestehend aus:
  - Verschleißwerte: $SCPX3 bis $SCPX11
  - Einrichtewerte: $ECPX3 bis $ECPX11

Ausführliche Informationen zu Summen- und Werkzeugkorrekturen finden sich in:

Weitere Informationen
Funktionshandbuch Werkzeugverwaltung

Programmierhandbuch Grundlagen; Werkzeugkorrekturen

Erforderliche Settingdaten

- SD42900 $SC_MIRROR_TOOL_LENGTH (Spiegeln von Werkzeuglängenkomponenten und Komponenten des Basismaßes) (Seite 186)
- SD42910 $SC_MIRROR_TOOL_WEAR (Spiegeln von Verschleißwerten der Werkzeuglängenkomponenten) (Seite 186)
- SD42920 $SC_WEAR_SIGN_CUTPOS (Vorzeichenbewertung der Verschleißkomponenten) (Seite 187)
- SD42930 $SC_WEAR_SIGN (Invertiert die Vorzeichen der Verschleißmaße) (Seite 187)
- SD42935 $SC_WEAR_TRANSFORM (Transformation der Verschleißwerte) (Seite 205)
- SD42940 $SC_TOOL_LENGTH_CONST (Zuordnung der Werkzeuglängenkomponenten zu den Geometrieachsen) (Seite 176)
- SD42950 $SC_TOOL_LENGTH_TYPE (Zuordnung der Werkzeuglängenkomponenten unabhängig vom Werkzeugtyp) (Seite 177)
- SD42960 $SC_TOOL_TEMP_COMP (Werkzeuglängenoffsets) (Seite 189)

2.14.2 Werkzeuglängen spiegeln

Aktivierung

Die Aktivierung der Werkzeuglängen-Spiegelung erfolgt über das Settingdatum:

SD42900 $SC_MIRROR_TOOL_LENGTH <> 0 (TRUE) (Vorzeichenwechsel Werkzeuglänge beim Spiegeln)

Funktion

Durch Vorzeicheninvertierung werden folgende Komponenten gespiegelt:

- Werkzeuglängen: $TC_DP3, $TC_DP4, $TC_DP5
- Basismaße: $TC_DP21, $TC_DP22, $TC_DP23

Die Spiegelung erfolgt für alle Basismaße, deren zugehörige Achsen gespiegelt sind. Verschleißwerte werden **nicht** mitgespiegelt.

Verschleißwerte spiegeln

Zum Spiegeln der Verschleißwerte ist folgendes Settingdatum zu setzen:

SD42910 $SC_MIRROR_TOOL_WEAR <> 0 (Vorzeichenwechsel Werkzeugverschleiß beim Spiegeln)

Durch Vorzeicheninvertierung werden die Verschleißwerte der Werkzeuglängenkomponenten, deren zugehörige Achsen gespiegelt sind, gespiegelt.

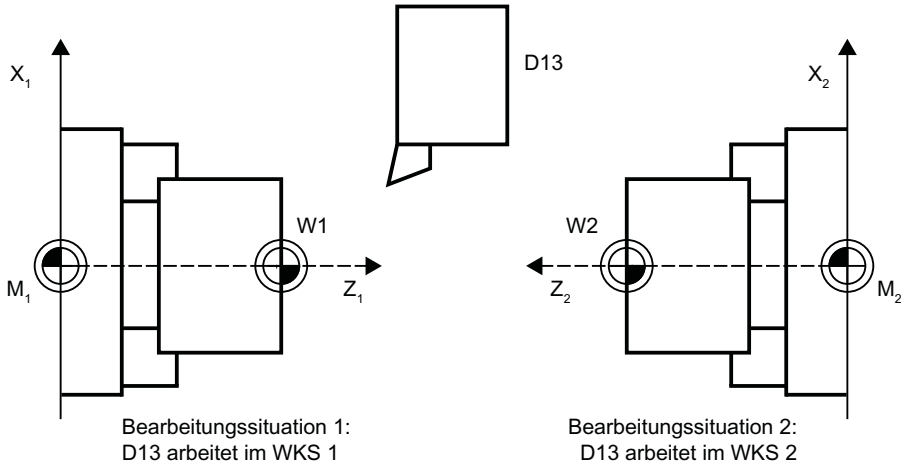


Bild 2-46 Anwendungsbeispiel: Doppelspindel-Drehmaschine

2.14.3 Verschleißlängen spiegeln

Aktivierung

Die Aktivierung der Verschleißlängen-Spiegelung erfolgt über:

SD42920 $SC_WEAR_SIGN_CUTPOS <> 0 (TRUE) (Vorzeichen des Verschleißes bei Werkzeugen mit Schneidenlage)

Funktion

Schneidenlage	Länge 1	Länge 2
1	---	---
2	---	invertiert
3	invertiert	invertiert
4	invertiert	---
5	---	---
6	---	---
7	---	invertiert
8	invertiert	---
9	---	---

Bei Werkzeugtypen ohne relevanter Schneidenlage wird keine Verschleißlängen-Spiegelung durchgeführt.

Hinweis

Die Spiegelung (Vorzeicheninvertierung) in einer oder mehreren Komponenten kann sich aufheben durch gleichzeitige Aktivierung der Funktionen:

Werkzeuglängen-Spiegelung (SD42900 <> 0)

Und:

Verschleißlängen-Spiegelung (SD42920 <> 0)

SD42930 $SC_WEAR_SIGN

Settingdatum ungleich Null:

Das Vorzeichen aller Verschleißmaße wird invertiert. Es wirkt sowohl auf die Werkzeuglänge als auch auf die übrigen Größen wie Werkzeugradius, Verrundungsradius usw.

Wird ein positives Verschleißmaß eingegeben, wird somit das Werkzeug "kürzer" und "dünner".

Wirksamwerden der veränderten Settingdaten

Die Neubewertung von Werkzeugkomponenten bei einer Änderung der beschriebenen Settingdaten wird erst wirksam, wenn das nächste Mal eine Werkzeugschneide angewählt wird. Ist ein Werkzeug bereits aktiv und die Bewertung der Daten dieses Werkzeugs soll verändert wirksam werden, muss dieses Werkzeug erneut angewählt werden.

Beispiel:

```
N10 $SC_WEAR_SIGN = 0              ; Keine Vorzeichenumkehr der Verschleißwerte
N20 $TC_DP1[1,1] = 120             ; Schaftfräser
N30 $TC_DP6[1,1] = 100             ; Werkzeugradius 100 mm
N40 $TC_DP15[1,1] = 1              ; Verschleißmaß des Werkzeugradius 1 mm, resul-
                                     tierender Werkzeugradius 101 mm
```

```
N100 T1 D1 G41 X150 Y20
....
N150 G40 X300N10
....
N200 $SC_WEAR_SIGN = 1              ; Vorzeichenumkehr aller Verschleißwerte, durch
                                      Neuanwahl (D1) wird der neue Radius von 99 mm
                                      wirksam. Ohne D1 wäre der wirksame Radius wei-
                                      terhin 101 mm.
N300 D1 G41 X350 Y-20
N310 ....
```

Entsprechendes gilt für den Fall, dass sich die resultierende Werkzeuglänge ändert, weil der Spiegelungszustand einer Achse geändert wurde. Das Werkzeug muss nach dem Spiegelbefehl erneut angewählt werden, damit die geänderten Werkzeuglängenkomponenten wirksam werden.

2.14.4 Werkzeuglängen im WKS unter Berücksichtigung der Orientierung

Wechsel Werkzeug oder Arbeitsebene

Die angezeigten Werte für das Werkzeug entsprechen der Ausdehnung im WKS. Soll ein Werkzeugträger mit schräger Werkzeugaufhängung verwendet werden, so ist darauf zu achten, dass die verwendete Transformation diesen unterstützt. Ist dies nicht der Fall, werden falsche Werkzeugabmessungen angezeigt. Beim Wechseln der Arbeitsebene von `G17` auf `G18` oder `G19` sollte ebenfalls sichergestellt werden, dass die Transformation auch für diese Arbeitsebenen verwendet werden kann. Handelt es sich um eine Transformation, die nur für `G17`-Bearbeitung vorgesehen ist, so werden beim Ebenenwechsel auch weiterhin die Abmessungen für ein Werkzeug in Z-Richtung angezeigt.

Bei abgeschalteter Transformation wird das Basiswerkzeug in X-, Y- oder Z-Richtung, je nach Arbeitsebene, angezeigt. Ein programmierter Werkzeugträger wird berücksichtigt. Diese Abmessungen des Werkzeugs werden beim Verfahren ohne Transformation nicht verändert.

2.14.5 Werkzeuglängenoffsets in Werkzeugrichtung

Temperaturkompensation in Echtzeit

Bei 5-Achs-Maschinen mit beweglichem Werkzeug können in den Bearbeitungsköpfen Temperaturveränderungen auftreten, die wiederum Ausdehnungsänderungen hervorrufen und in Form von Längenausdehnung auf die Werkzeugspindel übertragen. Ein typischer Fall ist bei 5-Achs-Köpfen z.B. die Wärmeausdehnung in Richtung der Spindellängsachse.

Diese Wärmeausdehnung kann auch bei vorhandener Werkzeugorientierung kompensiert werden, indem die Temperaturkompensationswerte nicht mehr den Maschinenachsen, sondern dem Werkzeug zugeordnet werden. Damit sind Längenänderungen bei Änderung der Werkzeugorientierung kompensierbar.

Mit Hilfe der Orientierungstransformation, deren Richtung von der aktuellen Werkzeugorientierung abhängt, können Bewegungen in Echtzeit überlagert und gleichzeitig mitgedreht werden. Dabei werden die Kompensationswerte ständig im Werkzeugkoordinatensystem mitgeführt.

Die Temperaturkompensation wird nur wirksam, wenn die zu kompensierende Achse auch wirklich referenziert ist.

Aktivierung

Die Temperaturkompensation in Werkzeugrichtung wird aktiviert, indem man das folgende Maschinendatum auf einen Wert **ungleich Null** setzt:

MD20390 $MC_TOOL_TEMP_COMP_ON (Aktivierung der Temperaturkompensation für Werkzeuglänge)

Zusätzlich muss das Bit 2 gesetzt sein für jede betroffene Kanalachse im Maschinendatum:

MD32750 $MA_TEMP_COMP_TYPE[<Achsindex>] (Temperaturkompensationstyp)

Das können mehr als drei Achsen sein, wenn als Folge von Geoachstausch oder Transformationsumschaltungen sukzessive mehr als drei Kanalachsen zeitweise zu Geometrieachsen werden können. Ist dieses Bit für eine bestimmte Kanalachse nicht gesetzt, kann in dieser Achse kein Kompensationswert wirksam werden. Andere Achsen bleiben davon jedoch unbeeinflusst. Ein Alarm wird in diesem Fall nicht ausgegeben.

Wirkungsbereiche

Die Temperaturkompensation in Werkzeugrichtung wirkt nur bei den generischen 5-Achstransformationen mit:

- Trafotyp 24
  Zwei Achsen drehen das Werkzeug.
- Trafotyp 56
  Eine Achse dreht das Werkzeug, die andere Achse dreht das Werkstück ohne Temperaturkompensation.

Bei der generischen 5-Achstransformation mit:

- Trafotyp 40
  Beim drehbaren Werkstück ist die Werkzeugorientierung konstant, so dass in diesem Fall eine Bewegung der Rundachsen der Maschine keinen Einfluss auf die Richtung der Temperaturkompensation hat.

Die Temperaturkompensation in Werkzeugrichtung wirkt außerdem bei den Orientierungstransformationen (sind keine generischen 5-Achs-Trafos) mit:

- Trafotyp 64 bis 69
  Drehbare Linearachse.

Hinweis

Bei allen anderen Transformationstypen ist die Temperaturkompensation aktivierbar. Sie wird aber nicht durch eine Änderung der Werkzeugorientierung beeinflusst. Die Achsen verfahren so, als ob keine Orientierungstransformation mit einer Temperaturkompensation aktiv wäre.

Grenzwerte

Die Kompensationswerte werden auf Maximalwerte begrenzt durch das Maschinendatum:

MD20392 $MC_TOOL_TEMP_COMP_LIMIT[0...2] (Maximale Temperaturkompensation für Werkzeuglänge)

Die Vorbelegung des Grenzwertes ist 1 mm. Wird ein Temperaturkompensationswert vorgegeben, der größer als dieser Grenzwert ist, wird dieser ohne Alarm begrenzt.

SD42960

Die drei Temperaturkompensationswerte bilden zusammen einen Kompensationsvektor und sind enthalten im Settingdatum:

SD42960 $SC_TOOL_TEMP_COMP[0...2] (Temperaturkompensation bezogen auf Werkzeuge)

Die Settingdaten werden z.B. mittels Synchronaktionen oder von der PLC aus vom Anwender beschrieben. Die Kompensationswerte können deshalb auch für andere Kompensationszwecke verwendet werden.

Alle drei Kompensationswerte wirken in Grundstellung bzw. bei ausgeschalteter Orientierungstransformation in Richtung der drei Geometrieachsen (in der typischen Reihenfolge X, Y, Z). Die Zuordnung der Komponenten zu den Geometrieachsen ist unabhängig vom Werkzeugtyp (Dreh- / Fräswerkzeuge bzw. Schleifwerkzeuge) und der angewählten Bearbeitungsebene `G17` bis `G19`. Werteveränderungen in den Settingdaten werden sofort wirksam.

Orientierbarer Werkzeugträger

Ist ein orientierbarer Werkzeugträger aktiv, so wird der Vektor der Temperaturkompensation bei einer Orientierungsänderung mitgedreht. Das gilt unabhängig von einer eventuell aktiven Orientierungstransformation.

Ist ein orientierbarer Werkzeugträger zusammen mit einer generischen 5-Achs-Transformation oder einer Transformation mit drehbarer Linearachse aktiv, so wird der Vektor der Temperaturkompensation beiden Drehungen unterworfen.

Hinweis

Die Transformationen mit drehbaren Linearachsen berücksichtigen zwar die Änderung des Werkzeugvektors (Länge), jedoch **nicht** dessen Orientierungsänderung, die durch einen orientierbaren Werkzeugträger bewirkt wird.

Die Temperaturkompensationswerte folgen unmittelbar jeder wirksam werdenden Orientierungsänderung. Dies gilt insbesondere auch dann, wenn eine Orientierungstransformation ein- oder ausgeschaltet wird.

Gleiches gilt bei einer Änderung der Geometrieachsenzuordnung zu den Kanalachsen. Der Temperaturkompensationswert für eine Achse, die z.B. nach einem Transformationswechsel nicht mehr Geometrieachse ist, wird (interpolatorisch) auf Null reduziert. Umgekehrt baut sich in einer Achse, die nach einer Umschaltung zur Geometrieachse wird, ein eventuell vorhandener Temperaturkompensationswert sofort auf.

Beispiele

Temperaturkompensation in Werkzeugrichtung

Es ist eine 5-Achs-Maschine mit drehbarem Werkzeug gegeben, bei der das Werkzeug um die C- und um die B-Achse gedreht werden kann.

In Grundstellung ist das Werkzeug parallel zur Z-Achse. Wird die B-Achse um 90 Grad gedreht, zeigt das Werkzeug in X-Richtung.

Deshalb wirkt ein Temperaturkompensationswert im folgenden Settingdatum bei aktiver Transformation ebenfalls in Richtung der Maschinen X-Achse:

SD42960 $SC_TOOL_TEMP_COMP[2] (Temperaturkompensation bezogen auf Werkzeuge)

Wird in dieser Stellung die Transformation ausgeschaltet, ist die Werkzeugorientierung definitionsgemäß wieder parallel zur Z-Achse und weicht damit von der tatsächlichen Orientierung ab. Der Temperaturoffset in Richtung der X-Achse wird deshalb abgebaut und gleichzeitig in Z-Richtung aufgebaut.

Es sei eine 5-Achs-Maschine mit drehbarem Werkzeug (Trafotyp 24) gegeben. Die wesentlichen Maschinendaten sind im Folgenden dargestellt:

- Die erste Rundachse dreht um Z C-Achse.
- Die zweite Rundachse dreht um Y B-Achse.

Die wesentlichen Maschinendaten sind in der folgenden Tabelle dargestellt:

Maschinendatum	Wert	Anmerkung
MD20390 $MC_TOOL_TEMP_COMP_ON	= TRUE	Temperaturkompensation aktiv
MD32750 $MA_TEMP_COMP_TYPE[AX1]	= 4	Korrektur in Werkzeugrichtung
MD32750 $MA_TEMP_COMP_TYPE[AX2]	= 4	Korrektur in Werkzeugrichtung
MD32750 $MA_TEMP_COMP_TYPE[AX3]	= 4	Korrektur in Werkzeugrichtung
		Zuordnung des Trafotyps 24:
MD24100 $MC_TRAFO_TYPE_1	= 24	Trafotyp 24 im 1. Kanal
MD24110 $MC_TRAFO_AXES_IN_1[0]	= 1	1. Achse der Transformation
MD24110 $MC_TRAFO_AXES_IN_1[1]	= 2	2. Achse der Transformation
MD24110 $MC_TRAFO_AXES_IN_1[2]	= 3	3. Achse der Transformation
MD24110 $MC_TRAFO_AXES_IN_1[3]	= 5	5. Achse der Transformation
MD24110 $MC_TRAFO_AXES_IN_1[4]	= 4	4. Achse der Transformation
MD24120 $MC_TRAFO_GEOAX_ASSIGN_TAB_1[0]	= 1	Geoachse zu Kanalachse 1
MD24120 $MC_TRAFO_GEOAX_ASSIGN_TAB_1[1]	= 2	Geoachse zu Kanalachse 2
MD24120 $MC_TRAFO_GEOAX_ASSIGN_TAB_1[2]	= 3	Geoachse zu Kanalachse 3
MD24570 $MC_TRAFO5_AXIS1_1[0]	= 0.0	
MD24570 $MC_TRAFO5_AXIS1_1[1]	= 0.0	Richtung
MD24570 $MC_TRAFO5_AXIS1_1[2]	= 1.0	1. Rundachse ist parallel Z

Maschinendatum	Wert	Anmerkung
MD24572 $MC_TRAFO5_AXIS1_2[0]	= 0.0	Richtung
MD24572 $MC_TRAFO5_AXIS1_2[1]	= 1.0	2. Rundachse ist parallel Y
MD24572 $MC_TRAFO5_AXIS1_2[2]	= 0.0	
MD25574 $MC_TRAFO5_BASE_ORIENT_1[0]	= 0.0	
MD25574 $MC_TRAFO5_BASE_ORIENT_1[1]	= 0.0	Werkzeuggrundorientierung
MD25574 $MC_TRAFO5_BASE_ORIENT_1[2]	= 1.0	in Z-Richtung

Temperaturkompensationswerte im NC-Programm

Die den beiden Achsen X und Z zugeordneten Kompensationswerte sind ungleich Null und werden bei der Temperaturkompensation bezüglich der Werkzeuglänge berücksichtigt. Die jeweils erreichten Maschinenachspositionen sind als Kommentare in den Programmzeilen angegeben.

Programmcode	Kommentar
$SC_TOOL_TEMP_COMP[0] = -0.3	; Kompensationswert in X
$SC_TOOL_TEMP_COMP[1] = 0.0	
$SC_TOOL_TEMP_COMP[2] = -1.0	; Kompensationswert in Z
	; Sollpositionen der Maschinenachsen
N10 G74 X0 Y0 Z0 A0 B0	; X Y Z
N20 X20 Y20 Z20 F10000	; 20.30 20.00 21.00
N30 TRAORI()	; 20.30 20.00 21.00
N40 X10 Y10 Z10 B90	; 11.00 10.00 9.70
N50 TRAFOOF	; 10.30 10.00 11.00
N60 X0 Y0 Z0 B0 C0	; 0.30 0.00 1.00
N70 M30	

Mit Ausnahme des Satzes `N40` wirkt die Temperaturkompensation immer in den Originalrichtungen, da das Werkzeug in Richtung der Grundorientierung zeigt. Das gilt insbesondere auch im Satz `N50`. Das Werkzeug zeigt zwar tatsächlich noch in Richtung der X-Achse, weil die B-Achse noch auf 90 Grad steht. Die zu berücksichtigende Orientierung ist aber wegen der bereits ausgeschalteten Transformation wieder parallel zu Z.

Maschinendatum	Wert	Anmerkung
MD20390 $MC_TOOL_TEMP_COMP_ON	= TRUE	Temperaturkompensation aktiv
MD32750 $MA_TEMP_COMP_TYPE[AX1]	= 4	Korrektur in Werkzeugrichtung
MD32750 $MA_TEMP_COMP_TYPE[AX2]	= 4	Korrektur in Werkzeugrichtung
MD32750 $MA_TEMP_COMP_TYPE[AX3]	= 4	Korrektur in Werkzeugrichtung

Weitere Informationen

- Weitere Erläuterungen zur "Temperaturkompensation" siehe:
  Funktionshandbuch Überwachungen und Kompensationen; Kompensationen
- Informationen zur "Generischen 5-Achstransformation" siehe:
  Funktionshandbuch Transformationen; Mehrachstransformationen

2.14.6 Besonderheiten bei orientierbaren Werkzeugträgern

Settingdaten SD42900 - SD42950

Die Settingdaten SD42900 - SD42950 wirken nicht auf die Komponenten eines eventuell aktiven orientierbaren Werkzeugträgers. Ein Werkzeug geht immer mit seiner gesamten resultierenden Länge (Werkzeuglänge + Verschleiß + Basismaß) in die Berechnung mit einem orientierbaren Werkzeugträger ein. Bei der Berechnung der resultierenden Gesamtlänge werden alle Änderungen, die durch die Settingdaten verursacht wurden, berücksichtigt.

Hinweis

Häufig werden beim Einsatz orientierbarer Werkzeugträger alle Werkzeuge für ein nicht gespiegeltes Grundsystem definiert, auch diejenigen, die nur bei Spiegelbearbeitung verwendet werden. Bei Bearbeitung mit gespiegelten Achsen wird dann der Werkzeugträger so gedreht, dass die tatsächliche Lage des Werkzeugs richtig beschrieben wird. Alle Werkzeuglängenkomponenten wirken dann automatisch in der richtigen Richtung, so dass sich eine Steuerung der Bewertung einzelner Komponenten über Settingdaten abhängig vom Spiegelungszustand einzelner Achsen erübrigt.

Der Einsatz orientierbarer Werkzeugträger ist auch dann sinnvoll, wenn der Maschinentyp es physikalisch nicht zulässt, Werkzeuge die mit verschiedenen Orientierungen fest installiert sind, auch zu drehen. Die Werkzeugvermaßung kann dann einheitlich in einer Grundorientierung vorgenommen werden, und die für die Bearbeitung relevanten Maße ergeben sich durch Drehungen eines virtuellen Werkzeugträgers.

2.15 Summen- und Einrichtekorrekturen

2.15.1 Allgemeine Informationen

Summenkorrekturen

Summenkorrekturen können als in der Bearbeitung **programmierbare Prozesskorrekturen** betrachtet werden und setzen sich aus all den Fehlergrößen zusammen (einschließlich dem Verschleiß), die die Abweichung des Werkstücks vom Sollmaß ausmachen.

Summenkorrekturen sind eine verallgemeinerte Form von Verschleiß. Sie sind Bestandteil der Werkzeugschneidendaten. Die Parameter der Summenkorrektur beziehen sich auf die geometrischen Daten einer Schneide.

Die Korrekturdaten einer Summenkorrektur werden über eine **DL-Nummer** angesprochen (**DL:** Location dependend; Korrekturen bezüglich des jeweiligen Einsatzortes).

Die Verschleißwerte einer D-Nummer beschreiben dagegen den physikalischen Verschleiß der Schneide, d.h. im speziellen Fall kann die Summenkorrektur dem Verschleiß der Schneide entsprechen.

Summenkorrekturen können allgemein eingesetzt werden; d.h. mit aktiver oder inaktiver Werkzeugverwaltung.

Die Summenkorrekturen werden durch Setzen entsprechender Maschinendaten aufgeteilt in:

- Summenkorrektur fein
- Summenkorrektur grob (Einrichtekorrektur)

Einrichtekorrektur

Die Einrichtekorrektur ist die vom Einrichter vor der Bearbeitung einzustellende Korrektur. Diese Werte belegen im NC eigenen Speicher. Dem Bediener wird dann nur die "Summenkorrekturen fein" über HMI zugänglich gemacht.

"Summenkorrektur fein" und "Summenkorrektur grob" werden NC-intern addiert. Dieser Wert soll im Folgenden als Summenkorrektur bezeichnet werden.

Hinweis

Die Funktion wird eingeschalten über die Maschinendatum-Einstellung:

MD18080 $MN_MM_TOOL_MANAGEMENT_MASK, Bit 8=1 (Stufenweise Speicher-Reservierung für die Werkzeugverw.)

Bei aktiven kinematischen Transformationen (z.B. 5-Achs-Transformation) wird die Werkzeuglänge zunächst unter Berücksichtigung der verschiedenen Verschleißkomponenten wie beschrieben berechnet. Diese gesamte Werkzeuglänge geht dann in die Transformation ein. Damit werden auch die Verschleißwerte unabhängig vom G-Befehl der Gruppe 56 - anders als beim orientierbaren Werkzeugträger - immer mittransformiert.

2.15.2 Funktionale Beschreibung

Summenkorrekturen

Pro D-Nummer können mehrere Summenkorrekturen (DL-Nummern) definiert werden. Damit lassen sich z.B. **werkstückortsabhängige** Korrekturwerte ermitteln und einer Schneide zuweisen. Summenkorrekturen wirken wie Verschleiß (d.h. additiv auf die Korrekturwerte der D-Nummer). Die Daten sind fest einer D-Nummer zugeordnet.

Einstellungen

Über Maschinendaten können Sie folgende Einstellungen vornehmen:

- Summenkorrektur aktivieren
- Anzahl der maximal im NC anzulegenden DL-Datensätze festlegen
- Anzahl der maximal einer D-Nummer zuzuordnenden DL-Nummern festlegen
- Festlegen, ob die Summenkorrekturen (fein / grob) bei Datensicherung mitgesichert werden sollen
- Festlegen, welche Summenkorrektur aktiviert werden soll, wenn:
  - eine neue Schneidenkorrektur aktiviert wird
  - ein Bedientafelfront-RESET erfolgt
  - ein Bedientafelfront-START erfolgt
  - das Programmende erreicht wurde

Dabei orientiert sich der Name an der Logik der entsprechenden Maschinendaten für Werkzeuge und Schneiden.

"Einrichtekorrektur" und "Summenkorrektur fein" können über Systemvariablen und entsprechende BTSS-Dienste gelesen und geschrieben werden.

Hinweis

Bei aktiver WZV kann über ein Maschinendatum festgelegt werden, ob die Summenkorrektur von einem Werkzeug, das während eines programmierten Werkzeugwechsels aktiviert wird, unverändert bleibt oder auf Null gesetzt werden soll.

Zusammenfassung der Korrekturparameter $TC_DPx

Bisher wurden folgende allgemeine Systemvariablen zur Beschreibung einer Schneide definiert:

$TC_DP1	Werkzeugtyp
$TC_DP2	Schneidenlage

Parameter für Geometrie und Verschleiß

Die Systemvariablen $TC_DP3 bis $TC_DP11 sind die Korrekturen der Werkzeuggeometrie zugeordnet. Die Systemvariablen $TC_DP12 bis $TC_DP20 erlauben es, für jeden dieser Parameter einen Verschleiß zu benennen:

Geometrie	Verschleiß	Korrekturen
Längenkorrekturen		
$TC_DP3	$TC_DP12	Länge 1
$TC_DP4	$TC_DP13	Länge 2
$TC_DP5	$TC_DP14	Länge 3
Radiuskorrektur		

Geometrie	Verschleiß	Korrekturen
$TC_DP6	$TC_DP15	Radius 1
$TC_DP7	$TC_DP16	Radius 2
Weitere Korrekturen		
$TC_DP8	$TC_DP17	Länge 4
$TC_DP9	$TC_DP18	Länge 5
$TC_DP10	$TC_DP19	Winkel 1
$TC_DP11	$TC_DP20	Winkel 2

Basismaß bzw. Adaptermaß

$TC_DP21	Adapter - Länge 1
$TC_DP22	Adapter - Länge 2
$TC_DP23	Adapter - Länge 3

Technologie

Systemvariable	Freiwinkel
$TC_DP24	Für ManualTurn wird hier der Freiwinkel abgelegt; WZ-Typ 5xx. Gleiche Bedeutung wie in Standardzyklen für Drehwerkzeuge. Für ShopMill wird hier der Spitzenwinkel des Bohrers abgelegt; WZ-Typ 2xx. Verwendung in Standardzyklen für Drehwerkzeuge; WZ-Typ 5xx. Bei diesen Werkzeugen ist das der Winkel an der Nebenschneide.
$TC_DP25	Für ManualTurn wird hier der Wert für die Schnittgeschwindigkeit abgelegt. Für ShopMill wird hier ein bitcodierter Wert für verschiedene Zustände von Werkzeugen des Typs 1xx und 2xx abgelegt.

Parameter der Summen und Einrichtekorrekturen ($TC_SCPxy-, $TC_ECPxy)

Die Nummerierung der Parameter orientiert sich an der Nummerierung der Systemvariablen $TC_DP3 bis $TC_DP11.

Die Wirkung der Parameter ist dem Verschleiß analog (additiv zur WZ-Geometrie). Pro Schneidenparameter sind maximal sechs Summen-/Einrichteparameter definierbar.

WZ-Geometrie-Parameter (auf den die Korrektur addiert wird)	Summen- / Einrichteparameter	WZ-Verschleiß-Parameter
Längenkorrekturen		
$TC_DP3	Länge 1 $TC_SCP**1**3, $TC_SCP**2**3,$TC_SCP**3**3, $TC_SCP**4**3,$TC_SCP**5**3,$TC_SCP**6**3 $TC_ECP**1**3, $TC_ECP**2**3,$TC_ECP**3**3, $TC_ECP**4**3,$TC_ECP**5**3,$TC_ECP**6**3 Die fettgedruckten Ziffern 1, 2, ... 6 bezeichnen die Parameter der maximal sechs (ortsabhängigen o.ä.) mit DL =1, ... 6 programmierbaren Korrekturen zum in Spalte eins genannten Parameter.	$TC_DP12
$TC_DP4	Länge 2 $TC_SCP**1**4, $TC_SCP**2**4,$TC_SCP**3**4, $TC_SCP**4**4,$TC_SCP**5**4,$TC_SCP**6**4 $TC_ECP**1**4, $TC_ECP**2**4,$TC_ECP**3**4, $TC_ECP**4**4,$TC_ECP**5**4,$TC_ECP**6**4	$TC_DP13
$TC_DP5	Länge 3 etc. ...	$TC_DP14
Radiuskorrektur		
$TC_DP6	Radius	$TC_DP15
$TC_DP7	Eckenradius	$TC_DP16
Weitere Korrekturen		
$TC_DP8	Länge 4	$TC_DP17
$TC_DP9	Länge 5	$TC_DP18
$TC_DP10	Winkel 1 ...etc.	$TC_DP19
$TC_DP11	Winkel 2 $TC_SCP**2**1, $TC_SCP**3**1,$TC_SCP**4**1, $TC_SCP**5**1,$TC_SCP**6**1,$TC_SCP**7**1 $TC_ECP**2**1, $TC_ECP**3**1,$TC_ECP**4**1, $TC_ECP**5**1,$TC_ECP**6**1,$TC_ECP**7**1 Die fettgedruckten Ziffern 2, 3, ... 7 bezeichnen die Parameter der maximal sechs (ortsabhängigen o.ä.) mit DL =1, ... 6 programmierbaren Korrekturen zum in Spalte eins genannten Parameter.	$TC_DP20

Randbedingungen

Die maximale Anzahl der DL-Datensätze einer Schneide und die Gesamtanzahl der Summenkorrekturen im NC werden über Maschinendaten festgelegt. Standardmäßig ist der Wert gleich Null, d.h. es können keine Summenkorrekturen programmiert werden.

Mit aktivierter "Überwachungsfunktion" ist es möglich, ein Werkzeug auf Verschleiß bzw. auch auf "Summenkorrektur" zu überwachen.

Die zusätzlichen Summen-/Einrichtedatensätze belegen zusätzlichen gepufferten Speicher. Pro Parameter werden 8 Byte benötigt.

Ein Summenkorrekturdatensatz benötigt: 8 Byte * 9 Parameter = 72 Bytes

Ein Einrichtekorrekturdatensatz benötigt ebensoviel. Zusätzlich werden einige Bytes an internen Verwaltungsdaten benötigt.

2.15.3 Aktivierung

Funktion

Die Funktion muss aktiviert werden über das Maschinendatum:

MD18108 $MN_MM_NUM_SUMCORR (Summenkorrekturen im TO-Bereich)

Über die Systemvariable $TC_ECPx und $TC_SCPx bzw. über die BTSS-Schnittstelle definierte Einrichte- und Summenkorrekturen ("fein") können im Teileprogramm aktiviert werden.

Dies erfolgt durch Programmierung des Sprachbefehls `DL=`"nr".

Mit dem Aktivieren einer neuen D-Nummer ist entweder eine neue DL-Nummer zu programmieren, oder es wird die über folgendes Maschinendatum festgelegte DL-Nummer wirksam:

MD20272 $MC_SUMCORR_DEFAULT (Grundstellung Summenkorrektur ohne Programm)

DL-Programmierung

Die Programmierung der Summenkorrektur ist immer relativ zur aktiven D-Nummer und erfolgt mit dem Befehl:

`DL =` "n"

Die Summenkorrektur "n" wird zum Verschleiß der aktiven D-Nummer addiert.

Hinweis

Wird mit "Einrichtekorrektur" **und** "Summenkorrektur fein" gearbeitet, werden beide Korrekturen zusammengefasst und dann zum Verschleiß des Werkzeuges addiert.

Die Abwahl der Summenkorrektur erfolgt mit dem Befehl:

```
DL = 0
```

Hinweis

DL0 ist nicht erlaubt. Mit der Abwahl der Korrektur (`D0` und `T0`) wird die Summenkorrektur ebenfalls unwirksam.

Programmieren einer nicht vorhandenen Summenkorrektur löst einen Alarm aus, analog der Programmierung einer nicht vorhandenen D-Korrektur.

Damit ist nur noch der definierte Verschleiß Bestandteil der Korrektur (in den Systemvariablen $TC_DP12 bis $TC_DP20 definiert).

Wird bei aktiver D-Korrektur eine Summenkorrektur programmiert (auch bei Abwahl), wirkt sich das auf die Bahn genauso aus wie die Programmierung von D. Damit verliert z.B. eine gerade aktive Radiuskorrektur ihren Bezug zu Nachbarsätzen.

Konfiguration

MD18112 $MN_MM_KIND_OF_SUMCORR, Bit 4=0: (Eigenschaften der Summenkorrekturen im TO-Bereich) Standardeinstellung:

Es ist nur **ein** Datensatz Summenkorrekturen pro DL-Nummer vorhanden.

Es wird allgemein von Summenkorrektur gesprochen.

Es sind damit die durch $TC_SCPx dargestellten Daten gemeint.

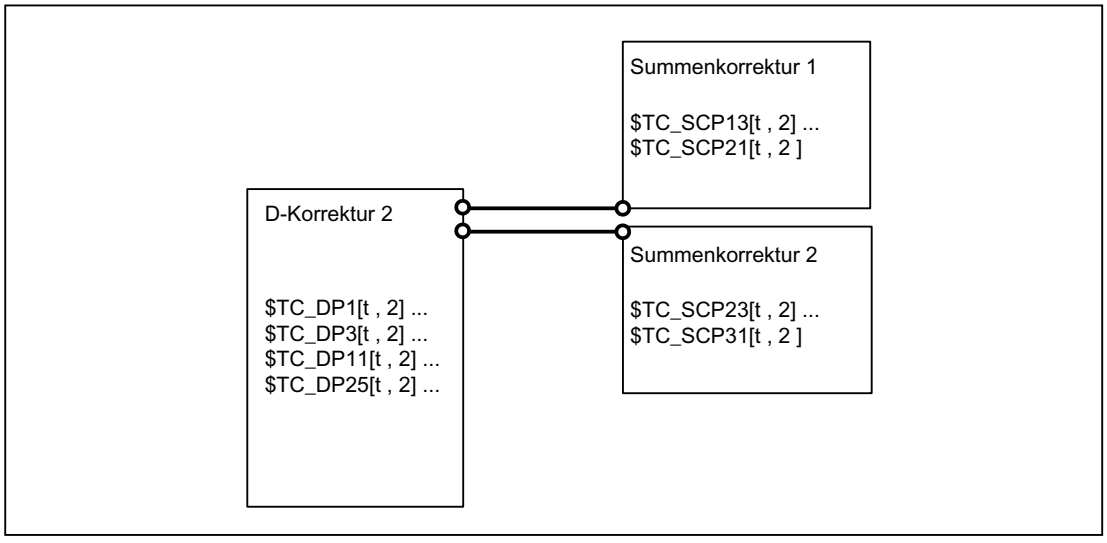


Bild 2-47 MD18112 $MN_MM_KIND_OF_SUMCORR, Bit 4 = 0

Das Werkzeug mit T = t ist aktiv. Mit den Daten im Bild wird programmiert:

D2	;	Schneiden Korrekturen; d.h. $TC_DP3,... $TC_DP11 + Verschleiß ($TC_DP12,...$TC_DP20) + Adaptermaß
...		
DL=1	;	zusätzlich zu den bisherigen Korrekturen von D2 wird die Summenkorrektur 1 addiert d.h. $TC_SCP13,...$TC_SCP21

...		
DL=2	;	zur Korrektur D2 wird nicht mehr Summenkorrektur 1 addiert, sondern Summenkorrektur 2 d.h. $TC_SCP23,...$TC_SCP31
...		
DL=0	;	Abwahl der Summenkorrektur; nur noch die Daten von D2 sind wirksam

MD18112 $MN_MM_KIND_OF_SUMCORR, Bit 4=1: Einrichtekorrekturen stehen zur Verfügung

Die Summenkorrektur setzt sich nun zusammen aus der "Summenkorrektur fein" (dargestellt durch $TC_SCPx) und der Einrichtekorrektur (dargestellt durch $TC_ECPx). Damit gibt es zu einer DL-Nummer zwei Datensätze. Die Summenkorrektur ergibt sich durch Addition der entsprechenden Komponenten ($TC_ECPx + $TC_SCPx).

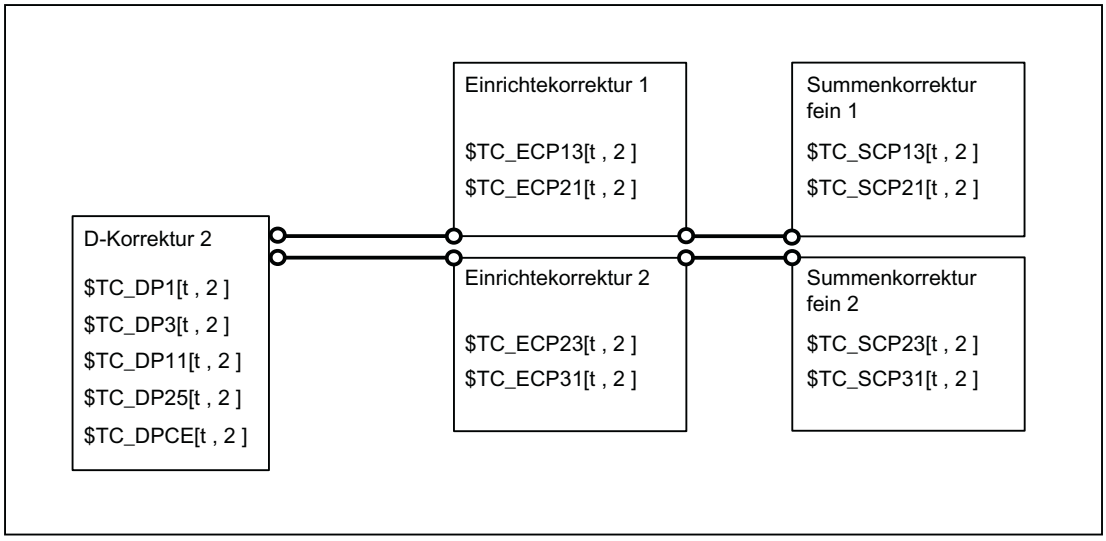


Bild 2-48 MD18112 $MN_MM_KIND_OF_SUMCORR, Bit 4 = 1 "Einrichtekorrekturen" + "Summenkorrekturen fein"

Das Werkzeug mit T = t ist aktiv. Mit den Daten im Bild wird programmiert:

D2	;	Schneiden Korrekturen; d.h. $TC_DP3,... $TC_DP11 + Verschleiß ($TC_DP12,...$TC_DP20) + Adaptermaß
...		
DL=1	;	zusätzlich zu den bisherigen Korrekturen von D2 wird die Summenkorrektur 1 addiert; d.h. $TC_ECP13 + $TC_SCP13,...$TC_ECP21 + $TC_SCP21
...		
DL=2	;	zur Korrektur D2 wird nicht mehr Summenkorrektur 1 addiert, sondern Summenkorrektur 2; d.h. $TC_ECP23 + $TC_SCP23,...$TC_ECP31 + $TC_SCP31
...		
DL=0	;	Abwahl der Summenkorrektur. Nur noch die Daten von D2 sind wirksam.

Lesen / Schreiben im Teileprogramm

Die einzelnen Summenkorrekturparametersätze werden durch Nummernbereiche der Systemvariablen $TC_SCP unterschieden.

Die Bedeutung der einzelnen Variablen ist analog den Geometrie-Variablen $TC_DP3 bis $TC_DP11. Für die Grundfunktionalität sind nur Länge 1, Länge 2, Länge 3 gesetzt (Variablen $TC_SCP13 - $TC_SCP15 für die erste Summenkorrektur der Schneide).

R5 = $TC_SCP13[t, d]	;	Setzt den Wert des R-Parameters auf den Wert der ersten Komponente der Summenkorrektur 1 der Schneide (d) des Werkzeuges (t).
R6 = $TC_SCP21[t, d]	;	Setzt den Wert des R-Parameters auf den Wert der letzten Komponente der Summenkorrektur 1 der Schneide (d) des Werkzeuges (t).
R50 = $TC_SCP23[t, d]	;	Setzt den Wert des R-Parameters auf den Wert der ersten Komponente der Summenkorrektur 2 der Schneide (d) des Werkzeuges (t).
$TC_SCP43[t, d] = 1.234	;	Setzt den Wert der ersten Komponente der Summenkorrektur 4 der Schneide (d) des Werkzeuges (t) auf den Wert 1.234.

Die Aussagen für die Einrichtekorrekturen (falls NCK damit konfiguriert ist) sind analog, d.h.:

R5 = $TC_ECP13[t, d]	;	Setzt den Wert des R-Parameters auf den Wert der ersten Komponente der Einrichtekorrektur 1 der Schneide (d) des Werkzeuges (t).
R6 = $TC_ECP21[t, d]	;	Setzt den Wert des R-Parameters auf den Wert der letzten Komponente der Einrichtekorrektur 1 der Schneide (d) des Werkzeuges (t).
etc.		

Beim Arbeiten mit Einrichtekorrekturen werden die "Summenkorrekturen fein" mit den Systemvariablen $TC_SCPx beschrieben.

Neuanlegen - Summenkorrektur

Falls der Korrekturdatensatz (x) noch nicht existiert, wird er mit der ersten Schreiboperation auf einen seiner Parameter (y) angelegt.

$TC_SCPxy[t, d] = r.r	;	Der Parameter y der Summenkorrektur x erhält den Wert "r.r." Die anderen Parameter von x haben den Wert Null.

Beim Arbeiten mit Einrichtekorrekturen werden die "Summenkorrekturen fein" mit den Systemvariablen $TC_SCPx beschrieben.

Hinweis

Beim Arbeiten mit Einrichtekorrekturen wird mit dem Anlegen eines Datensatzes für "Summenkorrektur fein" der zugehörige Datensatz für die Einrichtekorrektur mit angelegt, falls bis dahin zu [t, d] noch kein Datensatz existierte.

Neuanlegen - Einrichtekorrektur

Falls der Korrekturdatensatz (x) noch nicht existiert, wird er mit der ersten Schreiboperation auf einen seiner Parameter (y) angelegt.

$TC_ECPxy[t, d] = r.r	;	Der Parameter y der Einrichtekorrektur x erhält den Wert "r.r". Die anderen Parameter von x haben den Wert Null.

Hinweis

Beim Arbeiten mit Einrichtekorrekturen wird mit dem Anlegen eines Datensatzes für Einrichtekorrekturen der zugehörige Datensatz für "Summenkorrektur fein" mit angelegt, falls bis dahin zu [t, d] noch kein Datensatz existierte.

DELDL - Löschen Summenkorrektur

Summenkorrekturen sind meist nur während der Bearbeitung mit einer Schneide zu einem bestimmten Zeitpunkt an einem bestimmten Ort des Werkstückes von Bedeutung. Mit dem NC-Sprachbefehl `DELDL` können die Summenkorrekturen aus Schneiden gelöscht werden (Freigeben von benötigtem Speicher).

status = DELDL(t, d)	;	löscht alle Summenkorrekturen der Schneide d des Werkzeuges t
	;	t, d sind optionale Parameter

Ist d nicht angegeben, werden alle Summenkorrekturen aller Schneiden des Werkzeuges t gelöscht.

Sind d und t nicht angegeben, werden alle Summenkorrekturen der Schneiden aller Werkzeuge der TO-Einheit gelöscht (für den Kanal, in dem der Befehl programmiert wird).

Falls mit Einrichtekorrekturen gearbeitet wird, werden mit dem `DELDL`-Befehl sowohl die Einrichtekorrektur, als auch die "Summenkorrektur fein" der genannten Schneide(n) gelöscht.

Hinweis

Beim "Löschen" wird der Speicher für die Datensätze freigegeben.

Anschließend können die gelöschten Summenkorrekturen nicht mehr aktiviert oder programmiert werden.

Summenkorrekturen, Einrichtekorrekturen aktiver Werkzeuge können nicht gelöscht werden (analog zum Löschverhalten von D-Korrekturen bzw. den Werkzeugdaten).

Der Rückgabewert "status" zeigt das Ergebnis des Löschbefehls an:

0:	Löschen erfolgreich durchgeführt
-1:	Löschen nicht (eine Schneide) oder nicht vollständig (mehrere Schneiden) durchgeführt

Datensicherung

Die Daten werden im Rahmen der allgemeinen Werkzeugdatensicherung (als Bestandteil der D-Nummerndatensätze) gesichert.

Eine Datensicherung der Summenkorrekturen ist sinnvoll, da es möglich sein muss, bei akuten Problemen den momentanen Zustand zu sichern. Über Maschinendaten können die Summenkorrekturen von der Datensicherung ausgeschlossen werden (für "Einrichtekorrekturen" und "Summenkorrekturen fein" getrennt einstellbar).

Hinweis

Summenkorrekturen verhalten sich bzgl. Satzsuchlauf und Repos analog zu D-Korrekturen. Das Verhalten bei Reset und PowerOn kann durch Maschinendaten festgelegt werden.

Falls über das folgende Maschinedatum eingestellt sein, dass nach PowerOn die letzte aktive WZ-Korrekturnummer (D) aktiviert werden soll, ist die zuletzt aktive DL-Nummer jedoch nicht mehr aktiv:

MD20110 $MC_RESET_MODE_MASK (Festlegung der Steuerungs-Grundstellung nach Reset / TP-Ende)

2.15.4 Beispiele

Beispiel 1

Dass keine Korrektur und keine Summenkorrektur wirksam werden, soll beim Werkzeugwechsel festgelegt werden durch die Maschinendaten:

- MD20270 $MC_CUTTING_EDGE_DEFAULT=0 (Grundstellung der Werkzeugschneide ohne Programmierung)
- MD20272 $MC_SUMCORR_DEFAULT=0 (Grundstellung Summenkorrektur ohne Programm)

```
T5 M06              ; WZ mit Nummer 5 wird eingewechselt - keine Korrektur aktiv
D1 DL=3             ; Korrektur D1 + Summenkorrektur 3 von D1 werden aktiv
X10
DL=2                ; Korrektur D1 + Summenkorrektur 2 werden aktiv
X20
DL=0                ; Summenkorrekturabwahl, nur noch Korrektur D1 ist aktiv
```

```
D2                ; Korrektur D2 wird aktiv - kein Beitrag der Summenkorrektur zur
                    Korrektur
X1
DL=1              ; Korrektur D2 + Summenkorrektur 1 werden aktiv
X2
D0                ; Korrekturabwahl
X3
DL=2              ; ohne Auswirkung - DL2 von D0 ist Null (analog zur Programmie-
                    rung T0 D2)
```

Beispiel 2

Beim Werkzeugwechsel soll festgelegt werden, dass die Korrektur `D2` und die Summenkorrektur `DL=1` wirksam werden, durch die Maschinendaten:

MD20270 $MC_CUTTING_EDGE_DEFAULT=2 (Grundstellung der Werkzeugschneide ohne Programmierung)

MD20272 $MC_SUMCORR_DEFAULT=1 (Grundstellung Summenkorrektur ohne Programm)

```
T5 M06            ; WZ mit Nummer 5 wird eingewechselt - D2 + DL=1 sind aktiv (=Wer-
                    te der Maschinendaten)
D1 DL=3           ; Korrektur D1 + Summenkorrektur 3 von D1 werden aktiv
X10
DL=2              ; Korrektur D1 + Summenkorrektur 2 werden aktiv
X20
DL=0              ; Summenkorrekturabwahl, nur noch Korrektur D1 ist aktiv
D2                ; Korrektur D2 wird aktiv - Summenkorrektur DL=1 wird aktiviert
X1
DL=2              ; Korrektur D2 + Summenkorrektur 2 werden aktiv
D1                ; Korrektur D1 + Summenkorrektur 1 werden aktiv
```

2.15.5 Erweiterungen der Werkzeuglängenbestimmung

2.15.5.1 Korrekturen einsatzort- und werkstückspezifisch einrechnen

Zusammensetzung der wirksamen Werkzeuglänge

Bei einer Werkzeugkorrektur ohne aktive kinematische Transformation setzt sich die wirksame Werkzeuglänge aus maximal 8 Vektoren zusammen:

- Werkzeuglänge (Geometrie) ($TC_DP3 - $TC_DP5)
- Verschleiß ($TC_DP12 - $TC_DP14)
- Basismaß (siehe Hinweis) ($TC_DP21 - $TC_DP23)
- Adaptermaß (siehe Hinweis) ($TC_ADPT1 - $TC_ADPT3)

- Summenkorrekturen fein ($TC_SCPx3 - $TC_SCPx5)
- Summenkorrekturen grob bzw. Einrichtkorrekturen ($TC_ECPx3 - $TC_ECPx5)
- Offsetvektor l_1 des orientierbaren Werkzeugträgers ($TC_CARR1 - $TC_CARR3)
- Offsetvektor l_2 des orientierbaren Werkzeugträgers ($TC_CARR4 - $TC_CARR6)
- Offsetvektor l_3 des orientierbaren Werkzeugträgers ($TC_CARR15 - $TC_CARR17)

Hinweis

Basismaß und Adaptermaß können nur alternativ wirksam sein.

Wirkungsweise der einzelnen Vektoren

Die Wirkungsweise der einzelnen Vektoren oder Gruppen von Vektoren hängt dabei von folgenden weiteren Größen ab:

Einflussgröße	Wirkungsweise
G-Befehle	aktive Bearbeitungsebene
Werkzeugtyp	Fräswerkzeuge bzw. Dreh- oder Schleifwerkzeuge
Maschinendaten	Werkzeugverwaltung aktiv / nicht aktiv, orientierbarer WZ-Träger vorhanden / nicht vorh.
Settingdaten	Verhalten von Werkzeuglängenkomponenten beim Spiegeln oder bei Ebenenwechsel
Orientierbare Werkzeugträger	Einstellwerte des orientierbaren Werkzeugträgers
Adaptertransformationen	transformierte Werkzeug-Korrekturwerte

Verteilung auf die Geometrieachskomponenten

Wie die drei Vektorkomponenten von Teilsummen der beteiligten Vektoren auf die drei Geometrieachskomponenten verteilt werden, bestimmen die folgenden Größen:

Einflussgröße	Abhängigkeiten
aktive Bearbeitungsebene: `G17` X/Y-Richtung `G18` Z/X-Richtung `G19` Y/Z-Richtung	Zustellebene: Z Y X
Werkzeugtyp: Fräswerkzeuge, Bohrwerkzeuge, Schleifwerkzeuge, Drehwerkzeuge	Siehe Kapitel "WZ-Parameter 1: Werkzeugtyp (Seite 39)", Tabelle "Minimal notwendige WZ-Parameter"
SD42900 $SC_MIRROR_TOOL_LENGTH SD42910 $SC_MIRROR_TOOL_WEAR SD42920 $SC_WEAR_SIGN_CUTPOS SD42930 $SC_WEAR_SIGN SD42940 $SC_TOOL_LENGTH_CONST SD42950 $SC_TOOL_LENGTH_TYPE	Siehe Kapitel "Werkzeugkorrektur-Sonderbehandlungen (Seite 185)" und Kapitel "Settingdaten (Seite 243)".
Adaptertransformationen	**Weitere Informationen:** Funktionshandbuch Werkzeugverwaltung

Die resultierende Werkzeugorientierung bleibt dabei immer parallel zu einer der drei Achsrichtungen X, Y oder Z und hängt ausschließlich von der aktiven Bearbeitungsebene `G17-G19` ab, da dem Werkzeug bisher keine Orientierung zugeordnet werden konnte.

Stufenlose Veränderung der Werkzeugorientierung

Der orientierbare Werkzeugträger eröffnet zusätzlich zu weiteren Verschiebungen oder Längenänderungen mit Hilfe der Offsetvektoren l_1 - l_3 die Möglichkeit, die Werkzeugorientierung stufenlos zu verändern.

Für weitere Erläuterungen siehe Kapitel "Orientierbare Werkzeugträger (Seite 94)".

Kleinere Bediener-Korrekturen

Kleinere Korrekturen müssen hingegen auch im normalen Produktionsbetrieb verändert werden können.

Gründe dafür sind z. B.:

- Werkzeugverschleiß
- Aufspannfehler
- Temperaturgang der Maschine

Diese Korrekturen werden dabei wie folgt definiert:

Definition	Verschleißkomponenten
Verschleiß	$TC_DP12 - $TC_DP14,
Summenkorrekturen fein	$TC_SCPx3 - $TC_SCPx5,
Summenkorrekturen grob bzw. Einrichtkorrekturen	$TC_ECPx3 - $TC_ECPx5

Insbesondere sind Korrekturen, welche die Werkzeuglängenberechnung betreffen, in den Koordinaten einzutragen, in denen sie auch gemessen wurden.

Diese werkstückspezifischen Korrekturen können einfacher vorgenommen werden, mit Hilfe der G-Gruppe 56, mit den drei Werten `TOWSTD`, `TOWMCS` und `TOWWCS`, und des Settingdatums:

SD42935 $SC_WEAR_TRANSFORM (Transformation der Werkzeugkomponenten)

SD42935

Welche der Verschleißkomponenten:

- Verschleiß ($TC_DP12 - $TC_DP14):
- Einrichtkorrekturen bzw. Summenkorrekturen grob ($TC_ECPx3 - $TC_ECPx5)
- Summenkorrekturen fein ($TC_SCPx3 - $TC_SCPx5)

bei den Transformationen:

- Adaptertransformation
- Orientierbarer Werkzeugträger

transformiert bzw. nicht transformiert werden sollen, kann festgelegt werden über das Settingdatum:

SD42935 $SC_WEAR_TRANSFORM (Transformation der Verschleißwerte)

In der Grundstellung des Settingdatums werden alle Verschleißwerte transformiert.

Das Settingdatum wird bei folgenden Funktionen berücksichtigt:

- Verschleißwerte im Maschinenkoordinatensystem
  Teileprogrammanweisung: `TOWMCS`
- Verschleißwerte im Werkstückkoordinatensystem
  Teileprogrammanweisung: `TOWWCS`

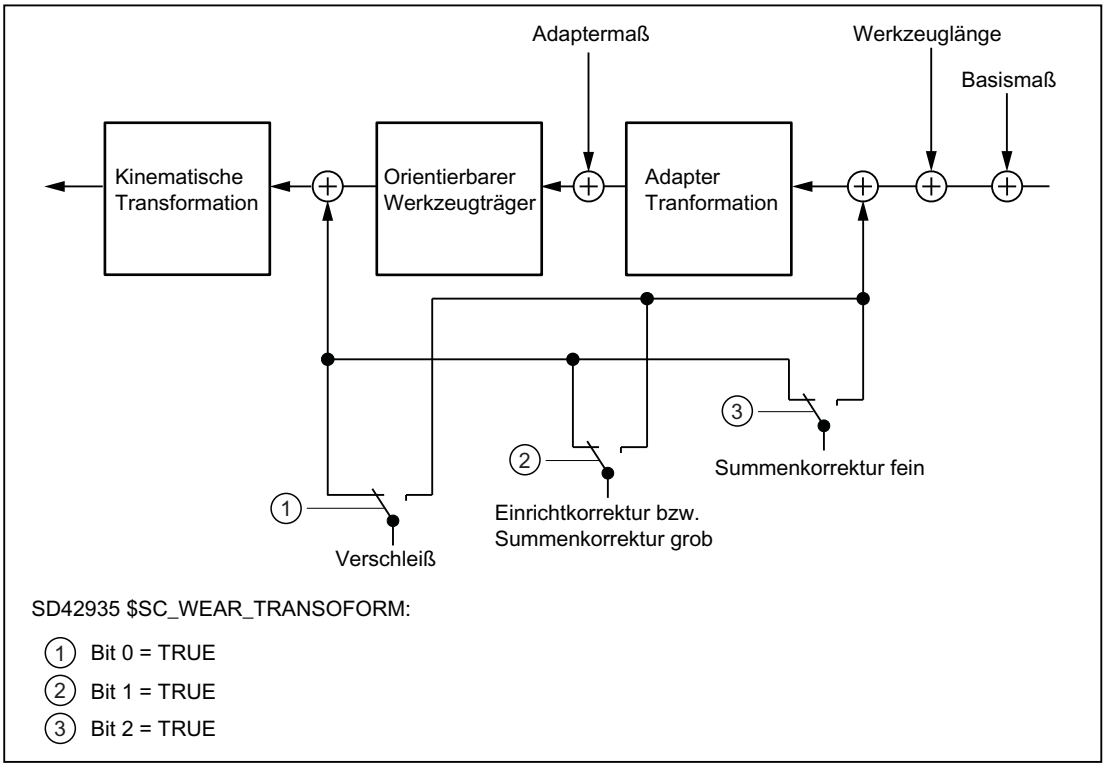


Bild 2-49 Transformation der Verschleißdaten abhängig von SD42935

Programmierung

Mit der G-Gruppe 56 können folgende Werte festgelegt werden:

Syntax	Korrekturen
`TOWSTD`	Grundstellungswert für Korrekturen in der Werkzeuglänge
`TOWMCS`	Verschleißwerte im Maschinenkoordinatensystem (MKS)
`TOWWCS`	Verschleißwerte im Werkstückkoordinatensystem (WKS)
`TOWBCS`	Verschleißwerte im Basiskoordinatensystem (BKS)
`TOWTCS`	Verschleißwerte im Werkzeugkoordinatensystem (TCS) an der Werkzeughalteraufnahme (T Werkzeugträgerbezugspunkt)
`TOWKCS`	Verschleißwerte im Werkzeugkoordinatensystem der kinematischen Transformation (KCS) des Werkzeugkopfes

Koordinatensysteme für Offsets der Werkzeuglänge

Mit den G-Befehlen `TOWMCS`, `TOWWCS`, `TOWBCS`, `TOWTCS` und `TOWKCS` kann z. B. die Werkzeuglängenkomponente Verschleiß in fünf verschiedenen Koordinatensystemen gemessen werden.

1. Maschinenkoordinatensystem MKS
1. Basiskoordinatensystem BKS
1. Werkstückkoordinatensystem WKS
1. Werkzeugkoordinatensystem der kinematischen Transformation KCS
1. Werkzeugkoordinatensystem TCS

Die berechnete Werkzeuglänge oder auch eine Werkzeuglängenkomponente kann mit der vordefinierten Funktion GETTCOR (Seite 218) in einem dieser Koordinatensysteme dargestellt und ausgelesen werden.

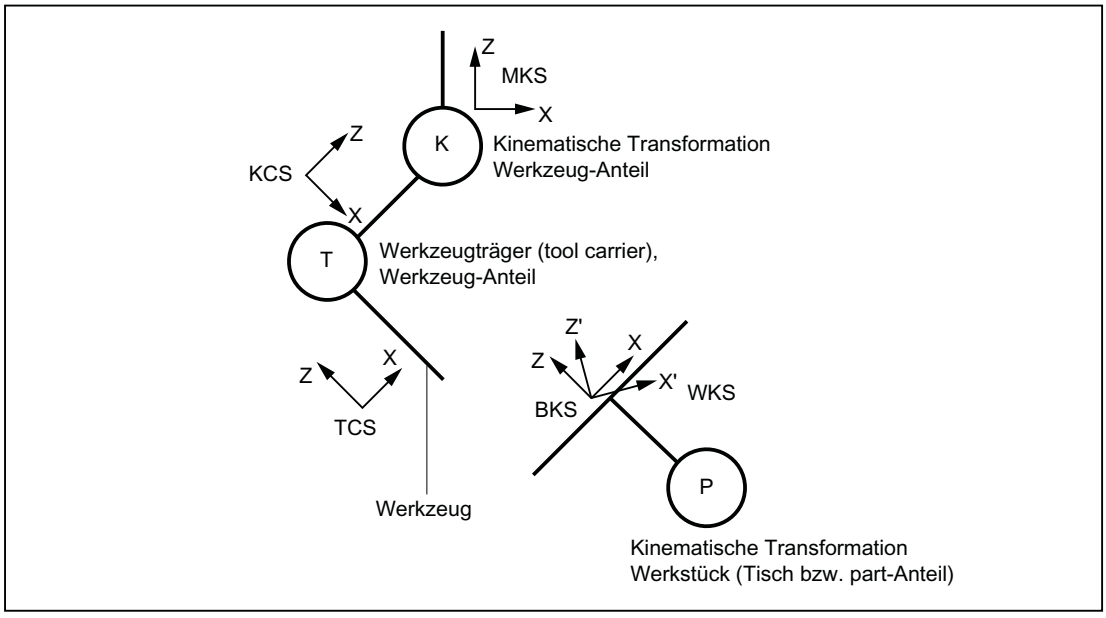


Bild 2-50 Koordinatensystem bei der Bewertung von Werkzeuglängen

2.15.5.2 Funktionalität der einzelnen Verschleißwerte

TOWSTD

Grundstellungswert (Standardbehandlung):

- Die Verschleißwerte werden zu den übrigen Werkzeuglängenkomponenten addiert. Die daraus resultierende Gesamtwerkzeuglänge geht dann gegebenenfalls in weitere Berechnungen ein.

Für den Fall eines aktiven orientierbaren Werkzeugträgers:

- Die Verschleißwerte werden der entsprechenden Drehung unterworfen.

TOWMCS

Verschleißwerte im Maschinenkoordinatensystem (MKS):

Für den Fall einer aktiven Drehung durch einen orientierbaren Werkzeugträger:

- Der Werkzeugträger dreht nur den Vektor der resultierenden Werkzeuglänge ohne Berücksichtigung des Verschleißes.
  Anschließend werden der so gedrehte Werkzeuglängenvektor und der Verschleiß addiert.
  Der Verschleiß wird der Drehung nicht unterworfen.

Wenn **kein** orientierbarer Werkzeugträger aktiv ist oder dieser keine Drehung bewirkt, sind TOWMCS und TOWSTD identisch.

Lineare Transformation

Die Werkzeuglänge im MKS ist nur dann eindeutig definiert, wenn das MKS aus dem BKS durch eine lineare Transformation hervorgeht.

Dies würde auch unter einer der folgenden Bedingungen der Fall sein:

- Keine kinematische Transformation aktiv.
- Orientierungstransformationen (3-, 4- und 5-Achs-Transformationen) wirksam.

TOWWCS

Verschleißwerte im Werkstückkoordinatensystem (WKS):

- Bei einem aktiven orientierbaren Werkzeugträger wird der Werkzeugvektor ohne Berücksichtigung des Verschleißes wie bei TOWMCS berechnet.
- Die Angaben des Verschleißes werden im Werkstückkoordinatensystem interpretiert.

Der Verschleißvektor im Werkstückkoordinatensystem wird auf das Maschinenkoordinatensystem umgerechnet und zum Werkzeugvektor addiert.

TOWBCS

Verschleißwerte im Basiskoordinatensystem (BKS):

- Bei einem aktiven orientierbaren Werkzeugträger wird der Werkzeugvektor ohne Berücksichtigung des Verschleißes wie bei TOWMCS berechnet.
- Die Angaben des Verschleißes werden im Werkstückkoordinatensystem interpretiert.

Der Verschleißvektor im Basiskoordinatensystem wird auf das Werkstückkoordinatensystem umgerechnet und zum Werkzeugvektor addiert.

Nichtlineare Transformation

Ist eine nichtlineare Transformation beispielsweise mit TRANSMIT aktiv und wird das MKS als gewünschtem Koordinatensystem angegeben, dann wird an Stelle des MKS automatisch das BKS verwendet.

Orientierbarer Werkzeugträger

Ein eventuell vorhandener Tischanteil des orientierbaren Werkzeugträgers geht im Gegensatz zu einem Tisch- (bzw. part-) Anteil der kinematischen Transformation nicht unmittelbar in die Koordinatensysteme ein. Eine durch solch einen Anteil beschriebene Drehung wird in einem

Basis-Frame oder Systemframe abgebildet und geht damit in den Übergang vom WKS in das BKS mit ein.

Kinematische Transformation

Der Tisch- (bzw. part-) der kinematischen Transformation wird durch den Übergang vom BKS in das MKS beschrieben.

TOWTCS

Verschleißwerte im Werkzeugkoordinatensystem (TCS):

- Bei einem aktiven orientierbaren Werkzeugträger wird der Werkzeugvektor ohne Berücksichtigung des Verschleißes wie bei TOWMCS berechnet.
- Die Angaben des Verschleißes werden im Werkzeugkoordinatensystem TCS interpretiert.

Der Verschleißvektor im Werkzeugkoordinatensystem (TCS) wird über das Werkzeugkoordinatensystem der kinematischen Transformation (KCS) auf das Maschinenkoordinatensystem umgerechnet und zum Werkzeugvektor addiert.

TOWKCS

Die Angaben der Verschleißwerte der kinematischen Transformation werden im dazugehörigen Werkzeugkoordinatensystem (KCS) interpretiert.

Der Verschleißvektor wird über das Werkzeugkoordinatensystem der kinematischen Transformation auf das Maschinenkoordinatensystem umgerechnet und zum Werkzeugvektor addiert.

G-Befehl Wechsel bei aktivem Werkzeug

Ein Wechsel des G-Befehls der Gruppe TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS und TOWKCS beeinflusst ein bereits aktives Werkzeug nicht und wird erst bei der nächsten Anwahl eines Werkzeugs wirksam.

Ein neuer G-Befehl dieser Gruppe wird auch dann wirksam, wenn er im gleichen Satz programmiert wird, in dem auch ein Werkzeug angewählt wird.

Bewertung einzelner Verschleißkomponenten

Die Bewertung der einzelnen Verschleißkomponenten (Zuordnung zu den Geometrieachsen, Vorzeichenbewertung) wird beeinflusst durch:

- Die aktive Ebene
- Die Adaptertransformation
- Die in der folgenden Tabelle dargestellten fünf Settingdaten

Settingdatum	Verschleißkomponenten		
SD42910 $SC_MIRROR_TOOL_WEAR	TOWSTD	TOWMCS	TOWWCS
SD42920 $SC_WEAR_SIGN_CUTPOS	X	X	—
SD42930 $SC_WEAR_SIGN	X	—	—

Settingdatum	Verschleißkomponenten		
SD42940 $SC_TOOL_LENGTH_CONST	X	X	X
SD42950 $SC_TOOL_LENGTH_TYPE	X	X	X

Hinweis

Verschleißkomponenten, welche einer aktiven Drehung durch eine Adaptertransformation oder einen orientierbaren Werkzeugträger unterworfen werden, werden als nicht-transformierte Verschleißkomponenten bezeichnet.

Besonderheiten

Ist TOWMCS oder TOWWCS aktiv, wirkt folgendes Settingdatum nicht auf die nicht-transformierten Verschleißkomponenten:

SD42920 $SC_WEAR_SIGN_CUTPOS (Vorzeichen des Verschleißes bei Werkzeugen mit Schneideanlagen)

Nicht auf die nicht-transformierten Verschleißkomponenten wirkt bei TOWWCS zusätzlich auch das Settingdatum:

SD42910 $SC_MIRROR_TOOL_WEAR (Vorzeichenwechsel Werkzeugverschleiß beim Spiegeln)

In diesem Fall ist eine eventuell aktive Spiegelung in dem Frame, der zur Bewertung der Verschleißkomponenten herangezogen wird, bereits enthalten.

Bei einem Ebenenwechsel bleibt die Zuordnung der nicht-transformierten Verschleißkomponenten zu den Geometrieachsen erhalten, d. h., diese werden nicht wie bei übrigen Längenkomponenten getauscht. Die Zuordnung der Komponenten hängt von der aktiven Ebene bei der Werkzeuganwahl ab.

Beispiel

Es sei ein Fräswerkzeug gegeben, bei dem nur der der Länge L1 zugeordnete Verschleißwert $TC_DP12 ungleich Null ist.

Ist G17 aktiv, wirkt diese Länge in Richtung der z-Achse.

Dieses Maß wirkt auch bei einem Ebenenwechsel nach der Werkzeuganwahl stets in Z-Richtung, wenn `TOWMCS` oder `TOWWCS` aktiv und das Bit 1 gesetzt ist im Settingdatum:

SD42935 $SC_WEAR_TRANSFORM (Transformationen für Werkzeugkomponenten)

Ist bei der Werkzeuganwahl z. B. `G18` wirksam, wirkt die Komponente stattdessen unter sonst gleichen Bedingungen stets in y-Richtung.

2.16 Mit Werkzeugumgebungen arbeiten

Übersicht Funktionen

- Werkzeugumgebung speichern (TOOLENV) (Seite 213)
- Werkzeugumgebung löschen (DELTOOLENV) (Seite 216)
- T-, D- und DL-Nummer lesen (GETTENV) (Seite 217)
- Werkzeuglängen bzw. -längenkomponenten lesen (GETTCOR) (Seite 218)
- Werkzeugkomponenten ändern (SETTCOR) (Seite 225)

Übersicht Systemvariablen

- Informationen zu gespeicherten Werkzeugumgebungen lesen ($P_TOOLENVN, ($P_TOOLENV) (Seite 218)

2.16.1 Werkzeugumgebung speichern (TOOLENV)

Die Funktion TOOLENV dient dazu, alle aktuellen Zustände zu speichern, die für die Bewertung der im Speicher abgelegten Werkzeugdaten von Bedeutung sind.

Das sind im Einzelnen die folgenden Daten:

- Der aktive G-Befehl der Gruppe:
  - 6 (G17, G18, G19)
  - 56 (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)
- Die aktive Planachse
- Die Maschinendaten:
  - MD18112 $MN_MM_KIND_OF_SUMCORR (Eigenschaften der Summenkorrekturen im TO-Bereich)
  - MD20360 $MC_TOOL_PARAMETER_DEF_MASK (Definition der WZ-Parameter)

- Die Settingdaten:
  - SD42900 $SC_MIRROR_TOOL_LENGTH (Vorzeichenwechsel Werkzeuglänge beim Spiegeln)
  - SD42910 $SC_MIRROR_TOOL_WEAR (Vorzeichenwechsel Werkzeugverschleiß beim Spiegeln)
  - SD42920 $SC_WEAR_SIGN_CUTPOS (Vorzeichen des Verschleißes bei Werkzeugen mit Schneidenanlagen)
  - SD42930 $SC_WEAR_SIGN (Vorzeichen des Verschleißes)
  - SD42935 $SC_WEAR_TRANSFORM (Transformationen für Werkzeugkomponenten)
  - SD42940 $SC_LENGTH_CONST (Wechsel der Werkzeuglängenkomponenten bei Ebenenwechsel)
  - SD42942 $SC_TOOL_LENGTH_CONST_T (Wechsel der Werkzeuglängenkomponenten für Drehwerkzeuge bei Ebenenwechsel)
  - SD42950 $SC_TOOL_LENGTH_TYPE (Zuordnung der Werkzeuglängenkomponenten unabhängig vom Werkzeugtyp)
  - SD42954 $SC_TOOL_ORI_CONST_M (Wechsel der Werkzeugorientierungskomponenten für Fräswerkzeuge bei Ebenenwechsel)
  - SD42956 $SC_TOOL_ORI_CONST_T (Wechsel der Werkzeugorientierungskomponenten für Drehwerkzeuge bei Ebenenwechsel)
- Der Orientierungsanteil des aktuellen Gesamtframes (Drehungen und Spiegelungen, keine Nullpunktverschiebungen oder Skalierungen)
- Der Orientierungsanteil und die resultierende Länge des aktiven orientierbaren Werkzeugträgers
- Der Orientierungsanteil und die resultierende Länge einer aktiven Transformation

Außer den genannten Daten, die die Umgebung des Werkzeugs beschreiben, werden auch T-Nummer, D-Nummer und DL-Nummer des aktiven Werkzeugs mit abgelegt, damit später auf dieses Werkzeug in der gleichen Umgebung wie beim Aufruf von TOOLENV zugegriffen werden kann, ohne das Werkzeug erneut bezeichnen zu müssen.

Syntax

```
<Status> = TOOLENV(<Name>)
```

Bedeutung

TOOLENV(...):	Vordefinierte Funktion zum Speichern einer Werkzeugumgebung	
	Alleine im Satz:	ja

<Status>:	Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.		
	Datentyp:		INT
	Wert:	0	Funktion OK
		-1	Kein Speicherplatz für Werkzeugumgebungen reserviert: MD18116 $MN_MM_NUM_TOOL_ENV = 0 D. h., die Funktionalität "Werkzeugumgebungen" ist nicht vorhanden.
		-2	Keine freien Speicherplätze für Werkzeugumgebungen mehr vorhanden.
		-3	Nullstring als Name einer Werkzeugumgebung ist nicht zulässig.
		-4	Kein Parameter (<Name>) angegeben.
Parameter			
1 <Name>:	Name, unter dem der aktuelle Datensatz abgelegt werden soll Ist ein Datensatz mit dem gleichen Namen bereits vorhanden, wird dieser überschrieben. Der Status ist in diesem Fall "0".		
	Datentyp:		STRING

Weitere Informationen

Basismaß/Adaptermaß - Werkzeuglängenkorrektur

Ob die Adapterlänge oder das Basismaß (schneidenspezifische Parameter $TC_DP21, $TC_DP22 und $TC_DP23) in die Berechnung der Werkzeuglänge eingeht, entscheidet bei aktiver Werkzeug-Magazinverwaltung (nur verfügbar mit der Option "Werkzeugverwaltung"!) der Wert des Maschinendatums:

MD18104 $MN_MM_NUM_TOOL_ADAPTER (WZ-Adapter im TO-Bereich)

Da eine Veränderung dieses Maschinendatums nur beim Hochlauf der Steuerung wirksam werden kann, wird es nicht in der Werkzeugumgebung gespeichert.

Resultierende Länge von orientierbaren Werkzeugträgern und Transformationen

Hinweis

Sowohl bei orientierbaren Werkzeugträgern als auch bei Transformationen gibt es Systemvariablen bzw. Maschinendaten, die wie zusätzliche Werkzeuglängenkomponenten wirken, und die ganz oder teilweise den von diesen ausführbaren Drehungen unterworfen werden. Die zusätzlichen Werkzeuglängenanteile, die sich dadurch ergeben, müssen beim Aufruf von TOOLENV mit abgelegt werden, da sie einen Teil der Umgebung bilden, in der das Werkzeug eingesetzt wird.

Adaptertransformation

Die Adaptertransformation ist eine Eigenschaft des Werkzeugadapters und damit des Gesamtwerkzeugs. Sie ist deshalb nicht Bestandteil einer Werkzeugumgebung, die auf ein anderes Werkzeug angewendet werden kann.

Durch das Abspeichern der gesamten für die Bestimmung der Werkzeuggesamtlänge erforderlichen Daten ist es möglich, zu einem späteren Zeitpunkt die effektive Länge eines

Werkzeugs zu berechnen, auch wenn es zu diesem Zeitpunkt nicht mehr aktiv ist bzw. wenn sich die Umgebungsbedingungen (z. B. G-Befehle oder Settingdaten) geändert haben. Ebenso kann die effektive Länge eines anderen Werkzeugs berechnet werden mit der Annahme, es würde unter den gleichen Bedingungen eingesetzt wie das Werkzeug, für das der Status abgespeichert wurde.

Maximale Anzahl Datensätze für Werkzeugumgebungen

Mit dem Maschinendatum MD18116 $MN_MM_NUM_TOOL_ENV wird festgelegt, wie viele Datensätze zur Beschreibung von Werkzeugumgebungen maximal abgelegt werden können. Die Daten liegen im TOA-Bereich. Sie bleiben auch beim Ausschalten der Steuerung erhalten.

Eine Datensicherung ist nicht möglich. Das bedeutet, dass diese Daten nicht zwischen verschiedenen Steuerungen übertragen werden können.

2.16.2 Werkzeugumgebung löschen (DELTOOLENV)

Mit der Funktion DELTOOLENV können Datensätze zur Beschreibung von Werkzeugumgebungen gelöscht werden. Löschen bedeutet, dass auf den unter einem bestimmten Namen abgelegten Datensatz nicht mehr zugegriffen werden kann (ein Zugriffsversuch führt zu einem Alarm).

Hinweis

Datensätze können nur mit der Funktion DELTOOLENV, durch einen INITIAL.INI-Download oder durch einen Kaltstart (NC-Hochlauf mit Standard-Maschinendaten) gelöscht werden. Es gibt keine weiteren automatischen Löschvorgänge.

Syntax

```
<Status> = DELTOOLENV(<Name>)
<Status> = DELTOOLENV()
```

Bedeutung

`DELTOOLENV(...)`:	Vordefinierte Funktion zum Löschen einer Werkzeugumgebung		
	Alleine im Satz:		ja
`<Status>`:	Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.		
	Datentyp:		INT
	Wert:	0	Funktion OK
		-1	Kein Speicherplatz für Werkzeugumgebungen reserviert: MD18116 $MN_MM_NUM_TOOL_ENV = 0 D. h., die Funktionalität "Werkzeugumgebungen" ist nicht vorhanden.
		-2	Ein Werkzeugumgebung mit dem angegebenen Namen existiert nicht.
Parameter			

1	<Name>:	Name des zu löschenden Datensatzes	
		Datentyp:	STRING
DELTOOLENV():		DELTOOLENV() ohne Angabe eines Namens löscht alle Datensätze zur Beschreibung von Werkzeugumgebungen	

2.16.3 T-, D- und DL-Nummer lesen (GETTENV)

Die Funktion GETTENV dient dazu, die in einer Werkzeugumgebung abgelegte T-, D- und DL-Nummer zu lesen.

Syntax

```
<Status> = GETTENV(<Name>, <TDDL>)
```

Bedeutung

GETTENV(...):		Vordefinierte Funktion zum Lesen von T-, D- und DL-Nummern in einem Datensatz zur Beschreibung einer Werkzeugumgebung	
		Alleine im Satz:	ja
<Status>:		Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.	
		Datentyp:	INT
		Wert: 0	Funktion OK
		-1	Kein Speicherplatz für Werkzeugumgebungen reserviert: MD18116 $MN_MM_NUM_TOOL_ENV = 0 D. h., die Funktionalität "Werkzeugumgebungen" ist nicht vorhanden.
		-2	Ein Werkzeugumgebung mit dem angegebenen Namen existiert nicht.
Parameter			
1	<Name>:	Name der Datensatzes, aus dem die T-, D- und DL-Nummer gelesen werden sollen	
		Datentyp:	STRING
2	<TDDL>:	Das Feld dieses Ergebnisparameters enthält die T-, D- und DL-Nummer des Werkzeugs, dessen Werkzeugumgebung im angegebenen Datensatz abgelegt ist: • <TDDL> [0]: T-Nummer • <TDDL> [1]: D-Nummer • <TDDL> [2]: DL-Nummer	
		Datentyp:	INT[3]
GETTENV(,<TDDL>), GETTENV("",<TDDL>):		Es ist zulässig, beim Aufruf der Funktion GETTENV den ersten Parameter wegzulassen oder als ersten Parameter den Nullstring zu übergeben. In diesen beiden Sonderfällen werden in <TDDL> die T-, D- und DL-Nummern des **aktiven** Werkzeugs zurückgeliefert.	

2.16.4 Informationen zu gespeicherten Werkzeugumgebungen lesen ($P_TOOLENVN, ($P_TOOLENV)

Informationen zu gespeicherten Werkzeugumgebungen sind über folgende Systemvariablen lesbar:

$P_TOOLENVN:	Liefert die Anzahl der mittels TOOLENV definierten (und noch nicht gelöschten) Datensätze zur Beschreibung von Werkzeugumgebungen			
	Syntax:	<n> = $P_TOOLENVN		
	Bedeutung:	<n>:	Anzahl der definierten Datensätze	
			Datentyp:	INT
			Wertebereich:	0 ... MD18116 $MN_MM_NUM_TOOL_ENV
	Der Zugriff auf diese Systemvariable ist auch dann zulässig, wenn keine Werkzeugumgebungen möglich sind (MD18116 = 0). In diesem Fall ist der Rückgabewert "0".			
$P_TOOLENV:	Liefert den Namen des <i>-ten Datensatzes zur Beschreibung einer Werkzeugumgebung			
	Syntax:	<Name> = $P_TOOLENV[<i>]		
	Bedeutung:	<Name>:	Name des Datensatzes mit der Nummer <i>	
			Datentyp:	STRING
		<i>:	Nummer des Datensatzes	
			Datentyp:	INT
			Wertebereich:	1 ... $P_TOOLENVN
	Die Zuordnung der Nummern zu den Datensätzen ist nicht fix, sondern kann sich durch das Löschen oder Neuanlegen von Datensätzen verändern. Die Nummerierung der Datensätze wird intern vorgenommen. Verweist <i> auf einen nicht definierten Datensatz, wird der Nullstring zurückgeliefert. Ist der Index <i> ungültig, d. h. <i> ist kleiner 1 oder größer als die maximale Anzahl Datensätze für Werkzeugumgebungen (MD18116 $MN_MM_NUM_TOOLENV), wird folgender Alarm ausgegeben: Alarm 17020 "unerlaubter Array-Index1"			

2.16.5 Werkzeuglängen bzw. -längenkomponenten lesen (GETTCOR)

Die Funktion GETTCOR dient dazu, Werkzeuglängen bzw. Werkzeuglängenkomponenten auszulesen.

Dabei kann durch Parametrierung angegeben werden, welche Komponenten berücksichtigt werden sollen, und unter welchen Einsatzbedingungen das Werkzeug betrachtet werden soll.

Syntax

```
<Status> = GETTCOR(<Len>[, <Comp>, <Stat>, <T>, <D>, <DL>])
```

Bedeutung

GETTCOR(...):	Vordefinierte Funktion zum Lesen von Werkzeuglängen bzw. Werkzeuglängenkomponenten		
	Alleine im Satz:		ja
<Status>:	Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.		
	Datentyp:		INT
	Wert:	0	Funktion OK
		-1	Kein Speicherplatz für Werkzeugumgebungen reserviert: MD18116 $MN_MM_NUM_TOOL_ENV = 0 D. h., die Funktionalität "Werkzeugumgebungen" ist nicht vorhanden.
		-2	Eine Werkzeugumgebung mit dem unter <Stat> angegebenen Namen existiert nicht.
		-3	Ungültiger String im Parameter <Comp>. Ursache dieses Fehlers können ungültige oder doppelt programmierte Zeichen sein.
		-4	Ungültige T-Nummer
		-5	Ungültige D-Nummer
		-6	Ungültige DL-Nummer
		-7	Zugriffsversuch auf nicht vorhandenes Speichermodul.
		-8	Zugriffsversuch auf eine nicht vorhandene Option (z. B. programmierbare Werkzeugorientierung, Werkzeugverwaltung).
		-9	Der String <Comp> enthält einen Doppelpunkt (Kennung für die Spezifikationen eines Koordinatensystems), es folgt jedoch kein gültiges Zeichen zur Bezeichnung des verlangten Koordinatensystems.
Parameter			

1	<Len>:	Ergebnisvektor	
		Datentyp:	REAL[11]
		Die Vektorkomponenten sind in folgender Reihenfolge angeordnet: • <Len> [0]: Werkzeugtyp • <Len> [1]: Schneidenlage • <Len> [2]: Abszisse • <Len> [3]: Ordinate • <Len> [4]: Applikate • <Len> [5]: Werkzeugradius Als Bezugskoordinatensystem für die Längenkomponenten gilt das in <Comp> und <Stat> definierte Koordinatensystem. Ist in <Comp> kein Koordinatensystem definiert, werden die Werkzeuglängen im Maschinenkoordinatensystem dargestellt. Die Zuordnung von Abszisse, Ordinate und Applikate zu den Geometrieachsen ist abhängig von der in der verwendeten Werkzeugumgebung aktiven Ebene. D. h., bei G17 ist die Abszisse parallel zu X, bei G18 zu Z usw. Die Komponenten <Len>[6] bis <Len>[10] enthalten die zusätzlichen Parameter, die für die Geometriebeschreibung eines Werkzeugs angegeben werden können (z. B. $TC_DP7 bis $TC_DP11 für die Geometrie bzw. die entsprechenden Komponenten für Verschleiß bzw. Summen- und Einrichtekorrekturen). Diese 5 zusätzlichen Elemente und der Werkzeugradius sind nur für die Komponenten E, G, S und W definiert. Deren Bewertung hängt nicht von <Stat> ab. Die entsprechenden Werte in <Len>[6] bis <Len>[10] können deshalb nur dann ungleich null sein, wenn an der Werkzeuglängenberechnung mindestens eine der vier genannten Komponenten beteiligt ist. Die übrigen Komponenten haben keinen Einfluss auf das Ergebnis. Die Maßangaben beziehen sich auf das Grundsystem der Steuerung (inch bzw. metrisch).	

2	<Comp>:	Werkzeuglängenkomponenten (optional)			
		Datentyp:	STRING		
		Die Zeichenkette besteht aus zwei Teilstrings, die durch einen Doppelpunkt voneinander getrennt sind. Allg. Form: "<SubStr_1> [: <SubStr_2]"			
		<SubStr_1>:	Der **erste Teilstring** bezeichnet die Werkzeuglängenkomponenten, die bei der Werkzeuglängenberechnung berücksichtigt werden sollen. Die Reihenfolge der Zeichen im Teilstring sowie deren Schreibweise (groß oder klein) ist beliebig. Zwischen den Zeichen können beliebig viele Leerzeichen oder Tabulatorzeichen (white spaces) eingefügt werden. **Hinweis:** Die Zeichen im Teilstring dürfen **nicht doppelt** programmiert werden!		
			Zeichen:	-	Minuszeichen (nur als erstes Zeichen zulässig!) Es wird die gesamte Werkzeuglänge abzüglich der Komponenten berechnet, die im nachfolgenden String spezifiziert sind.
				C	Adaptermaß oder Basismaß (diejenige der beiden alternativ existierenden Komponenten, die für das verwendete Werkzeug aktiv ist)
				E	Einrichtkorrekturen
				G	Geometrie
				K	Kinematische Transformation (wird nur bei generischer 3-, 4- und 5-Achs-Transformation ausgewertet!)
				S	Summenkorrekturen
				T	Orientierbarer Werkzeugträger
				W	Verschleiß
			Ist der erste Teilstring (mit Ausnahme von white spaces) leer, so bedeutet das, dass die gesamte Werkzeuglänge unter Berücksichtigung aller Komponenten berechnet werden soll. Gleiches gilt auch dann, wenn der Parameter <Comp> nicht angegeben wird.		
		<Substr_2>:	Der optionale **zweite Teilstring** bezeichnet das Koordinatensystem, in dem die Werkzeuglänge ausgegeben werden soll. Der zweite Teilstring besteht nur aus einem einzigen relevanten Zeichen.		
			Zeichen:	A	Einstellbares Koordinatensystem (ACS)
				B	Basiskoordinatensystem (BKS)
				K	Werkzeugkoordinatensystem der kinematischen Transformation (KCS)
				M	Maschinenkoordinatensystem (MKS)
				T	Werkzeugkoordinatensystem (TCS)
				W	Werkstückkoordinatensystem (WKS)
			Wird kein Koordinatensystem angegeben, erfolgt die Bewertung im Maschinenkoordinatensystem (MKS). Die gegebenen-		

			falls zu berücksichtigenden Drehungen werden durch die in <Stat> definierte Werkzeugumgebung festgelegt.
3	<_Stat>:	Name des Datensatzes zur Beschreibung einer Werkzeugumgebung (optional)	
		Datentyp:	STRING
		Ist der Wert dieses Parameters der Nullstring ("") oder wird er nicht angeben, wird der aktuelle Zustand verwendet. Ist kein Werkzeug spezifiziert, wird das aktuelle Werkzeug verwendet.	
4	<T>:	Interne T-Nummer des Werkzeugs (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben oder ist sein Wert "0", wird das in <Stat> abgelegte Werkzeug verwendet. Ist der Wert dieses Parameters "-1", wird die T-Nummer des aktiven Werkzeugs verwendet. Es ist auch zulässig, die Nummer des aktiven Werkzeugs explizit anzugeben. **Hinweis:** Ist <Stat> nicht angegeben, wird als Werkzeugumgebung der aktuelle Zustand verwendet. Da mit <T> = 0 auf die in der Werkzeugumgebung gespeicherte T-Nummer verwiesen wird, wird in dieser das aktive Werkzeug verwendet, d. h. die Angaben <T> = 0 und <T> = -1 sind in diesem Sonderfall gleichbedeutend.	
5	<D>:	Schneidennummer des Werkzeugs (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben oder ist sein Wert "0", richtet sich die verwendetet D-Nummer nach der Quelle der T-Nummer. Wird die T-Nummer aus der Werkzeugumgebung verwendet, wird auch die D-Nummer der Werkzeugumgebung gelesen, andernfalls die des aktuell aktiven Werkzeugs.	
6	<DL>:	Nummer der ortsabhängigen Korrektur (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben, richtet sich die verwendete DL-Nummer nach der Quelle der T-Nummer. Wird die T-Nummer aus der Werkzeugumgebung verwendet, wird auch die D-Nummer der Werkzeugumgebung gelesen, andernfalls die des aktuell aktiven Werkzeugs.	

Beispiele

`GETTCOR(_LEN)`	Berechnet die Werkzeuglänge des aktuellen aktiven Werkzeugs im Maschinenkoordinatensystem unter Berücksichtigung aller Komponenten.
`GETTCOR(_LEN,"CGW:W")`	Es wird die Werkzeuglänge bestehend aus Adapter- bzw. Basismaß, Geometrie und Verschleiß für das aktive Werkzeug berechnet. Weitere Anteile wie z. B. orientierbarer Werkzeugträger oder kinematische Transformationen werden nicht berücksichtigt. Die Ausgabe erfolgt im Werkstückkoordinatensystem.

`GETTCOR (_LEN,"-K:B")`	Es wird die gesamte Werkzeuglänge des aktiven Werkzeugs ohne Berücksichtigung der Längenkomponenten einer eventuell aktiven kinematischen Transformation berechnet. Ausgabe im Basiskoordinatensystem.
`GETTCOR (_LEN,":M","Testenv1",,3)`	Es wird die gesamte Werkzeuglänge für das in der Werkzeugumgebung mit dem Namen "Testenv1" abgelegte Werkzeug im Maschinenkoordinatensystem berechnet. Die Berechnung erfolgt jedoch unabhängig von der abgelegten Schneidennummer für die Schneidennummer D3.

Weitere Informationen

Adaptertransformation / Orientierbare Werkzeugträger / Kinematische Transformation

Die Drehungen und Komponentenvertauschungen, die gegebenenfalls von der Adaptertransformation, dem orientierbaren Werkzeugträger und einer kinematischen Transformation ausgeführt werden, sind Bestandteil der Werkzeugumgebung. Sie werden deshalb immer ausgeführt, auch dann, wenn die entsprechende Längenkomponente nicht berücksichtigt werden soll. Ist das nicht gewünscht, müssen Werkzeugumgebungen definiert werden, bei denen die entsprechenden Transformationen nicht aktiv sind. In vielen Fällen (nämlich immer dann, wenn an einer Maschine keine Transformationen bzw. orientierbare Werkzeugträger eingesetzt werden), erfüllen die abgespeicherten Datensätze für Werkzeugumgebungen diese Bedingungen automatisch, sodass sie vom Anwender nicht speziell beachtet werden müssen.

Dreh- und Schleifwerkzeuge: Berechnung der Werkzeuglänge in Abhängigkeit von MD20360 $MC_TOOL_PARAMETER_DEF_MASK

Wie bei Dreh- und Schleifwerkzeugen der Verschleiß bzw. die Werkzeuglänge in einer eventuell vorhandenen Durchmesserachse zu bewerten ist, wird festgelegt im Maschinendatum:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK (Definition der WZ-Parameter)

Bit	Wert	
0	Bei Dreh- und Schleifwerkzeugen wird der **Verschleißparameter** der Planachse als Durchmesserwert eingerechnet:	
	= 0 (Default)	nein
	= 1	ja
1	Bei Dreh- und Schleifwerkzeugen wird die **Werkzeuglängenkomponente** der Planachse als Durchmesserwert eingerechnet:	
	= 0 (Default)	nein
	= 1	ja

Sind die betreffenden Bits gesetzt, wird der zugehörige Eintrag mit dem Faktor 0,5 bewertet. Diese Bewertung schlägt sich auch in der von GETTCOR gelieferten Werkzeuglänge nieder.

Beispiel:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK = 3

MD20100 $MC_DIAMETER_AX_DEF (Geometrieachse mit Planachsfunktion) = "X"

X ist Durchmesserachse (Standard-Drehmaschinenkonfiguration)

Programmcode	Kommentar
N30 $TC_DP1[1,1]=500	
N40 $TC_DP2[1,1]=2	
N50 $TC_DP3[1,1]=3.0	; Geometrie L1
N60 $TC_DP4[1,1]=4.0	
N70 $TC_DP5[1,1]=5.0	
N80 $TC_DP12[1,1]=12.0	; Verschleiß L1
N90 $TC_DP13[1,1]=13.0	
N100 $TC_DP14[1,1]=14.0	
N110 T1 D1 G18	
N120 R1=GETTCOR(_LEN,"GW")	
N130 R3=_LEN[2]	; 17.0 (= 4.0 + 13.0)
N140 R4=_LEN[3]	; 7.5 (= 0.5 * 3.0 + 0.5 * 12.0)
N150 R5=_LEN[4]	; 19.0 (= 5.0 + 14.0)
N160 M30	

Längenkomponenten von kinematischer Transformation und orientierbarem Werkzeugträger

Wird bei der Werkzeuglängenberechnung ein **orientierbarer Werkzeugträger** berücksichtigt, gehen folgende Vektoren in die Werkzeuglängenberechnung ein:

Typ	Vektoren
M	l1 und l2
T	l1, l2 und l3
P	Werkzeuglänge wird durch den orientierbaren Werkzeugträger nicht beeinflusst.

Bei der generischen **5-Achstransformation** gehen bei den Trafotypen 24 und 56 folgende Maschinendaten in die Werkzeuglängenberechnung ein:

Trafotyp	Maschinendaten
24	MD24550/24650 $MC_TRAFO5_BASE_TOOL_1/2 MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 MD24558/24658 $MC_TRAFO5_PART_OFFSET_1/2
56	MD24550/24650 $MC_TRAFO5_BASE_TOOL_1/2 MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2

Der Trafotyp 56 (bewegliches Werkzeug und bewegliches Werkstück) entspricht dem Typ M beim orientierbaren Werkzeugträger.

Bei dieser 5-Achs-Transformation entspricht in den bisherigen Softwareständen der Vektor MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 (Vektor des kinematischen Versatzes der 1./2. 5-Achstransformation im Kanal) der Summe der beiden Vektoren l_1 und l_3 beim orientierbaren Werkzeugträger vom Typ M.

Für die Transformation ist in beiden Fällen nur die Summe relevant. Die Art der Zusammensetzung aus den beiden Einzelkomponenten ist bedeutungslos. Bei der Berechnung der Werkzeuglänge ist es jedoch von Bedeutung, welcher Anteil dem Werkzeug und welcher dem Werkzeugtisch zuzuordnen ist. Deshalb wurde das Maschinendatum MD24558/24658 $MC_TRAFO5_JOINT_OFFSET_PART_1/2 (Vektor kinematischer Versatz

im Tisch) eingeführt. Es entspricht dem Vektor l3. Das Maschinendatum MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 entspricht jetzt nicht mehr der Summe aus l1 und l3, sondern nur noch dem Vektor l1. Ist das Maschinendatum MD24558/24658 $MC_TRAFO5_JOINT_OFFSET_PART_1/2 gleich null, so ist das Verhalten identisch zum bisherigen Verhalten.

Kompatibilität

Die Funktion GETTCOR dient zusammen mit den Funktionen TOOLENV und SETTCOR unter anderem dazu, Teile der Funktionalität zu ersetzen, die bisher extern in den Messzyklen realisiert war.

In den Messzyklen wurde nur ein Teil der Parameter ausgewertet, die letztlich die effektive Werkzeuglänge bestimmen. Die genannten Funktionen können so parametriert werden, dass das Verhalten der Messzyklen bezüglich der Werkzeuglängenberechnung reproduziert werden kann.

2.16.6 Werkzeugkomponenten ändern (SETTCOR)

Die Funktion SETTCOR dient dazu, Werkzeugkomponenten unter Berücksichtigung aller Randbedingungen, die in die Bewertung der einzelnen Komponenten eingehen können, zu verändern.

Hinweis

Zur Terminologie: Wenn im Folgenden im Zusammenhang mit der Werkzeuglänge von Werkzeugkomponenten die Rede ist, sind die vektoriell betrachteten Komponenten gemeint, aus denen sich die gesamte Werkzeuglänge zusammensetzt (z. B. Geometrie oder Verschleiß). Eine solche Komponente setzt sich deshalb aus drei Einzelwerten (L1, L2, L3) zusammen, die im Folgenden als Koordinatenwerte bezeichnet werden.

Die Werkzeugkomponente "Geometrie" besteht somit beispielsweise aus den drei Koordinatenwerten $TC_DP3 bis $TC_DP5.

Syntax

```
<Status> = SETTCOR(<CorVal>, <Comp>, [<CorComp>, <CorMode>, <GeoAx>,
<Stat>, <T>, <D>, <DL>])
```

Bedeutung

`SETTCOR(...)`:	Vordefinierte Funktion zum Ändern von Werkzeugkomponenten	
	Alleine im Satz:	ja

<Status>:	Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.		
	Datentyp:		INT
	Wert:	0	Funktion OK
		-1	Kein Speicherplatz für Werkzeugumgebungen reserviert: MD18116 $MN_MM_NUM_TOOL_ENV = 0 D. h., die Funktionalität "Werkzeugumgebungen" ist nicht vorhanden.
		-2	Eine Werkzeugumgebung mit dem unter <Stat> angegebenen Namen existiert nicht.
		-3	Ungültiger String im Parameter <Comp>. Ursache dieses Fehlers können ungültige oder doppelt programmierte Zeichen sein.
		-4	Ungültige T-Nummer.
		-5	Ungültige D-Nummer.
		-6	Ungültige DL-Nummer.
		-7	Zugriffsversuch auf nicht vorhandenes Speichermodul.
		-8	Zugriffsversuch auf eine nicht vorhandene Option (z. B. programmierbare Werkzeugorientierung, Werkzeugverwaltung).
		-9	Unzulässiger Zahlenwert für den Parameter <CorComp>.
		-10	Unzulässiger Zahlenwert für den Parameter <CorMode>.
		-11	Die Inhalte der Parameter <Comp> und <CorComp> sind widersprüchlich.
		-12	Die Inhalte der Parameter <Comp> und <CorMode> sind widersprüchlich.
		-13	Der Inhalt des Parameters <GeoAx> bezeichnet keine Geometrieachse.
		-14	Schreibversuch auf nicht vorhandene Einrichtekorrektur.
Parameter			
1	<CorVal>:	Korrekturvektor In dem durch <Stat> definierten Werkstückkoordinatensystem (WKS) gilt folgende Zuordnung: • <CorVal> [0]: Abszisse • <CorVal> [1]: Ordinate • <CorVal> [2]: Applikate Soll nur eine Werkzeugkomponente korrigiert werden (d. h. keine vektorielle Korrektur, siehe Parameter <CorMode>), steht der Korrekturwert immer in <CorVal>[0], gleichgültig auf welche Achse er wirkt. Der Inhalt der beiden übrigen Komponenten wird dann nicht ausgewertet. Bezieht sich <CorVal> bzw. eine Komponente von <CorVal> auf die Planachse, so wird die Angabe als **Radiusmaß** bewertet. Das bedeutet, dass ein Werkzeug z. B. um das angegebene Maß "länger" wird, was entsprechend zu einer doppelt so großen Änderung des Werkstückdurchmessers führt. Die Maßangaben beziehen sich auf das **Grundsystem** (inch bzw. metrisch) der Steuerung.	
		Datentyp:	REAL[3]

2	<Comp>:	Werkzeugkomponente(n)			
		Datentyp:	STRING		
		Die Zeichenkette besteht aus zwei Teilstrings, die durch einen Doppelpunkt voneinander getrennt sind. Allg. Form: "<SubStr_1> [: <SubStr_2]"			
		<SubStr_1>:	Der **erste Teilstring** muss immer vorhanden sein und kann entweder aus ein oder zwei Zeichen bestehen. Dabei steht das erste oder einzige Zeichen für die 1. Komponente (Val_1) und das zweite Zeichen für die 2. Komponente (Val_2), die entsprechend den nachfolgenden Parametern <CorComp> und <CorMode> bearbeitet werden.		
			Zeichen:	C	Adaptermaß oder Basismaß (diejenige der beiden alternativ existierenden Komponenten, die für das verwendete Werkzeug aktiv ist)
				E	Einrichtkorrekturen
				G	Geometrie
				S	Summenkorrekturen
				W	Verschleiß
		<Substr_2>:	Der **zweite Teilstring** ist optional. Er kann alternativ aus dem (einzigen) Buchstaben "W" oder "T" bestehen.		
			Zeichen:	W	Ist der zweite Teilstring leer oder enthält er den Buchstaben "W", werden die Korrekturwerte so verrechnet, als seien sie im Werk**stück**koordinatensystem (WKS) vermessen worden.
				T	Enthält der zweite Teilstring den Buchstaben "T", werden Korrekturwerte werden so verrechnet, als seien sie im Werk**zeug**koordinatensystem (**T**ool Coordinate System, TCS) vermessen worden.
		Die Schreibweise der Zeichen im String (groß oder klein) ist beliebig. Es können beliebig viele Leerzeichen oder Tabulatorzeichen (white spaces) eingefügt werden.			

3	<CorComp>:	Spezifiziert die Komponente(n) des Werkzeugdatensatzes, die beschrieben werden sollen (optional)		
		Datentyp:		INT
		Wert:	0	Der Korrekturwert <CorVal>[0] bezieht sich auf die im Parameter <GeoAx> übergebene Geometrieachse im Werkstückkoordinatensystem oder im Werkzeugkoordinatensystem (siehe dazu die Beschreibung des Parameters <Comp>). D. h., der Korrekturwert muss in die bezeichneten Werkzeugkomponenten so eingerechnet werden, dass sich unter Berücksichtigung aller Parameter, die die Werkzeuglängenberechnung beeinflussen können, als Ergebnis eine Änderung der gesamten Werkzeuglänge in der gegebenen Achsrichtung um den vorgegebenen Wert ergibt. Diese Änderung soll erreicht werden durch die Korrektur der unter <Comp> angegebenen Komponente und der unter <CorMode> (siehe nachfolgenden Parameter) angegebenen symbolischen Rechenvorschrift. Die resultierende Korrektur kann sich deshalb in allen drei Achskomponenten auswirken.
			1	Wie "0", jedoch vektoriell. Der Inhalt des Vektors <CorVal> bezieht sich auf Abszisse, Ordinate und Applikate im Werkstückkoordinatensystem oder Werkzeugkoordinatensystem (siehe dazu die Beschreibung des Parameters <Comp>). Der nachfolgende Parameter <GeoAx> wird nicht ausgewertet.
			2	Vektorielle Korrektur, d. h. L1, L2 und L3 können simultan verändert werden. Im Gegensatz zu den Varianten "0" und "1" beziehen sich die in <CorVal> enthaltenen Korrekturwerte jedoch auf die Koordinaten der Val_1-Komponente (siehe nachfolgenden Parameter <CorMode>) des Werkzeugs. Eine eventuell vorhandene Schrägstellung des Werkzeugs gegenüber dem Werkstückkoordinatensystem hat keinen Einfluss auf die Korrektur.
			3 - 5	Korrektur der Werkzeuglänge L1 bis L3 ($TC_DP3 bis $TC_DP5) bzw. der entsprechenden Werte bei Verschleiß, Einrichte- oder Summenkorrekturen. Der Korrekturwert ist in <CorVal>[0] enthalten. Er ist in den Koordinaten der Val_1-Komponente (siehe nachfolgenden Parameter <CorMode>) des Werkzeugs gemessen. Eine eventuell vorhandene Schrägstellung des Werkzeugs gegenüber dem Werkstückkoordinatensystem hat keinen Einfluss auf die Korrektur.
			6	Korrektur des Werkzeugradius ($TC_DP6) bzw. der entsprechenden Werte bei Verschleiß, Einrichte- oder Summenkorrekturen. Die Bits 10 und 11 (Bewertung der Durchmesser- bzw. Durchmesserverschleißangabe wahlweise als Radius- oder Durchmesserangabe) im Maschinendatum MD20360 $MC_TOOL_PARAMETER_DEF_MASK werden berücksichtigt.
			7 – 11	Korrektur von $TC_DP7 bis $TC_DP11 bzw. der entsprechenden Werte bei Verschleiß, Einrichte- oder Summenkorrekturen. Diese Parameter werden wie der Werkzeugradius behandelt.
		Ist dieser Parameter nicht angeben, ist sein Wert "0".		

<table>
<tr><td rowspan="8">4</td><td rowspan="8"><CorMode>:</td><td colspan="3">Spezifiziert die Art der auszuführenden Schreiboperation (optional)</td></tr>
<tr><td colspan="2">Datentyp:</td><td>INT</td></tr>
<tr><td rowspan="4">Wert:</td><td>0</td><td>Val_{1neu} = <CorVal></td></tr>
<tr><td>1</td><td>Val_{1neu} = Val_{1alt} + <CorVal></td></tr>
<tr><td>2</td><td>Val_{1neu} = <CorVal>
Val_{2neu} = 0</td></tr>
<tr><td>3</td><td>Val_{1neu} = Val_{1alt} + Val_{2alt} + <CorVal>
Val_{2neu} = 0</td></tr>
<tr><td colspan="3">Die Schreibweise Val_{1alt} + Val_{2alt} ist symbolisch zu verstehen. Werden die beiden Komponenten (aufgrund des Zustandes von <_Stat>) unterschiedlich bewertet, d. h. ist zwischen beiden Komponenten eine Drehung wirksam, so wird Val_{2alt} vor der Addition so transformiert, dass die resultierende Werkzeuglänge nach dem Löschen von Val_{2neu} und vor der Addition von <CorVal> unverändert bleibt.

<CorVal> bezieht sich immer auf Val_1. <CorVal> ist ein Wert, der abhängig vom zweiten Teils des Parameters <Comp> im Werkstückkoordinatensystem (WKS) oder im Werkzeugkoordinatensystem (TCS) gemessen wird. Er ist deshalb gegenüber der Werkzeugkomponente, in die er eingerechnet werden soll, gegebenenfalls bereits transformiert. Er kann somit nicht unmittelbar mit dem abgespeicherten Wert verrechnet werden, sondern muss vor der Addition zu Val_1 bzw. Val_2 zurücktransformiert werden. Das kann zur Folge haben, dass sich die Korrektur in einer anderen als in der durch <CorComp> definierten Achse auswirkt, bzw. dass diese auf mehrere Achsen wirkt.

Für den Fall <CorComp> = 0, d. h., wenn <CorVal> keinen Vektor, sondern nur einen Einzelwert enthält, werden die beschriebene Operationen in den Koordinaten ausgeführt, in denen <CorVal> gemessen wurde (WKS / TCS). Insbesondere gilt das auch für das Nullsetzen von Val_{2neu} in den Varianten 2 und 3. Dieses Ergebnis wird dann in die Koordinaten des Werkzeugs zurücktransformiert. Das kann zur Folge haben, dass keine der nullzusetzenden Koordinatenwerte (L1, L2, L3) Null wird bzw. dass Koordinatenwerte, die vorher Null waren, nun ungleich Null sind. Werden die entsprechenden Operationen jedoch sukzessive für alle drei Geometrieachsen ausgeführt, ist gewährleistet, dass auch alle drei Koordinatenwerte der zu löschenden Komponente Null werden. Ist das Werkzeug gegenüber dem Werkstückkoordinatensystem nicht gedreht, oder aber so gedreht, dass alle Werkzeugkomponenten parallel zu den Koordinatenachsen bleiben (Achsvertauschungen), ist gewährleistet, dass sich jeweils auch nur eine Werkzeugkoordinate ändert.

Die sukzessive Ausführung der gleichen Operation (<CorMode>) mit <CorComp> = 0 für alle drei Koordinatenachsen in beliebiger Reihenfolge ist identisch zu der einmaligen Ausführung der gleichen Operation mit <CorComp>=1.

Für die Parameterwerte "0" und "1" muss der Parameter <Comp> einen und für die Parameterwerte "2" und "3" zwei Zeichen enthalten.

Beispiel:

<Comp> enthält den String "ES", <CorMode> den Wert "2"

⇒ $Einrichtkorrektur_{neu}$ = <CorVal>, $Summenkorrektur_{neu}$ = 0

Ist der Parameter <CorMode> nicht angeben, ist sein Wert "0".</td></tr>
</table>

5	<GeoAx>:	Gibt den Index der Geometrieachse an, in der der Korrekturwert <CorVal>[0] gemessen wurde (optional)	
		Datentyp:	INT
		Wertebereich:	0 ... 2
		Die Indizes 0 bis 2 beziehen sich auf Abszisse, Ordinate und Applikate in der wirksamen Ebene (G17/G18/G19) der aktuellen Werkzeugumgebung. Der Inhalt dieses Parameters wird nur dann ausgewertet, wenn der Parameter <CorComp> den Wert "0" hat.	
6	<Stat>:	Name des Datensatzes zur Beschreibung einer Werkzeugumgebung (optional)	
		Datentyp:	STRING
		Ist der Wert dieses Parameters der Nullstring ("") oder wird er nicht angeben, wird der aktuelle Zustand verwendet. Ist kein Werkzeug spezifiziert, wird das aktuelle Werkzeug verwendet.	
7	<T>:	Interne T-Nummer des Werkzeugs (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben oder ist sein Wert "0", wird das in <Stat> abgelegte Werkzeug verwendet. Ist der Wert dieses Parameters "-1", wird die T-Nummer des aktiven Werkzeugs verwendet. Es ist auch zulässig, die Nummer des aktiven Werkzeugs explizit anzugeben. **Hinweis:** Ist <Stat> nicht angegeben, wird als Werkzeugumgebung der aktuelle Zustand verwendet. Da mit <T> = 0 auf die in der Werkzeugumgebung gespeicherte T-Nummer verwiesen wird, wird in dieser das aktive Werkzeug verwendet, d. h. die Angaben <T> = 0 und <T> = -1 sind in diesem Sonderfall gleichbedeutend.	
8	<D>:	Schneidennummer des Werkzeugs (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben oder ist sein Wert "0", richtet sich die verwendetet D-Nummer nach der Quelle der T-Nummer. Wird die T-Nummer aus der Werkzeugumgebung verwendet, wird auch die D-Nummer der Werkzeugumgebung gelesen, andernfalls die des aktuell aktiven Werkzeugs.	
9	<TL>:	Nummer der ortsabhängigen Korrektur (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben, richtet sich die verwendete DL-Nummer nach der Quelle der T-Nummer. Wird die T-Nummer aus der Werkzeugumgebung verwendet, wird auch die D-Nummer der Werkzeugumgebung gelesen, andernfalls die des aktuell aktiven Werkzeugs. Spezifizieren T, D und DL ein Werkzeug ohne ortsabhängige Korrekturen, dürfen im Parameter <Comp> keine Summen- oder Einrichtkorrekturen spezifiziert werden (Fehlercode in <Status>).	

Hinweis

Nicht alle möglichen Kombinationen der drei Parameter <Comp>, <CorComp> und <CorMode> sind sinnvoll. Beispielsweise erfordert die Rechenvorschrift 3 in <CorComp> die Angabe von zwei Zeichen in <Comp>. Wird eine unzulässige Parameterkombination angegeben, wird im <Status> ein entsprechender Fehlercode zurückgegeben.

Beispiele

Beispiel 1

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fräswerkzeug
N30 $TC_DP3[1,1]=10.0	; Geometrie L1
N40 $TC_DP12[1,1]=1.0	; Verschleiß L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"G",0,0,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-Position X0.000 Y0.000 **Z1.333**
N90 M30	

<CorComp> ist "0", deshalb wird der in Z-Richtung wirkende Koordinatenwert der Geometrie-Komponente durch den Korrekturwert 0,333 ersetzt.

Die resultierende Werkzeuggesamtlänge ist somit: L1 = 0,333 + 1,000 = 1,333

Beispiel 2

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fräswerkzeug
N30 $TC_DP3[1,1]=10.0	; Geometrie L1
N40 $TC_DP12[1,1]=1.0	; Verschleiß L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"W",0,1,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-Position X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> ist "1", deshalb wird der in Z-Richtung wirkende Korrekturwert von 0,333 zum Verschleißwert von 1,0 addiert.

Die resultierende Werkzeuggesamtlänge ist somit: L1 = 10,0 + 1,333 = 11,333

Beispiel 3

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fräswerkzeug
N30 $TC_DP3[1,1]=10.0	; Geometrie L1
N40 $TC_DP12[1,1]=1.0	; Verschleiß L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"GW",0,2,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-Position X0.000 Y0.000 Z0.333
N90 M30	

<CorComp> ist "2", deshalb wird die in Z-Richtung wirkende Korrektur in die Geometrie-Komponente eingetragen (der alte Wert wird überschrieben), und der Verschleißwert wird gelöscht.

Die resultierende Werkzeuggesamtlänge ist somit: L1 = 0,333 + 0,0 = 0,333

Beispiel 4

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fräswerkzeug
N30 $TC_DP3[1,1]=10.0	; Geometrie L1
N40 $TC_DP12[1,1]=1.0	; Verschleiß L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"GW",0,3,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-Position X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> ist "3", deshalb werden Verschleißwert und Korrekturwert zur Geometrie-Komponente addiert und die Verschleißkomponente gelöscht.

Die resultierende Werkzeuggesamtlänge ist somit: L1 = 11,333 + 0,0 = 11,333.

Beispiel 5

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fräswerkzeug
N30 $TC_DP3[1,1]=10.0	; Geometrie L1
N40 $TC_DP12[1,1]=1.0	; Verschleiß L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"GW",0,3,0)**	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-Position X0.333 Y0.000 Z11.000
N90 M30	

<CorComp> ist "3" wie im vorigen Beispiel, die Korrektur wirkt jetzt jedoch auf die Geometrieachse mit dem Index "0" (X-Achse), der wegen G17 bei einem Fräswerkzeug die Werkzeugkomponente L3 zugeordnet ist. Der Aufruf von SETTCOR beeinflusst deshalb die Werkzeugparameter $TC_DP3 und $TC_DP12 nicht. Der Korrekturwert wird stattdessen in $TC_DP5 eingetragen.

Beispiel 6

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; Drehwerkzeug
N30 $TC_DP3[1,1]=10.0	; Geometrie L1
N40 $TC_DP4[1,1]=15.0	; Geometrie L2
N50 $TC_DP12[1,1]=10.0	; Verschleiß L1

Programmcode	Kommentar
N60 $TC_DP13[1,1]=0.0	; Verschleiß L2
N70 _CORVAL[0]=5.0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=**SETTCOR(_CORVAL,"GW",0,3,1)**	
N110 T1 D1 X0 Y0 Z0	; ==> MKS-Position X24.330 Y0.000 Z17.500
N120 M30	

Das Werkzeug ist ein Drehwerkzeug. In Satz N80 wird eine Framedrehung aktiviert, sodass das Basiskoordinatensystem (BKS) gegenüber dem Werkstückkoordinatensystem (WKS) gedreht ist. Der Korrekturwert (N70) wirkt im WKS auf die Geometrieachse mit dem Index "1", d. h., da G18 aktiv ist, auf die X-Achse. Da <CorMode> = 3 gilt, muss der Werkzeugverschleiß in Richtung der X-Achse des WKS nach dem Ausführen von N100 null werden.

Der Inhalt der relevanten Werkzeugparameter ist deshalb am Programmende:

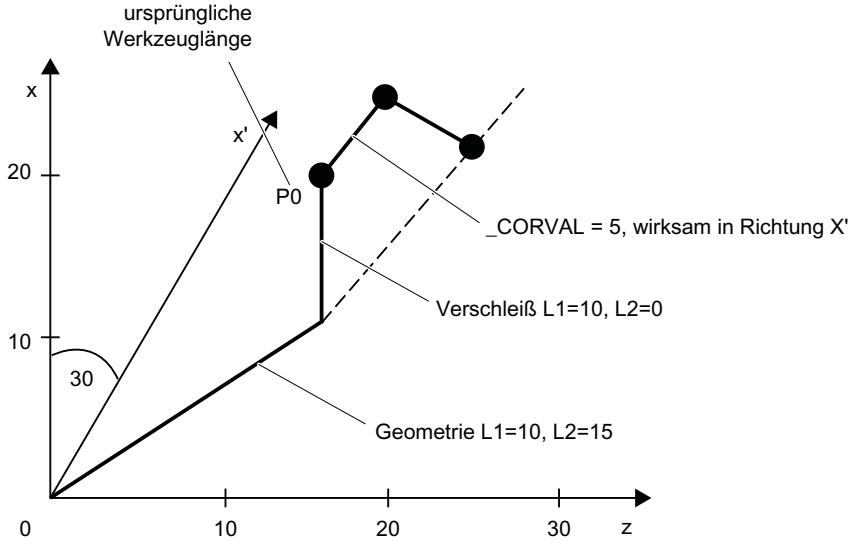
$TC_DP3[1,1] : 21.830 ; Geometrie L1

$TC_DP4[1,1] : 21.830 ; Geometrie L2

$TC_DP12[1,1] : 2.500 ; Verschleiß L1

$TC_DP13[1,1] : -4.330 ; Verschleiß L2

Die geometrischen Verhältnisse sind im folgenden Bild dargestellt. Der gesamte Verschleiß einschließlich _CORVAL wird auf die X'-Richtung im WKS projiziert. Das ergibt den Punkt P2. Die Koordinaten dieses Punkts (gemessen in X-Y-Koordinaten) werden in die Geometriekomponente des Werkzeugs eingetragen. Im Verschleiß bleibt der Differenzvektor P_2 - P_1 übrig. Damit hat der Verschleiß keine Komponente mehr in Richtung von _CORVAL.



Setzt man das Beispielprogramm nach N110 mit den folgenden Anweisungen fort, wird der restliche Verschleiß vollständig in die Geometrie übernommen, da die Korrektur jetzt in der Z'-Achse wirkt (Parameter <GeoAx> = 0):

```
N120 _CORVAL[0]=0.0
```

```
N130 R1=SETTCOR(_CORVAL,"GW",0,3,0)
N140 T1 D1 X0 Y0 Z0 ; ==> MKS-Position X24.330 Y0.000 Z17.500
```

Da der neue Korrekturwert "0" ist, darf sich die Werkzeuggesamtlänge und damit auch die in N140 angefahrene Position nicht verändern. Wäre _CORVAL in N120 ungleich "0", würde sich eine neue Werkzeuggesamtlänge und damit auch eine veränderte Position in N140 ergeben, der Verschleißanteil der Werkzeuglänge wäre jedoch in jedem Fall null, d. h., die gesamte Werkzeuglänge ist anschließend in jedem Fall im Geometrieanteil des Werkzeugs enthalten.

Das gleiche Ergebnis wie mit dem zweimaligen Aufruf der Funktion SETTCOR mit dem Parameter <CorComp> = 0 erreicht man auch durch einmaligen Aufruf mit <CorComp> = 1 (vektorielle Korrektur):

Programmcode	**Kommentar**
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; Drehwerkzeug
N30 $TC_DP3[1,1]=10.0	; Geometrie L1
N40 $TC_DP4[1,1]=15.0	; Geometrie L2
N50 $TC_DP12[1,1]=10.0	; Verschleiß L1
N60 $TC_DP13[1,1]=0.0	; Verschleiß L2
N70 _CORVAL[0]=0.0	
N71 _CORVAL[1]=5.0	
N72 _CORVAL[2]=0.0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=**SETTCOR(_CORVAL,"GW",1,3,1)**	
N110 T1 D1 X0 Y0 Z0	; ==> MKS-Position X24.330 Y0.000 Z17.500
N120 M30	

In diesem Fall sind alle Verschleißkomponenten des Werkzeugs sofort nach dem ersten Aufruf von SETTCOR in N100 gleich null.

Beispiel 7

Programmcode	**Kommentar**
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; Drehwerkzeug
N30 $TC_DP3[1,1]=10.0	; Geometrie L1
N40 $TC_DP4[1,1]=15.0	; Geometrie L2
N50 $TC_DP12[1,1]=10.0	; Verschleiß L1
N60 $TC_DP13[1,1]=0.0	; Verschleiß L2
N70 _CORVAL[0]=5.0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=**SETTCOR(_CORVAL,"GW",3,3)**	
N110 T1 D1 X0 Y0 Z0	; ==> MKS-Position X25.000 Y0.000 Z15.000
N120 M30	

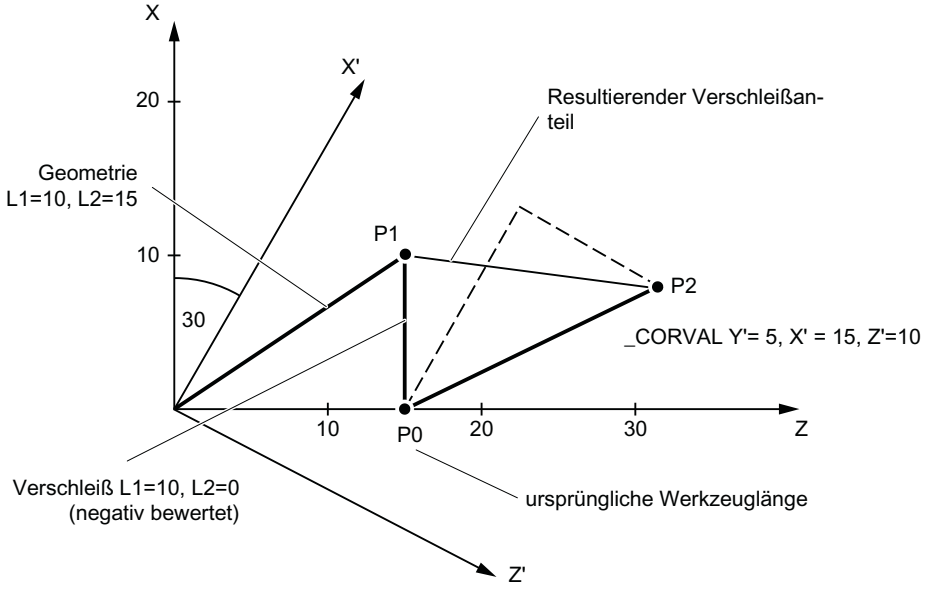
Gegenüber Beispiel 6 ist hier der Parameter <CorComp> = 3, deshalb kann die Angabe des Parameters <GeoAx> entfallen. Der in _CORVAL[0] enthaltene Wert wirkt jetzt unmittelbar auf

die Werkzeugkomponente L1, die Drehung in N80 hat auf das Ergebnis keinen Einfluss, die Verschleißkomponenten in $TC_DP12 wird zusammen mit _CORVAL[0] in den Geometrieanteil übernommen, sodass wegen $TC_DP13 die gesamte Werkzeuglänge bereits nach dem ersten Aufruf von SETTCOR in N100 im Geometrieteil des Werkzeugs steht.

Beispiel 8

```
Programmcode                                Kommentar
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=500                        ; Drehwerkzeug
N30 $TC_DP3[1,1]=10.0                       ; Geometrie L1
N40 $TC_DP4[1,1]=15.0                       ; Geometrie L2
N50 $TC_DP5[1,1]=20.0                       ; Geometrie L3
N60 $TC_DP12[1,1]=10.0                      ; Verschleiß L1
N70 $TC_DP13[1,1]=0.0                       ; Verschleiß L2
N80 $TC_DP14[1,1]=0.0                       ; Verschleiß L3
N90 $SC_WEAR_SIGN=TRUE
N100 _CORVAL[0]=10.0
N110 _CORVAL[1]=15.0
N120 _CORVAL[2]=5.0
N130 ROT Y-30
N140 T1 D1 G18 G0
N150 R1=SETTCOR(_CORVAL,"W",1,1)
N160 T1 D1 X0 Y0 Z0                         ; ==> MKS-Position X7.990 Y25.000 Z31.160
N170 M30
```

In N90 ist das Settingdatum SD42930 $SC_WEAR_SIGN gesetzt, d. h., der Verschleiß muss mit negativem Vorzeichen bewertet werden. Die Korrektur ist vektoriell (<CorComp> = 1), und der Korrekturvektor muss zum Verschleiß addiert werden (<CorMode> = 1). Die geometrischen Verhältnisse in der Z-X-Ebene sind im folgenden Bild dargestellt:



Wegen <CorMode> = 1 bleibt der Geometrieanteil des Werkzeugs unverändert. Der im WKS (Drehung um y-Achse) definierte Korrekturvektor muss so in den Verschleißanteil übernommen werden, dass die gesamte Werkzeuglänge in Bild 3 auf den Punkt P_2 verweist. Der resultierende Verschleißanteil des Werkzeugs ist deshalb durch die Distanz der beiden Punkte P_1 und P_2 gegeben.
Da der Verschleiß wegen des Settingdatums SD42930 $SC_WEAR_SIGN aber negativ bewertet wird, muss die so ermittelte Korrektur mit negativem Vorzeichen in den Korrekturspeicher eingetragen werden. Der Inhalt der relevanten Werkzeugparameter ist am Programmende deshalb:

$TC_DP3[1,1] : 10.000 ; Geometrie L1 (unverändert)

$TC_DP4[1,1] : 15.000 ; Geometrie L2 (unverändert)

$TC_DP5[1,1] : 10.000 ; Geometrie L3 (unverändert)

$TC_DP12[1,1] : 2.010 ; Verschleiß L1 (= 10 - 15 * cos(30) + 10 * sin(30))

$TC_DP13[1,1] : -16.160 ; Verschleiß L2 (= -15 * sin(30) - 10 * cos(30))

$TC_DP14[1,1] : -5.000 ; Verschleiß L3

An der L3-Komponente in Y-Richtung lässt sich die Wirkung des Settingdatums SD42930 $SC_WEAR_SIGN ohne die zusätzliche Komplizierung durch die Framedrehung erkennen.

Weitere Informationen

Dreh- / Schleifwerkzeuge: Berechnung der Werkzeuglänge in Abhängigkeit von MD20360 $MC_TOOL_PARAMETER_DEF_MASK

Wie bei Dreh- / Schleifwerkzeugen der Verschleiß bzw. die Werkzeuglänge in einer eventuell vorhandenen Durchmesserachse zu bewerten ist, wird festgelegt im Maschinendatum:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK.<Bit> = <Wert>

<Bit>	<Wert>	Bedeutung
0	0	Bei Dreh- / Schleifwerkzeugen wird der **Verschleißparameter** der Planachse im **Radiuswert** eingerechnet:
	1	Bei Dreh- / Schleifwerkzeugen wird der **Verschleißparameter** der Planachse als **Durchmesserwert** eingerechnet:
1	0	Bei Dreh- / Schleifwerkzeugen wird die **Werkzeuglängenkomponente** der Planachse als **Radiuswert** eingerechnet:
	1	Bei Dreh- / Schleifwerkzeugen wird die **Werkzeuglängenkomponente** der Planachse als **Durchmesserwert** eingerechnet:

Sind die betreffenden Bits gesetzt, wird der zugehörige Eintrag mit dem Faktor 0,5 bewertet. Die Korrektur mittels SETTCOR wird so ausgeführt, dass die gesamte effektive Werkzeuglängenänderung gleich dem in <CorVal> übergebenen Wert ist. Wird eine Länge auf Grund des Maschinendatums MD20360 $MC_TOOL_PARAMETER_DEF_MASK bei der Längenberechnung mit dem Faktor 0,5 bewertet, muss die Korrektur dieser Komponente deshalb mit dem doppelten übergebenen Wert erfolgen.

Beispiel

MD20360 $MC_TOOL_PARAMETER_DEF_MASK = 2 (Werkzeuglänge muss in der Durchmesserachse mit dem Faktor 0,5 bewertet werden)

Achse X ist die Durchmesserachse.

```
Programmcode                                    Kommentar
N10 DEF REAL _LEN[11]
N20 DEF REAL _CORVAL[3]
N30 $TC_DP1[1,1]=500                            ; Werkzeugtyp
N40 $TC_DP2[1,1]=2                              ; Schneidenlage
N50 $TC_DP3[1,1]=3.                             ; Geometrie - Länge 1
N60 $TC_DP4[1,1]=4.                             ; Geometrie - Länge 2
N70 $TC_DP5[1,1]=5.                             ; Geometrie - Länge 3
N80 _CORVAL[0]=1.
N90 _CORVAL[1]=1.
N100 _CORVAL[2]=1.
N110 T1 D1 G18 G0 X0 Y0 Z0                      ; ==> MKS-Position X1.5 Y5 Z4
N120 R1=SETTCOR(_CORVAL,"G",1,1)
N130 T1 D1 X0 Y0 Z0                             ; ==> MKS-Position X2.5 Y6 Z5
N140 R3=$TC_DP3[1,1]                            ; = 5. = (3.000 + 2.*1.000)
N150 R4=$TC_DP4[1,1]                            ; = 5. = (4.000 + 1.000)
N160 R5=$TC_DP5[1,1]                            ; = 6. = (5.000 + 1.000)
N170 M30
```

Die Korrektur der Werkzeuglänge soll in jeder Achse 1 mm sein (N80 bis N100). In den Längen L2 und L3 wird deshalb zur Originallänge jeweils 1 mm addiert. Zur Originalwerkzeuglänge in L1 wird dagegen der doppelte Korrekturwert (2 mm) addiert, damit sich die Gesamtwerkzeuglänge wie verlangt um 1 mm ändert. Im Vergleich der angefahrenen Positionen in den Sätzen N110 und N130 erkennt man, dass sich jede Achsposition um 1 mm verändert hat.

2.17 Zuordnung der Werkzeuglängen L1, L2, L3 zu den Koordinatenachsen lesen (LENTOAX)

Die Funktion LENTOAX liefert Informationen über die Zuordnung der Werkzeuglängen L1, L2 und L3 des **aktiven** Werkzeugs zu Abszisse, Ordinate und Applikate. Die Zuordnung von Abszisse, Ordinate und Applikate zu den Geometrieachsen wird durch Frames und die aktive Ebene (G17 - G19) beeinflusst.

Betrachtet wird dabei nur der Geometrieanteil eines Werkzeugs ($TC_DP3[<t>,<d>] bis $TC_DP5[<t>,<d>]), d. h., eine gegebenenfalls davon abweichende Achszuordnung weiterer Komponenten (z. B. Verschleiß) hat auf das Ergebnis keinen Einfluss.

Syntax

```
<Status> = LENTOAX(<AxInd>, <Matrix>[, <Coord>])
```

Prinzip

LENTOAX(...):	Vordefinierte Funktion zum Lesen der Zuordnung der Werkzeuglängen L1, L2 und L3 des aktiven Werkzeugs zu den Koordinatenachsen		
	Alleine im Satz:		ja
<Status>:	Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.		
	Datentyp:		INT
	Wert:	0	Funktion OK Information in <AxInd> reicht zur Beschreibung aus (alle Werkzeuglängenkomponenten sind parallel zu Geometrieachsen).
		1	Funktion ist OK, zu einer korrekten Beschreibung muss aber der Inhalt von <Matrix> ausgewertet werden (die Werkzeuglängenkomponenten sind nicht parallel zu den Geometrieachsen).
		-1	Ungültiger String im Parameter <Coord>.
		-2	Kein Werkzeug aktiv.
Parameter			
1 <AxInd>:	Sind die Werkzeuglängenkomponenten parallel zu den Geometrieachsen, werden die Achsindizes, die den Längenkomponenten L1 bis L3 zugeordnet sind, im Feld <AxInd> zurückgeliefert: • <AxInd> [0]: Abszisse • <AxInd> [1]: Ordinate • <AxInd> [2]: Applikate		
	Datentyp:		INT[3]
	Wert:	0	Keine Zuordnung vorhanden (Achse existiert nicht)
		1 ... 3 bzw. -1 ... -3	Nummer der Länge, die in der entsprechende Koordinatenachse wirkt. Das Vorzeichen ist negativ, wenn die Werkzeuglängenkomponente in negative Koordinatenrichtung zeigt.
	Sind nicht alle Längenkomponenten parallel bzw. antiparallel zu den Geometrieachsen, wird in <AxInd> jeweils der Index der Achse zurückgegeben, die den größten Anteil einer Werkzeuglängenkomponente enthält. In diesem Fall (falls die Funktion nicht aus einem anderen Grund einen Fehler liefert) ist der Rückgabewert <Status> = 1. Die Abbildung der Werkzeuglängenkomponenten L1 bis L3 auf die Geometrieachsen 1 bis 3 wird dann durch den Inhalt des 2. Parameters <Matrix> vollständig beschrieben.		
2 <Matrix>:	Matrix, die den Vektor der Werkzeuglängen (L1=1, L2=1, L3=1) in den Vektor der Koordinatenachsen (Abszisse, Ordinate, Applikate) abbildet, d. h., den Spalten sind die Werkzeuglängenkomponenten in der Reihenfolge L1, L2, L3 zugeordnet, den Zeilen die Achsen in der Reihenfolge Abszisse, Ordinate, Applikate.		
	Datentyp:		REAL
	In der Matrix sind immer alle Elemente gültig, auch dann, wenn die zu einer Koordinatenachse gehörende Geometrieachse nicht vorhanden ist, d. h. wenn der entsprechende Eintrag in <AxInd> null ist.		

3	<Coord>:	Koordinatensystem, für das die Zuordnung gilt (optional)		
		Datentyp:		STRING
		Zeichen:	MCS M	Abbildung der Werkzeuglänge in das Maschinenkoordinatensystem
			BCS B	Abbildung der Werkzeuglänge in das Basiskoordinatensystem
			WCS W	Abbildung der Werkzeuglänge in das Werkstückkoordinatensystem (Default)
			KCS K	Abbildung der Werkzeuglänge in das Werkzeugkoordinatensystem der kinematischen Transformation
			TCS T	Abbildung der Werkzeuglänge in das Werkzeugkoordinatensystem
		Die Schreibweise der Zeichen im String (groß oder klein) ist beliebig. Wird der Parameter <Coord> nicht angegeben, wird das WKS verwendet (Default).		

Hinweis

Im TCS sind immer alle Werkzeuglängenkomponenten parallel oder antiparallel zu den Achsen.

Antiparallel können die Komponenten nur dann sein, wenn Spiegeln aktiv ist und das folgende Settingdatum gesetzt ist:

SD42900 $SC_MIRROR_TOOL_LENGTH (Vorzeichenwechsel Werkzeuglänge beim Spiegeln)

Beispiel

Standardfall Fräswerkzeug bei G17.

L1 wirkt in Z (Applikate), L2 wirkt in Y (Ordinate), L3 wirkt in X (Abszisse).

Aufruf der Funktion in der Form:
```
<Status>=LENTOAX(<AxInd>,<Matrix>,"WCS")
```

Der Ergebnisparameter <AxInd> enthält dann die Werte:

<AxInd>[0] = 3

<AxInd>[1] = 2

<AxInd>[2] = 1

oder kurz: (3, 2, 1)

Die zugehörige Matrix (<Matrix>) ist in diesem Fall:

$$\text{<Matrix>} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

Ein Wechsel von G17 nach G18 oder G19 ändert am Ergebnis nichts, da die Zuordnung der Längenkomponenten zu den Geometrieachsen sich in gleicher Weise ändert wie die Zuordnung von Abszisse, Ordinate und Applikate.

Es wird nun bei aktivem G17 eine Framedrehung um Z von 60 Grad programmiert, z. B.:
`ROT Z60`

Die Richtung der Applikate (Z-Richtung) bleibt unverändert, der Hauptanteil von L2 liegt nun in Richtung der neuen X-Achse, der Hauptanteil von L1 in Richtung der negativen Y-Achse. Der Rückgabewert (<Status>) ist deshalb "1", <AxInd> enthält die Werte (2, -3, 1).

Die zugehörige Matrix (<Matrix>) ist in diesem Fall:

$$\text{<Matrix>} = \begin{pmatrix} 0 & \sin 60° & \cos 60° \\ 0 & \cos 60° & -\sin 60° \\ 1 & 0 & 0 \end{pmatrix}$$

2.18 Randbedingungen

2.18.1 SD42935 Erweiterungen

Transformation der Verschleißwerte

Welche der Verschleißkomponenten im Zusammenhang mit den Anweisungen `TOWMCS` (Verschleißwerte im Maschinen-Koordinatensystem) und `TOWWCS` (Verschleißwerte im Werkstück-Koordinatensystem) transformiert bzw. nicht transformiert werden sollen, kann festgelegt werden durch:

SD42935 $SC_WEAR_TRANSFORM (Transformation der Verschleißwerte)

2.18.2 Ankratzen

Ankratzen

Wird die Funktion "Ankratzen" im Reset-Zustand verwendet, so ist bezüglich des Grundstellungswerts Folgendes zu beachten:

- Für die Bewertung der Verschleißkomponenten ist der Grundstellungswert der G-Gruppe TOWSTD, TOWMCS und TOWWCS maßgebend.
- Wird zur korrekten Berechnung ein anderer als der Grundstellungswert benötigt, darf das Ankratzen nur im Stopp-Zustand erfolgen.

2.19 Datenlisten

2.19.1 Maschinendaten

2.19.1.1 NC-spezifische Maschinendaten

Nummer	Bezeichner: $MN_	Beschreibung
18082	MM_NUM_TOOL	Anzahl der Werkzeuge, die NC verwalten kann (SRAM)
18088	MM_NUM_TOOL_CARRIER	Maximale Anzahl der definierter Werkzeugträger
18094	MM_NUM_CC_TDA_PARAM	Anzahl der Werkzeugdaten (SRAM)
18096	MM_NUM_CC_TOA_PARAM	Anzahl der Daten, pro Werkzeugschneide für Compile-zyklen (SRAM)
18100	MM_NUM_CUTTING_EDGES_IN_TOA	Werkzeugkorrekturen im TOA-Bereich (SRAM)
18102	MM_TYPE_OF_CUTTING_EDGE	Art der D-Nummer Programmierung (SRAM)
18105	MM_MAX_CUTTING_EDGE_NO	Maximaler Wert der D-Nummer
18106	MM_MAX_CUTTING_EDGE_PERTOOL	Maximale Anzahl der D-Nummern pro Werkzeug
18108	MM_NUM_SUMCORR	Anzahl aller Summenkorrekturen im NC
18110	MM_MAX_SUMCORR_PER_CUTTEDGE	Anzahl von Summenkorrekturen pro Schneide
18112	MM_KIND_OF_SUMCORR	Eigenschaften der Summenkorrekturen im TO-Bereich (SRAM)
18114	MM_ENABLE_TOOL_ORIENT	Werkzeugschneiden Orientierung zuordnen
18116	MM_NUM_TOOL_ENV	Werkzeugumgebungen im TO-Bereich (SRAM)

2.19.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: $MC_	Beschreibung
20096	T_M_ADDRESS_EXT_IS_SPINO	Bedeutung der Adresserweiterung bei T, M Werkzeug-wechsel
20110	RESET_MODE_MASK	Festlegung der Steuerungs-Grundstellung nach Reset/ Teileprogrammende
20120	TOOL_RESET_VALUE	Werkzeug Längenkorrekturim Hochlauf (Reset/TP-En-de)
20121	TOOL_PRESEL_RESET_VALUE	Vorgewähltes Werkzeug bei Reset
20125	CUTMOD_ERR	Fehlerbehandlung für die Funktion "Modifikation der Korrekturdaten bei drehbaren Werkzeugen"
20126	TOOL_CARRIER_RESET_VALUE	Wirksamer Werkzeugträger bei Reset
20127	CUTMOD_INIT	Initialisierungswert für CUTMOD
20130	CUTTING_EDGE_RESET_VALUE	Werkzeugschneide Längenkorrektur im Hochlauf (Re-set/TP-Ende)
20132	SUMCORR_RESET_VALUE	Wirksame Summenkorrektur bei Reset

Nummer	Bezeichner: $MC_	Beschreibung
20140	TRAFO_RESET_VALUE	Transformationsdatensatz im Hochlauf (Reset/TP-Ende)
20180	TOCARR_ROT_ANGLE_INCR[i]	Rundachseninkrement des orientierbaren Werkzeugträgers
20182	TOCARR_ROT_ANGLE_OFFSET[i]	Rundachsoffset des orientierbaren Werkzeugträgers
20184	TOCARR_BASE_FRAME_NUMBER	Nummer des Basisframes für Aufnahme des Tischoffsets
20188	TOCARR_FINE_LIM_LIN	Limit ineare Feinverschiebung TCARR
20190	TOCARR_FINE_LIM_ROT	Limit der rotatorische Feinverschiebung TCARR
20202	WAB_MAXNUM_DUMMY_BLOCKS	Max. Satzanzahl ohne Verfahrbewegung bei WAB
20204	WAB_CLEARANCE_TOLERANCE	Richtungsumkehr bei WAB
20210	CUTCOM_CORNER_LIMIT	Maximalwinkel für Ausgleichssätze bei WRK
20220	CUTCOM_MAX_DISC	Maximaler Wert für DISC
20230	CUTCOM_CURVE_INSERT_LIMIT	Maximalwert für Schnittpunktberechnung bei WRK
20240	CUTCOM_MAXNUM_CHECK_BLOCKS	Sätze für vorausschauende Konturberechnung bei WRK
20250	CUTCOM_MAXNUM_DUMMY_BLOCKS	Maximale Satzanzahl ohne Verfahrbewegung bei WRK
20252	CUTCOM_MAXNUM_SUPPR_BLOCKS	Maximale Satzzahl mit Korrekturunterdrückung
20256	CUTCOM_INTERS_POLY_ENABLE	Schnittpunktverfahren für Polynome möglich
20270	CUTTING_EDGE_DEFAULT	Grundstellung der Werkzeugschneide ohne Programmierung
20272	SUMCORR_DEFAULT	Grundstellung Summenkorrektur ohne Programm
20360	TOOL_PARAMETER_DEF_MASK	Definition der Werkzeug-Parameter
20390	TOOL_TEMP_COMP_ON	Aktivierung der Temperaturkompensation für Werkzeuglänge
20392	TOOL_TEMP_COMP_LIMIT	Maximale Temperaturkompensation für Werkzeuglänge
20610	ADD_MOVE_ACCEL_RESERVE	Beschleunigungsreserve für überlagerte Bewegungen
21080	CUTCOM_PARALLEL_ORI_LIMIT	Minimaler Winkel (Bahntangente und Werkzeugorientierung) bei 3D-Werkzeugradiuskorrektur
22530	TOCARR_CHANGE_M_CODE	M-Code bei Werkzeugträgerwechsel
22550	TOOL_CHANGE_MODE	Neue WZK bei M-Funktion
22560	TOOL_CHANGE_M_CODE	M-Funktion für Werkzeugwechsel
22562	TOOL_CHANGE_ERROR_MODE	Verhalten bei Fehlern im Werkzeugwechsel
24558	TRAFO5_JOINT_OFFSET_PART_1	Vektor kinematischer Versatz im Tisch, Trafo 1
24658	TRAFO5_JOINT_OFFSET_PART_2	Vektor kinematischer Versatz im Tisch, Trafo 2
28085	MM_LINK_TOA_UNIT	Zuordnung der TO-Einheit zu einem Kanal (SRAM)

2.19.1.3 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: $MA_	Beschreibung
32750	TEMP_COMP_TYPE	Temperaturkompensationstyp

2.19.2 Settingdaten

2.19.2.1 Kanal-spezifische Settingdaten

Nummer	Bezeichner: $SC_	Beschreibung
42442	TOOL_OFFSET_INCR_PROG	Werkzeuglängenkorrekturen
42470	CRIT_SPLINE_ANGLE	Ecken-Grenzwinkel für Kompressor
42480	STOP_CUTCOM_STOPRE	Alarmreaktion bei Werkzeugradiuskorrektur und Vorlaufstopp
42494	CUTCOM_ACT_DEACT_CTRL	An-/Abfahrverhalten bei Wz.-Radiuskorrektur
42496	CUTCOM_CLSDT_CONT	Verhalten der Werkzeugradiuskorrektur bei geschlossener Kontur
42900	MIRROR_TOOL_LENGTH	Vorzeichenwechsel Werkzeuglängen beim Spiegeln
42910	MIRROR_TOOL_WEAR	Vorzeichenwechsel Werkzeugverschleiß beim Spiegeln
42920	WEAR_SIGN_CUTPOS	Vorzeichen des Verschleißes bei Werkzeugen mit Schneidenlage
42930	WEAR_SIGN	Vorzeichen des Verschleißes
42935	WEAR_TRANSFORM	Transformationen für Werkzeugkomponenten
42940	TOOL_LENGTH_CONST	Wechsel der Werkzeuglängenkomponenten bei Ebenenwechsel
42950	TOOL_LENGTH_TYPE	Zuordnung der Werkzeuglängenkompensation unabhängig vom Werkzeugtyp
42960	TOOL_TEMP_COMP	Temperaturkompensationswert bezogen auf das Werkzeug
42974	TCARR_FINE_CORRECTION	Feinverschiebung `TCARR` ein/aus
42984	CUTDIRMOD	Modifikation von $P_AD[2] bzw. $P_AD[11]
42998	CUTMOD_PLANE_TOL	Abweichung Schneidplatten-/Bearbeitungsebene bei CUTMOD bzw. CUTMODK
42999	ORISOLH_INCLINE_TOL	Abweichung Schneidplatten-/ Bearbeitungsebene bei ORISOLH

2.19.2.2 Kanal-spezifische Settingdaten

Nummer	Bezeichner: $SC_	Beschreibung
42442	TOOL_OFFSET_INCR_PROG	Werkzeuglängenkorrekturen
42470	CRIT_SPLINE_ANGLE	Ecken-Grenzwinkel für Kompressor
42480	STOP_CUTCOM_STOPRE	Alarmreaktion bei Werkzeugradiuskorrektur und Vorlaufstopp
42494	CUTCOM_ACT_DEACT_CTRL	An-/Abfahrverhalten bei Wz.-Radiuskorrektur
42496	CUTCOM_CLSDT_CONT	Verhalten der Werkzeugradiuskorrektur bei geschlossener Kontur
42900	MIRROR_TOOL_LENGTH	Vorzeichenwechsel Werkzeuglängen beim Spiegeln

Nummer	Bezeichner: $SC_	Beschreibung
42910	MIRROR_TOOL_WEAR	Vorzeichenwechsel Werkzeugverschleiß beim Spiegeln
42920	WEAR_SIGN_CUTPOS	Vorzeichen des Verschleißes bei Werkzeugen mit Schneidenlage
42930	WEAR_SIGN	Vorzeichen des Verschleißes
42935	WEAR_TRANSFORM	Transformationen für Werkzeugkomponenten
42940	TOOL_LENGTH_CONST	Wechsel der Werkzeuglängenkomponenten bei Ebenenwechsel
42950	TOOL_LENGTH_TYPE	Zuordnung der Werkzeuglängenkompensation unabhängig vom Werkzeugtyp
42960	TOOL_TEMP_COMP	Temperaturkompensationswert bezogen auf das Werkzeug
42974	TCARR_FINE_CORRECTION	Feinverschiebung `TCARR` ein/aus
42984	CUTDIRMOD	Modifikation von $P_AD[2] bzw. $P_AD[11]
42998	CUTMOD_PLANE_TOL	Abweichung Schneidplatten-/Bearbeitungsebene bei CUTMOD

W4: Schleifspezifische Werkzeugkorrektur und -überwachung

3

3.1 Schleifspezifische Werkzeugdaten

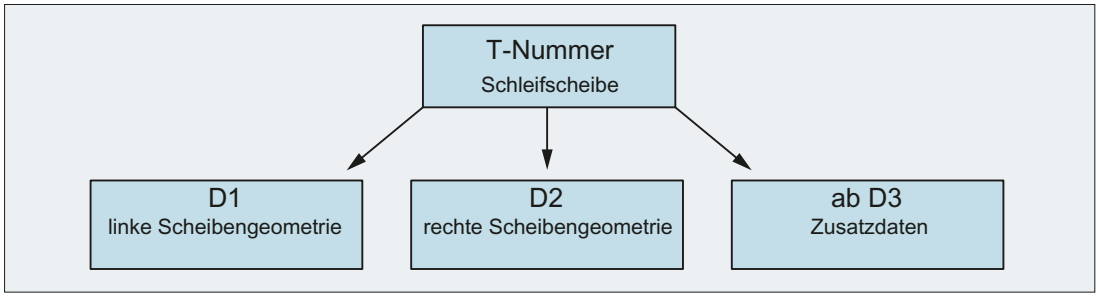
3.1.1 Struktur der Werkzeugdaten

Schleifwerkzeuge (WZ-Typ: 400 bis 499) besitzen in der Regel neben schneidenspezifischen auch werkzeug- und abrichterspezifische Daten.

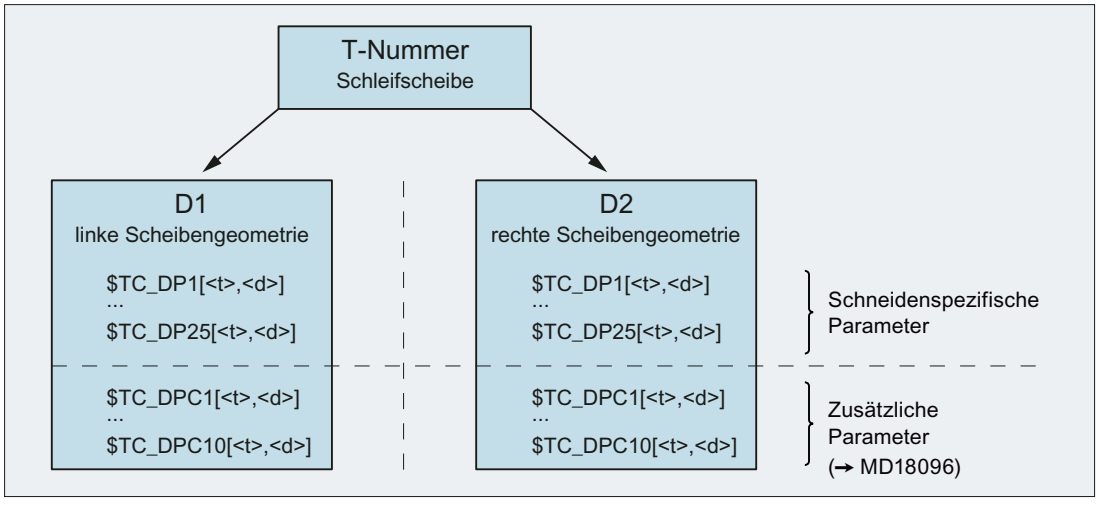
Unter einer T-Nummer können die schleifscheibenspezifischen Daten für die linke und rechte Scheibengeometrie in den WZ-Schneiden D1 und D2 abgelegt werden.

Werden Daten für die Abrichtergeometrie benötigt, können sie wie folgt hinterlegt werden:

- in den WZ-Schneiden ab D3 einer T-Nummer



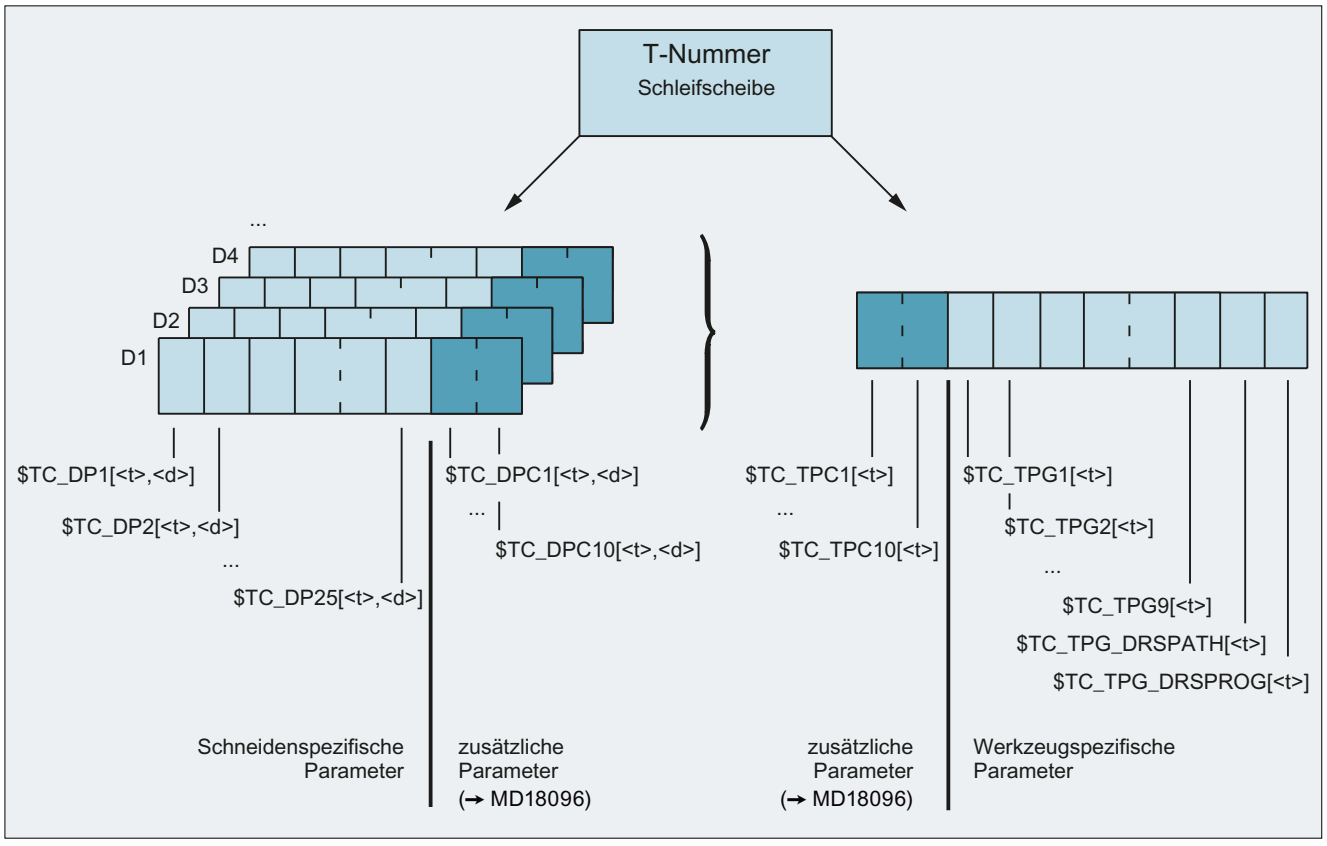
- in zusätzlichen schneidenspezifischen Parametern (siehe MD18096 $MN_MM_NUM_CC_TOA_PARAM)



<t>: T-Nummer

<d>: D-Nummer

Die folgende Abbildung gibt eine Übersicht über die (mögliche) Ablagestruktur der Werkzeugdaten bei Schleifwerkzeugen:



<t>: T-Nummer

<d>: D-Nummer

Bild 3-1 Struktur der Werkzeugdaten bei Schleifwerkzeugen

3.1.2 Schneidenspezifische Parameter

3.1.2.1 Liste der schneidenspezifischen Parameter

Die schneidenspezifischen Werkzeugparameter haben für Schleifwerkzeuge die gleiche Bedeutung wie für Dreh- und Fräswerkzeuge.

WZ-Parameter	Bedeutung	Bemerkung	
$TC_DP1	Werkzeugtyp		
$TC_DP2	WZ-Schneidenlage	nur für Dreh- und Schleifwerkzeuge	
Geometrie - Werkzeuglängenkorrektur			

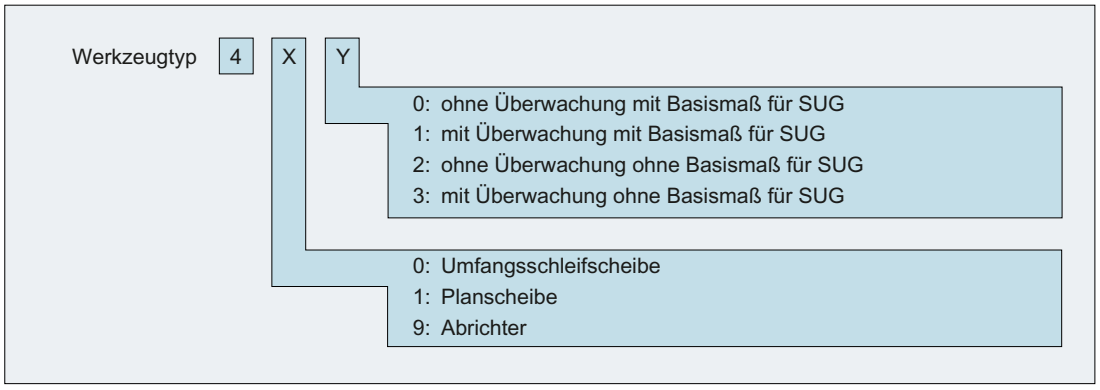
WZ-Parameter	Bedeutung	Bemerkung	
$TC_DP**3**	Länge 1		
$TC_DP**4**	Länge 2		
$TC_DP**5**	Länge 3		
Geometrie - Werkzeugradiuskorrektur			
$TC_DP**6**	Radius 1		
$TC_DP**7**			reserviert
$TC_DP**8**			reserviert
$TC_DP**9**			reserviert
$TC_DP**10**			reserviert
$TC_DP**11**			reserviert
Verschleiß - Werkzeuglängenkorrektur			
$TC_DP**12**	Länge 1		
$TC_DP**13**	Länge 2		
$TC_DP**14**	Länge 3		
Verschleiß - Werkzeugradiuskorrektur			
$TC_DP**15**	Radius 1		
$TC_DP**16**			reserviert
$TC_DP**17**			reserviert
$TC_DP**18**			reserviert
$TC_DP**19**			reserviert
$TC_DP**20**			reserviert
Basismaß/Adaptermaß - Werkzeuglängenkorrektur			
$TC_DP**21**	Basis-Länge 1		
$TC_DP**22**	Basis-Länge 2		
$TC_DP**23**	Basis-Länge 3		
Technologie			
$TC_DP**24**	Freischneidwinkel	nur für Drehwerkzeuge	
$TC_DP**25**			reserviert

Hinweis

Die Schneidendaten für D1 und D2 eines gewählten Schleifwerkzeugs können verkettet werden, d. h., wird ein Parameter in D1 bzw. D2 geändert, wird automatisch der gleiche Parameter in D1 bzw. D2 mit dem neuen Wert überschrieben (siehe $TC_TPG2 (Seite 250)).

3.1.2.2 $TC_DP1

Der Parameter $TC_DP1 enthält die 3-stellige Nummer des Schleifwerkzeugtyps:



Nummer	Schleifwerkzeugtyp
400	Umfangsschleifscheibe
401	Umfangsschleifscheibe mit Überwachung mit Basismaß für SUG
402	Umfangsschleifscheibe ohne Überwachung ohne Basismaß für SUG
403	Umfangsschleifscheibe mit Überwachung ohne Basismaß für SUG
410	Planscheibe
411	Planscheibe mit Überwachung mit Basismaß für SUG
412	Planscheibe ohne Überwachung ohne Basismaß für SUG
413	Planscheibe mit Überwachung ohne Basismaß für SUG
490	Abrichter

3.1.2.3 Definition zusätzlicher Parameter $TC_DPC1 ... 10

Für anwenderbezogene Schneidendaten können die zusätzlichen Parameter $TC_DPC1 bis $TC_DPC10 unabhängig vom Werkzeugtyp über das folgende speicherkonfigurierende Maschinendatum eingerichtet werden:

MD18096 $MN_MM_NUM_CC_TOA_PARAM

ACHTUNG
Datenverlust durch Umkonfiguration Eine Umkonfiguration des lokalen statischen NC-Speichers hat den Verlust der Daten des aktiven und passiven Filesystems zur Folge. Es wird daher dringend empfohlen, vor Aktivierung einer geänderten Speicherkonfiguration alle relevanten Daten durch Erstellen eines Inbetriebnahmearchivs zu archivieren/sichern.

3.1.3 Werkzeugspezifische Parameter

3.1.3.1 Liste der werkzeugspezifischen Parameter

Die werkzeugspezifischen Parameter werden automatisch mit jedem neuen Schleifwerkzeug (WZ-Typ: 400 bis 499) angelegt.

Hinweis

Die werkzeugspezifischen Parameter verhalten sich wie eine Schneide.

Bei der Angabe der Anzahl Schneiden ist dies gegebenenfalls zu berücksichtigen:

MD18100 $MN_MM_NUM_CUTTING_EDGES_IN_TOA

Mit dem Löschen aller Schneiden eines Werkzeugs werden auch automatisch die zugehörigen werkzeugspezifischen Parameter gelöscht.

Parameter

Die Parameter sind wie folgt belegt:

Parameter	Bedeutung	Datentyp
$TC_TPG1 (Seite 250)	Spindelnummer	INT
$TC_TPG2 (Seite 250)	Verkettungsvorschrift	INT
$TC_TPG3 (Seite 252)	Minimaler Scheibenradius	REAL
$TC_TPG4 (Seite 252)	Minimale Scheibenbreite	REAL
$TC_TPG5 (Seite 252)	Aktuelle Scheibenbreite	REAL
$TC_TPG6 (Seite 252)	Maximale Drehzahl	REAL
$TC_TPG7 (Seite 252)	Maximale Umfangsgeschwindigkeit	REAL
$TC_TPG8 (Seite 252)	Winkel der schrägen Scheibe	REAL
$TC_TPG9 (Seite 253)	Parameter-Nummer für Radiusberechnung	INT
$TC_TPG_DRSPATH (Seite 254)	Verzeichnispfad auf das Abrichtprogramm	STRING
$TC_TPG_DRSPROG (Seite 254)	Abrichtprogrammname	STRING

Hinweis

Die Überwachungsdaten gelten sowohl für die linke als auch für die rechte Schneide der Schleifscheibe.

Wirksam werden die werkzeugspezifischen Parameter:

- beim Einschalten der schleifspezifischen Werkzeugüberwachung (Seite 269)
- beim Einschalten der konstanten Scheibenumfangsgeschwindigkeit (SUG) (Seite 272)

Soll ein geändertes Datum wirksam werden, müssen diese Funktionen erneut eingeschaltet werden.

Die Längenkorrekturen geben in jedem Fall die Abstände zwischen Werkzeugträgerbezugspunkt und Werkzeugspitze in den kartesischen Koordinaten an (bei schräger Scheibe ist dies zu beachten).

3.1.3.2 $TC_TPG1

In den Parameter $TC_TPG1 wird die Spindelnummer eingetragen. Für diese Spindel wird dann die Scheibenumfangsgeschwindigkeit überwacht.

3.1.3.3 $TC_TPG2

Im Parameter $TC_TPG2 werden die WZ-Parameter der beiden Schneiden D1 und D2 eingetragen, deren Werte immer gleich sein müssen. Wird bei einer Schneide der Wert eines dieser Parameter geändert, wird der entsprechende Parameter der anderen Schneide automatisch auf den gleichen Wert gesetzt.

Die WZ-Parameter werden im Parameter $TC_TPG2 codiert angegeben:

WZ-Parameter	Bedeutung	Codierung		
		Bit	Hex	Dez
$TC_DP1	Werkzeugtyp	0	0001	1
$TC_DP2	Schneidenlage	1	0002	2
Geometrie - Werkzeuglängenkorrektur				
$TC_DP3	Länge 1	2	0004	4
$TC_DP4	Länge 2	3	0008	8
$TC_DP5	Länge 3	4	0010	16
$TC_DP6	Radius	5	0020	32
$TC_DP7	reserviert	6	0040	64
$TC_DP8	Länge 4	7	0080	128
$TC_DP9	Länge 5	8	0100	256
$TC_DP10	Winkel 1 etc.	9	0200	512
$TC_DP11	reserviert	10	0400	1 024
Verschleiß - Werkzeuglängenkorrektur				
$TC_DP12	Länge 1	11	0800	2 048
$TC_DP13	Länge 2	12	1000	4 096

WZ-Parameter	Bedeutung	Codierung		
		Bit	Hex	Dez
$TC_DP14	Länge 3	13	2000	8 192
$TC_DP15	Radius	14	4000	16 384
$TC_DP16	reserviert	15	8000	32 768
$TC_DP17	Verschleiß Länge 4	16	1 0000	65 536
$TC_DP18	Verschleiß Länge 5	17	2 0000	131 072
$TC_DP19	Verschleiß Winkel 1 etc.	18	4 0000	262 144
$TC_DP20	reserviert	19	8 0000	524 288
Basismaß/Adaptermaß - Werkzeuglängenkorrektur				
$TC_DP21	Basis-Länge 1	20	10 0000	1 048 576
$TC_DP22	Basis-Länge 2	21	20 0000	2 097 152
$TC_DP23	Basis-Länge 3	22	40 0000	4 194 304
Technologie				
$TC_DP24	reserviert	23	80 0000	8 388 608
$TC_DP25	reserviert	24	100 0000	16 777 216

Wirksamwerden einer Änderung

Wenn der Parameter $TC_TPG2 zu einem späteren Zeitpunkt geändert wird, werden die Werte der WZ-Parameter der beiden Schneiden nicht sofort automatisch abgeglichen. Der Abgleich erfolgt erst mit der nächsten Änderung eines WZ-Parameters der beiden Schneiden.

Beispiel

Bei einem Schleifwerkzeug (T1) sollen für die Schneiden D1 un d D2 folgende Parameter gleich gehalten werden:

WZ-Parameter	Bedeutung	Codierung: Hex	Codierung: Dezmal
$TC_DP1	Werkzeugtyp	0001	1
Geometrie - Werkzeuglängenkorrektur			
$TC_DP3	Länge 1	0004	4
$TC_DP4	Länge 2	0008	8
$TC_DP5	Länge 3	0010	16
Verschleiß - Werkzeuglängenkorrektur			
$TC_DP12	Länge 1	0800	2 048
$TC_DP13	Länge 2	1000	4 096
$TC_DP14	Länge 3	2000	8 192
Basismaß/Adaptermaß - Werkzeuglängenkorrektur			
$TC_DP21	Basis-Länge 1	10 0000	1 048 576
$TC_DP22	Basis-Länge 2	20 0000	2 097 152
$TC_DP23	Basis-Länge 3	40 0000	4 194 304

In den Parameter $TC_TPG2 wird die Summe aller Parameter-Codierungen eingetragen:

$TC_TPG2 =	'H70381D'	'D7354397'

3.1.3.4 $TC_TPG3, $TC_TPG4

Für die schleifspezifische Werkzeugüberwachung (Seite 269) werden in die Parameter die jeweiligen **unteren** Grenzwerte der Schleifscheibe eingetragen:

- Parameter $TC_TPG3: Radius [mm] bzw. [inch]
- Parameter $TC_TPG4: Breite [mm] bzw. [inch]

Hinweis

Bei schrägstehender Schleifscheibe muss der minimale Scheibenradius in kartesischen Koordinaten angegeben werden.

3.1.3.5 $TC_TPG5

In den Parameter $TC_TPG5 wird die aktuelle Ist-Breite der Schleifscheibe, z. B. nach dem Abrichten, eingetragen.

3.1.3.6 $TC_TPG6 und $TC_TPG7

Für die schleifspezifische Werkzeugüberwachung (Seite 269) werden in die Parameter die jeweiligen **oberen** Grenzwerte der Schleifscheibe eingetragen:

- Parameter $TC_TPG6: Drehzahl [$min^{-1}$]
- Parameter $TC_TPG7: Umfangsgeschwindigkeit [$m*s^{-1}$] bzw. bzw. [$inch*s^{-1}$]

3.1.3.7 $TC_TPG8

Für die Funktion "Konstante Scheibenumfangsgeschwindigkeit (SUG) (Seite 272)" wird bei schrägstehender Schleifscheibe in den Parameter $TC_TPG8 der Neigungswinkel α bezogen auf die aktuell aktive Ebene (Arbeitsebene `G17`, `G18`, `G19`) eingetragen.

Der zulässige Wertebereich des Neigungswinkels α beträgt: $-90° \leq \alpha < +90°$

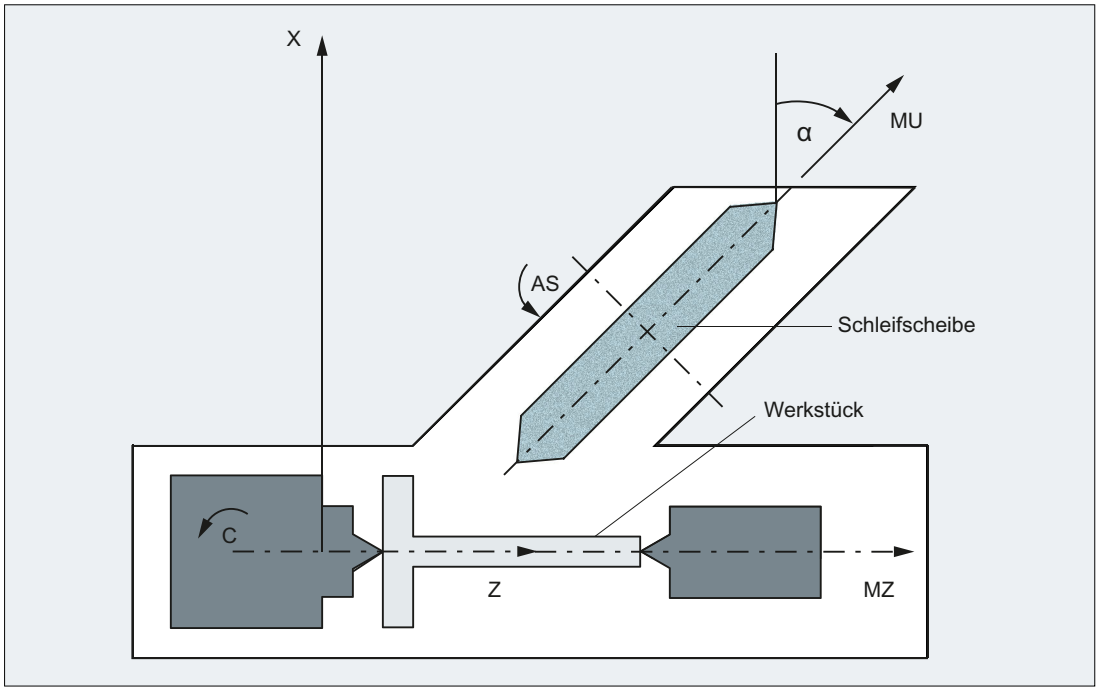


Bild 3-2 Schräg stehende Schleifscheibe

Hinweis

Bei einer Änderung der Winkelangabe im Parameter $TC_TPG8 erfolgt keine automatische Korrektur der Werkzeuglängen.

Hinweis

Bei Maschinen mit schrägstehender Achse muss der gleiche Winkel für die schräg stehene Achse wie für die schräg stehende Scheibe angegeben werden.

3.1.3.8 $TC_TPG9

Mit dem Parameter $TC_TPG9 wird festgelegt, auf welche Werkzeugparameter sich die folgenden Funktionen beziehen sollen:

- Schleifspezifische Werkzeugüberwachung (Seite 269)
- Konstante Scheibenumfangsgeschwindigkeit (SUG) (Seite 272)

Beispiele:

$TC_TPG9 = <Wert>

<Wert>	Bezugsgröße
3	Länge 1 (Geometrie + Verschleiß + Basis, abhängig vom WZ-Typ)
4	Länge 2 (Geometrie + Verschleiß + Basis, abhängig vom WZ-Typ)
5	Länge 3 (Geometrie + Verschleiß + Basis, abhängig vom WZ-Typ)
6	Radius

3.1.3.9 $TC_TPG_DRSPATH und $TC_TPG_DRSPROG

Die Parameter $TC_TPG_DRSPATH (Verzeichnispfad auf das Abrichtprogramm) und $TC_TPG_DRSPROG (Abrichtprogrammname) ermöglichen das Zuordnen eines Abrichtprogramms zu einer Schleifscheibe und erlauben damit das Erstellen einer allgemeingültigen Abrichtprozedur, in der die Ausführung des Abrichtprogramms durch einen indirekten Programmaufruf erfolgt.

Beispiel:

```
Programmcode                                          Kommentar
...
                                                      ; Abrichtbetrag in $TC_DP12 eingeben:
$TC_DP12[GETT("GRINDING_WHEEL",0),1]= $TC_DP12[GETT("GRINDING_WHEEL",0),1] - 0.01
G500
CYCLE435("GRINDING_WHEEL",1,"DRESSER",1)              ; Abrichterposition berechnen.
T="DRESSER" D1                                        ; Abrichter aktivieren.
                                                      ; Abrichtprogramm indirekt aufrufen:
CALL "/"<<$TC_TPG_DRSPATH[GETT("GRINDING_WHEEL",0)]<<"/"<<$ TC_TPG_DRSPROG[GETT("GRINDING_WHEEL",0)]
M17
```

3.1.3.10 Definition zusätzlicher Parameter $TC_TPC1 ... 10

Für anwenderbezogene Werkzeugdaten können die zusätzlichen Parameter $TC_TPC1 bis $TC_TPC10 unabhängig vom Werkzeugtyp über das folgende speicherkonfigurierende Maschinendatum eingerichtet werden:

MD18096 $MN_MM_NUM_CC_TDA_PARAM

ACHTUNG

Datenverlust durch Umkonfiguration

Eine Umkonfiguration des lokalen statischen NC-Speichers hat den Verlust der Daten des aktiven und passiven Filesystems zur Folge. Es wird daher dringend empfohlen, vor Aktivierung einer geänderten Speicherkonfiguration alle relevanten Daten durch Erstellen eines Inbetriebnahmearchivs zu archivieren/sichern.

3.1.3.11 Zugriff auf werkzeugspezifische Parameter

Vom Teileprogramm aus können werkzeugspezifische Parameter gelesen und geschrieben werden.

Beispiele:

```
Programmcode          Kommentar
R10=$TC_TPG5[2]       ; Lesen der aktuellen Breite von Werkzeug 2 und Ablegen in R10.
$TC_TPG6[3]=2000      ; Schreiben der maximalen Drehzahl von Werkzeug 3 mit dem Wert
                      2000.
```

Über die folgende Systemvariable kann auf die werkzeugspezifischen Parameter für das **aktuelle** Werkzeug zugegriffen werden:

$P_ATPG[<m>]

mit <m> = Parameter-Nummer (Datentyp: REAL)

Beispiel:

Parameter 3 ($TPG3[<t>])

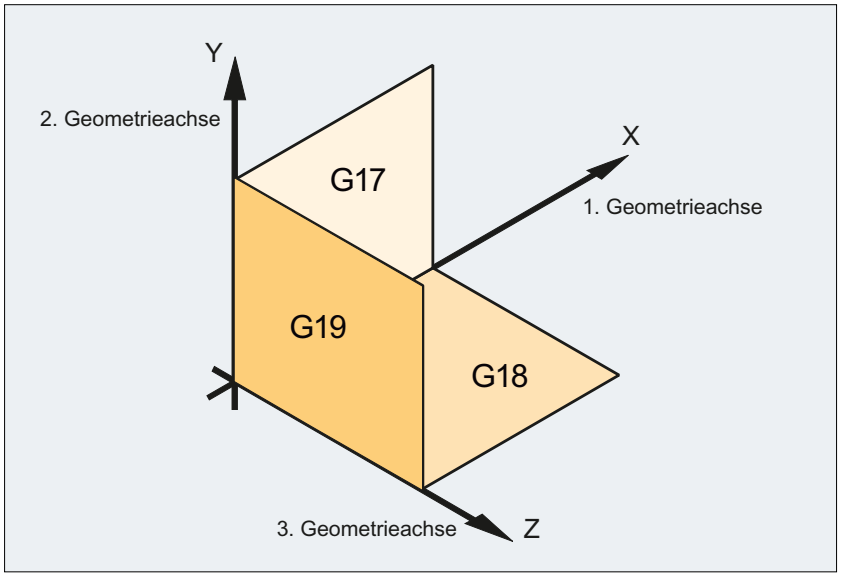
Programmcode	Kommentar
`$P_ATPG[3]=R10`	`; Lesen des minimalen Scheibenradius des aktuellen Werkzeugs und Ablegen in R10.`

3.1.4 Ebenen und Achszuordnungen

Die Zuordnung der Längenkorrekturen für die Geometrieachsen bzw. die Radiuskorrektur in der Ebene richtet sich nach der aktuellen Arbeitsebene.

Folgende Ebenen und Achszuordnungen sind möglich (Abszisse, Ordinate, Applikate zu 1., 2. und 3. Geometrieachse):

Befehl	Ebene (Abszisse / Ordinate)	Senkrechte Achse darauf (Applikate)
G17	X / Y	Z
G18	Z / X	Y
G19	Y / Z	X



3.1.5 Beispiele

Einträge in Werkzeugparameter

$TC_DP1	400
$TC_DP3	Länge 1
$TC_DP4	Länge 2
$TC_DP6	Radius

Die nicht verwendeten Parameter sind auf 0 zu setzen!

Wirkung	
G17	Länge 1 in Y Länge 2 in X Radius in X / Y
G18	Länge 1 in X Länge 2 in Z Radius in Z / X
G19	Länge 1 in Z Länge 2 in Y Radius in Y / Z

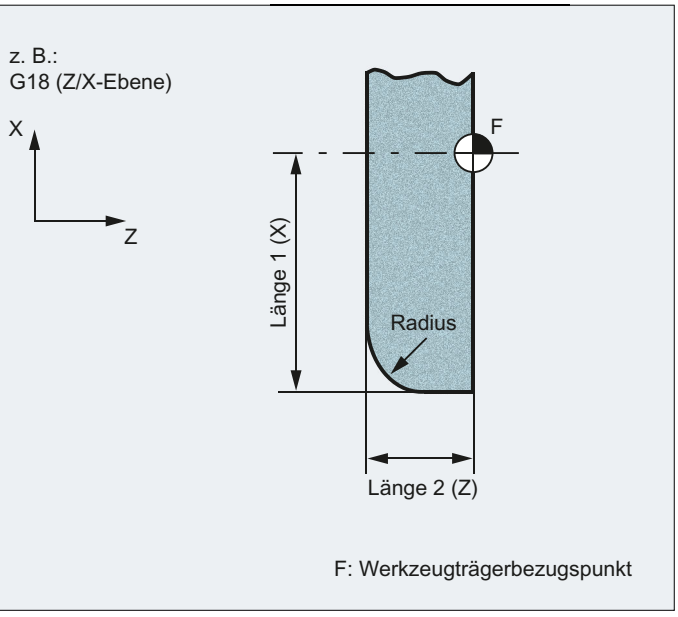


Bild 3-3 Erforderliche Korrekturdaten einer Umfangsschleifscheibe

Einträge in Werkzeugparameter

$TC_DP1	403
$TC_DP3	Länge 1
$TC_DP4	Länge 2
$TC_DP6	Radius

Die nicht verwendeten Parameter sind auf 0 zu setzen!

Wirkung	
G17	Länge 1 in Y Länge 2 in X Radius in X / Y
G18	Länge 1 in X Länge 2 in Z Radius in Z / X
G19	Länge 1 in Z Länge 2 in Y Radius in Y / Z

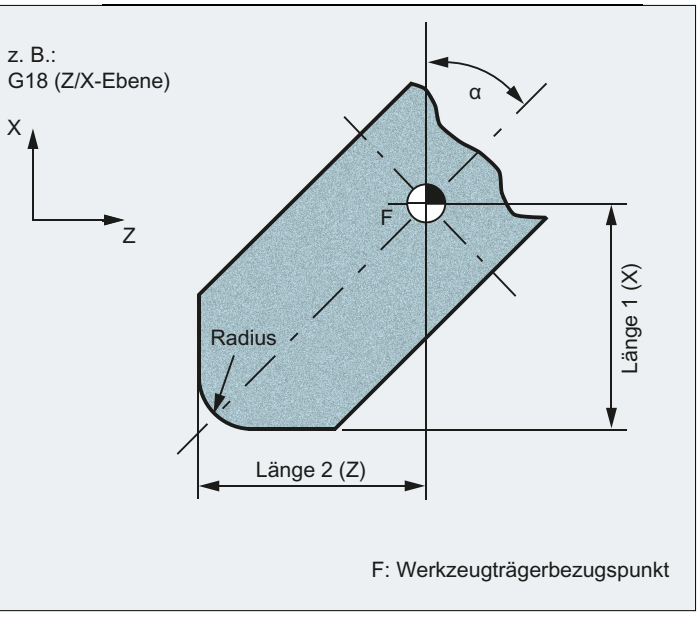


Bild 3-4 Erforderliche Korrekturdaten bei schräger Scheibe mit impliziter Überwachungsanwahl und ohne Basismaß für SUG

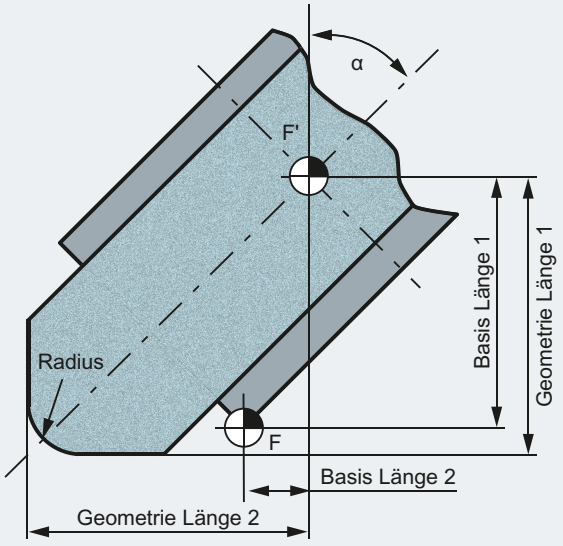
Einträge in Werkzeugparameter

$TC_DP1	401
$TC_DP3	Länge 1
$TC_DP4	Länge 2
$TC_DP6	Radius
$TC_DP21	L1 Basis
$TC_DP22	L2 Basis

Verschleißwerte entsprechend Erfordernis!

Die nicht verwendeten Parameter sind auf 0 zu setzen!

	Wirkung
G17	Länge 1 in Y Länge 2 in X Radius in X / Y
G18	Länge 1 in X Länge 2 in Z Radius in Z / X
G19	Länge 1 in Z Länge 2 in Y Radius in Y / Z



F: Werkzeugträgerbezugspunkt

Bild 3-5 Erforderliche Korrekturdaten bei schräger Scheibe mit impliziter Überwachungsanwahl und mit Basismaß für SUG

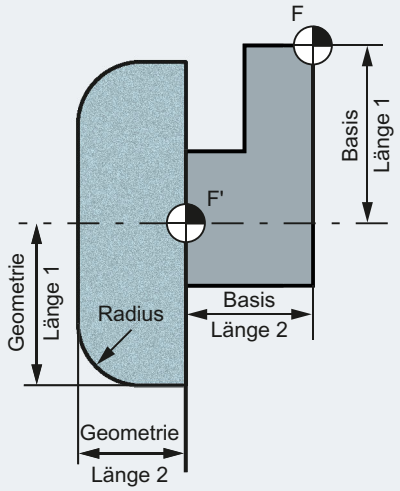
Einträge in Werkzeugparameter

$TC_DP1	403
$TC_DP3	Länge 1
$TC_DP4	Länge 2
$TC_DP6	Radius
$TC_DP21	L1 Basis
$TC_DP22	L2 Basis

Verschleißwerte entsprechend Erfordernis!

Die nicht verwendeten Parameter sind auf 0 zu setzen!

	Wirkung
G17	Länge 1 in Y Länge 2 in X Radius in X / Y
G18	Länge 1 in X Länge 2 in Z Radius in Z / X
G19	Länge 1 in Z Länge 2 in Y Radius in Y / Z



F: Werkzeugträgerbezugspunkt

Bild 3-6 Erforderliche Korrekturdaten einer Umfangsschleifscheibe ohne Basismaß für SUG

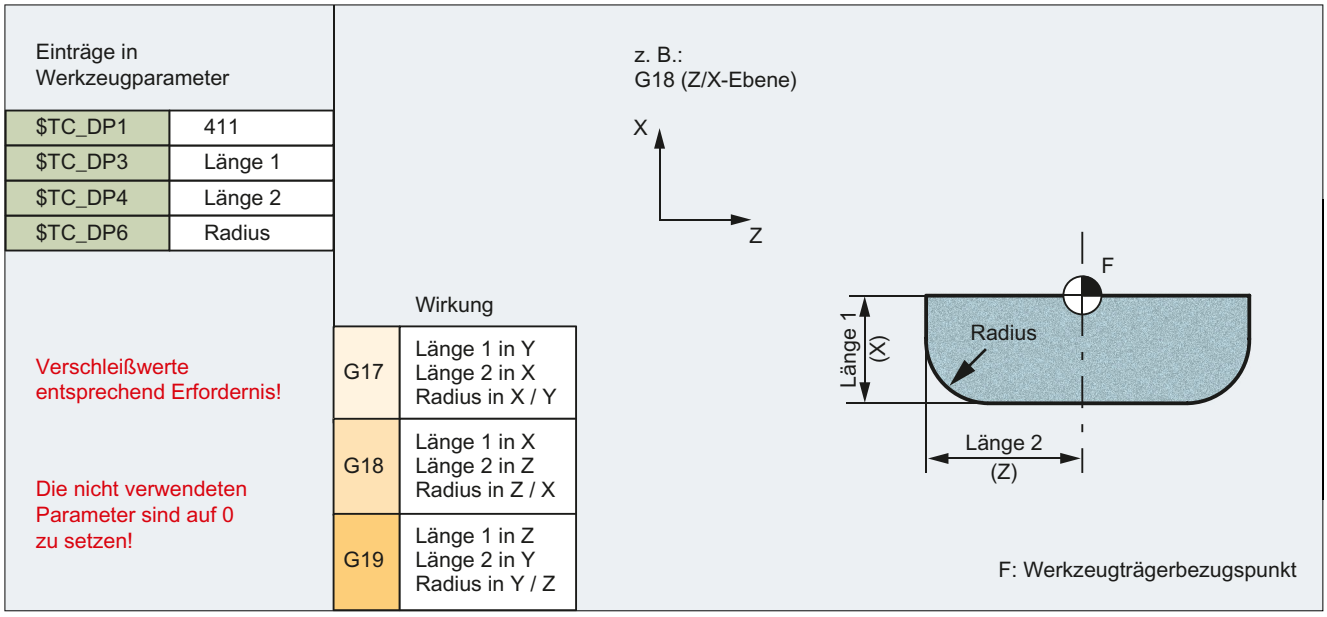


Bild 3-7 Erforderliche Korrekturdaten einer Planscheibe mit Überwachungsparametern

3.2 Online-Werkzeugkorrektur

3.2.1 Funktion

Schleifen bedeutet zum einen das Bearbeiten eines Werkstücks und zum anderen das Abrichten der Schleifscheibe. Dies kann sowohl in einem Kanal als auch in getrennten Kanälen erfolgen.

Damit eine Bearbeitung während dem Abrichten möglich ist, muss die durch das Abrichten der Schleifscheibe entstehende Verkleinerung der Schleifscheibe als Werkzeugkorrektur dem aktuellen Werkzeug im Bearbeitungskanal übergeben und sofort wirksam werden. Das kann mit der Funktion "Online-Werkzeugkorrektur" realisiert werden.

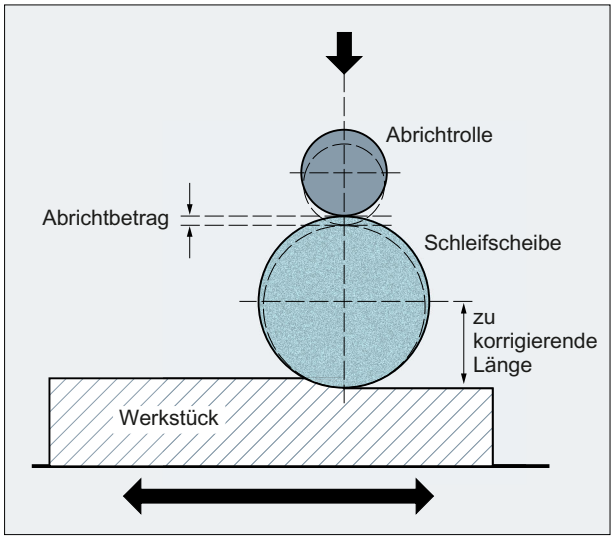


Bild 3-8 Abrichten während der Bearbeitung mit einer Abrichtrolle

Die Funktion bietet folgende Möglichkeiten zum Schreiben der Werkzeugkorrektur:

- Schreiben kontinuierlich satzweise (PUTFTOCF (Seite 262))
  Mit PUTFTOCF erfolgt der Abrichtvorgang zeitgleich mit der Bearbeitung.
  Die Werkzeugkorrektur wird im Bearbeitungskanal kontinuierlich nach einer Polynom-Funktion 1., 2. oder 3. Grades geändert, die vorher mit FCTDEF (Seite 261) definiert werden muss.
  PUTFTOCF wirkt immer satzweise, d. h. im anschließenden Verfahrsatz.
- Schreiben kontinuierlich modal: ID=1 DO FTOC (weitere Informationen siehe Funktionshandbuch Synchronaktionen)
- Schreiben diskret (PUTFTOC (Seite 263))
  Mit PUTFTOC erfolgt der Abrichtvorgang nicht zeitgleich mit der Bearbeitung aus einem parallelen Kanal. Der mit PUTFTOC angegebene Korrekturwert wird im Zielkanal sofort wirksam.

Aktivierung/Deaktivierung

Die Aktivierung/Deaktivierung der Online-WZK im Teileprogramm erfolgt mittels G-Befehl FTOCON/FTOCOF (Seite 264).

Weitere Eigenschaften

- Die Online-Werkzeugkorrektur kann für jedes Schleifwerkzeug (WZ-Typ: 400 bis 499) in jedem Kanal aktiviert werden.
- Im Allgemeinen wirkt die Online-Werkzeugkorrektur auf den Längenverschleiß. Die Zuordnung der Längen zu den Geometrieachsen erfolgt anhand der aktuellen Ebene entsprechend des Werkzeugtyps wie bei Geometrie und Verschleiß.
- Die Überwachung der Schleifspindel wirkt auch bei Online-Werkzeugkorrektur.

Hinweis

Die Online-Werkzeugkorrektur kann nur bei Schleifwerkzeugen angewendet werden.

Korrigiert wird immer der Verschleißparameter der gewählten Länge. Ist die Längenkorrektur für mehrere Schneiden identisch, so muss über Verkettungsvorschrift dafür gesorgt werden, dass die Werte automatisch auch für die 2. Schneide korrigiert werden.

Ist im Bearbeitungskanal Online-Werkzeugkorrektur aktiv, dürfen die Verschleißwerte für das aktive Werkzeug in diesem Kanal nicht aus dem Bearbeitungsprogramm oder über Bedienung geändert werden.

Eine Änderung des Verschleißes für den Radius (P15) wird erst bei Neuanwahl des Werkzeugs berücksichtigt.

Die Online-Werkzeugkorrektur wirkt auch für die konstante Scheibenumfangsgeschwindigkeit (SUG), d. h. die Spindeldrehzahl wird um den entsprechenden Wert korrigiert.

3.2.2 Inbetriebnahme

Anzahl definierbarer Werkzeughalter

Die Funktion "Online-Werkzeugkorrektur" kann nicht gleichzeitig mit der Funktion "Orientierbare Werkzeugträger“ verwendet werden.

Bei Verwendung der Funktion "Online-Werkzeugkorrektur" ist die maximale Anzahl definierbarer Werkzeughalter im TO-Bereich auf "0" zu setzen:

MD18075 $MN_MM_NUM_TOOLHOLDERS = 0

Hinweis

Bei SINUMERIK 828D ist der Wert von MD18075 nur in der Technologie "Schleifen" einstellbar.

Freigabe Online-Werkzeugradiuskorrektur

Für die Online-Werkzeugradiuskorrektur (Seite 268) ist zusätzlich die Freigabe über ein Maschinendatum erforderlich:

MD20254 $MC_ONLINE_CUTCOM_ENABLE = 1

3.2.3 Programmierung

3.2.3.1 Polynom-Funktion definieren (FCTDEF)

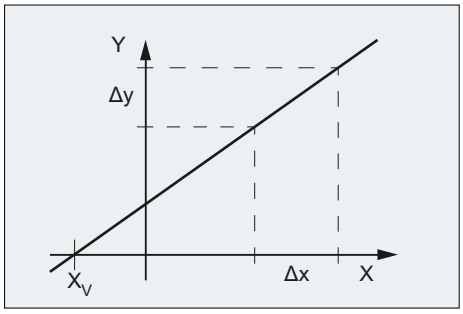
Bestimmte Abrichtstrategien (z. B. Abrichtrolle) zeichnen sich dadurch aus, dass die Schleifscheibe kontinuierlich (linear) mit der Zustellung der Abrichtrolle am Radius abnimmt. Hierfür benötigt man eine lineare Funktion zwischen der Zustellung der Abrichtrolle und dem Schreiben des Verschleißwerts der jeweiligen Länge. Die Definition der linearen Funktion erfolgt über die vordefinierte Prozedur FCTDEF(...) für Polynom-Funktionen bis maximal dritten Grades.

Geradengleichung

$y = f(x) = a_0 + a_1 * x_1$

a_1: Steigung der Geraden, mit $a_1 = \Delta x / \Delta y$

a_0: Verschiebung der Geraden entlang der X-Achse, mit $a_0 = -a1 * X_V$



Syntax

```
FCTDEF(<Func>,<LLimit>,<ULimit>,<a0>,<a1>,<a2>,<a3>)
```

Bedeutung

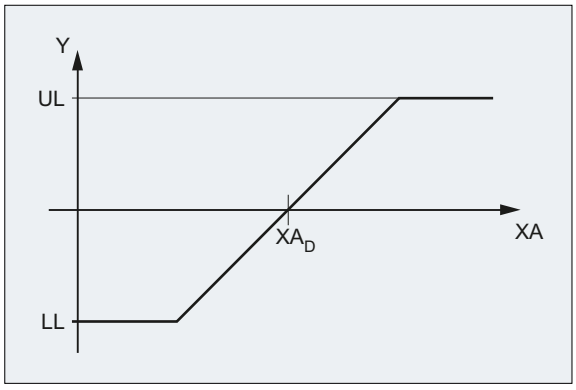
FCTDEF(...):	Definition einer Polynom-Funktion für PUTFTOCF(...): $y = f(x) = a_0 + a_1*x + a_2*x^2 + a_3*x^3$	
<Func>:	Funktionsnummer	
	Datentyp:	INT
	Wertebereich:	1, 2, 3
<LLimit>:	Unterer Begrenzungswert	
	Datentyp:	REAL
<ULimit>:	Oberer Begrenzungswert	
	Datentyp:	REAL
<a0>,<a1>,<a2>,<a3>:	Koeffizienten der Polynom-Funktion	
	Datentyp:	REAL

Beispiel

Festlegungen

- Funktionsnummer: 1
- Unterer und Oberer Begrenzungswert: -100, 100
- Steigung der Kennlinie: $a_1 = 1$
- Der Arbeitspunkt soll in der Mitte der Kennlinie liegen. Die Kennlinie muss dazu anhand der Sollposition der Achse XA im WKS zum Zeitpunkt der Funktionsdefinition im NC-Programm in negativer Y-Richtung verschoben werden: $a_0 = -a_1 * XA_D = -1 * \$AA_IW$
- $a_2 = a_3 = 0$

Kennlinie



UL	Oberer Begrenzungswert
LL	Unterer Begrenzungswert
XA_D	Sollwert der Achse XA zum Zeitpunkt der Funktionsdefinition im NC-Programm

Programmierung

Programmcode	Kommentar
`FCTDEF(1,-100,100,-$AA_IW[XA],1)`	`; Funktionsdefinition`

3.2.3.2 Online-Werkzeugkorrektur schreiben, kontinuierlich (PUTFTOCF)

Mit der vordefinierten Prozedur PUTFTOCF(...) wird eine Online-Werkzeugkorrektur über eine zuvor mit FCTDEF(...) (Seite 261) definierte Polynom-Funktion vorgenommen.

Hinweis

Die Online-Werkzeugkorrektur kann auch über eine Synchronaktion erfolgen.

Weitere Informationen siehe Funktionshandbuch Synchronaktionen.

Syntax

```
PUTFTOCF(<Func>,<RefVal>,<ToolPar>,<Chan>,<Sp>)
```

Bedeutung

PUTFTOCF(...):	Online-WZK schreiben, kontinuierlich satzweise anhand der mit FCTDEF(...) definierten Polynom-Funktion	
<Func>:	Funktionsnummer, festgelegt bei der Funktionsdefinition mit FCTDEF(...)	
	Datentyp:	INT
	Wertebereich:	1, 2, 3
<RefVal>:	Bezugswert, von dem die Korrektur abgeleitet werden soll (z. B. Sollwert einer Achse)	
	Datentyp:	VAR REAL
<ToolPar>:	Nummer des Verschleißparameters (Länge 1, 2 oder 3), in dem der Korrekturwert verrechnet werden soll	
	Datentyp:	INT
<Chan>:	Nummer des Kanals, in dem die Online-WZK wirksam werden soll **Hinweis**: Nur erforderlich, wenn die Korrektur nicht im aktiven Kanal wirksam werden soll.	
	Datentyp:	INT
<Sp>:	Nummer der Spindel, für die die Online-WZK wirksam werden soll **Hinweis:** Nur erforderlich, wenn statt dem aktiven, im Einsatz befindlichen Werkzeug eine nicht aktive Schleifscheibe korrigiert werden soll.	
	Datentyp:	INT

3.2.3.3 Online-Werkzeugkorrektur schreiben, diskret (PUTFTOC)

Funktion

Mit der vordefinierten Prozedur PUTFTOC(...) wird eine Online-Werkzeugkorrektur mit einem festen Korrekturwert vorgenommen.

Syntax

```
PUTFTOC(<CorrVal>,<ToolPar>,<Chan>,<Sp>)
```

Bedeutung

PUTFTOC(...):	Online-WZK schreiben	
<CorrVal>:	Korrekturwert, der zum Verschleißparameter addiert wird	
	Datentyp:	REAL
<ToolPar>:	Nummer des Verschleißparameters (Länge 1, 2 oder 3), in dem der Korrekturwert verrechnet werden soll	
	Datentyp:	INT
<Chan>:	Nummer des Kanals, in dem die Online-WZK wirksam werden soll **Hinweis:** Nur erforderlich, wenn die Korrektur nicht im aktiven Kanal wirksam werden soll.	
	Datentyp:	INT

<Sp>:	Nummer der Spindel, für die die Online-WZK wirksam werden soll **Hinweis:** Nur erforderlich, wenn statt dem aktiven, im Einsatz befindlichen Werkzeug eine nicht aktive Schleifscheibe korrigiert werden soll.	
	Datentyp:	INT

3.2.3.4 Online-Werkzeugkorrektur ein-/ausschalten (FTOCON/FTOCOF)

Mit den G-Befehlen FTOCON und FTOCOF wird die Online-Werkzeugkorrektur ein- bzw. ausgeschaltet.

Syntax

```
FTOCON
...
FTOCOF
```

Bedeutung

FTOCON:	Online-Werkzeugkorrektur einschalten Der Befehl muss in dem Kanal programmiert werden, in dem die Online-Werkzeugkorrektur eingeschaltet werden soll.
FTOCOF:	Online-Werkzeugkorrektur ausschalten Der Befehl muss in dem Kanal programmiert werden, in dem die Online-Werkzeugkorrektur ausgeschaltet werden soll. **Hinweis:** Mit FTOCOF wird die Werkzeugkorrektur nicht weiter herausgefahren. In den schneidenspezifischen Korrekturdaten bleibt aber der mit PUTFTOC/PUTFTOCF eingerechnete Wert erhalten. Zum endgültigen Deaktivieren der Online-WZK muss nach FTOCOF noch eine An-/Abwahl des Werkzeugs (T...) erfolgen.

3.2.4 Randbedingungen

DRF-Verschiebung

Die Online-Werkzeugkorrektur wird der programmierten Bewegung der Achse unter Berücksichtigung der vorgegebenen Grenzwerte überlagert (z. B. Geschwindigkeit).

Stehen für eine Achse gleichzeitig DRF-Verschiebung und Online-Korrektur an, so wird zuerst die DRF-Verschiebung berücksichtigt.

Beschleunigungsreserve

Die anstehende Online-Werkzeugkorrektur wird mit der JOG-Geschwindigkeit unter Berücksichtigung der maximalen Beschleunigung herausgefahren.

Bei FTOCON wird folgendes kanalspezifisches Maschinendatum berücksichtigt:

MD20610 $MC_ADD_MOVE_ACCEL_RESERVE

Damit kann eine Beschleunigungsreserve für die Bewegung reserviert werden, so dass die überlagerte Bewegung sofort ausgeführt werden kann.

Referenzpunktfahren

Bei Referenzpunktfahren mit G74 wird die anstehende Online-Korrektur gelöscht.

Werkzeugwechsel

Sofern seit dem letzten Werkzeug- bzw. Schneidenwechsel FTOCON aktiv war, wird bei Werkzeugwechsel steuerungsintern Vorlaufstopp mit Neusynchronisation ausgelöst.

Schneidenwechsel ist ohne Vorlaufstopp möglich.

Hinweis

Der Werkzeugwechsel in Verbindung mit der Online-Werkzeugkorrektur kann durch Anwahl über T-Nummern durchgeführt werden.

Werkzeugwechsel mit M6 ist in Verbindung mit der Online-Werkzeugkorrektur nicht möglich.

Bearbeitungsebene und Transformation

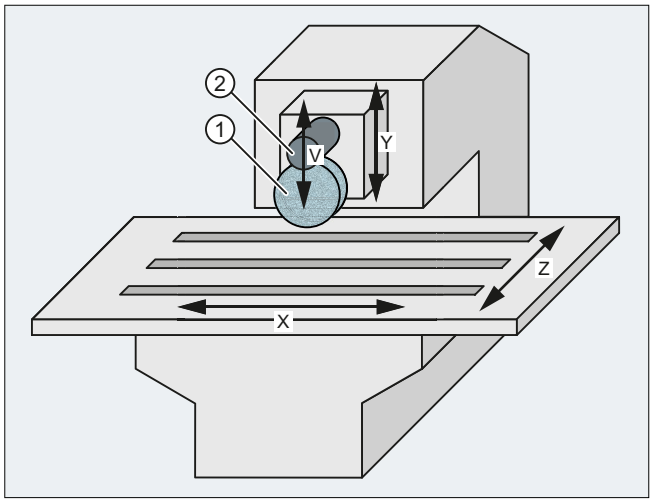
- FTOCON ist nur mit der Transformation "Schräge Achse" möglich.
- Ebenenwechsel (z. B. G17 nach G18) und Transformationswechsel ist bei FTOCON nicht möglich, außer im Zustand FTOCOF.

Reset und Betriebsartenwechsel

- NC-Stop und Programmende mit M2/M30 wird bei anstehender Online-Korrektur solange verzögert, bis die Korrektur herausgefahren ist.
- Bei NC-Reset wird die Online-Werkzeugkorrektur sofort abgewählt.
- Online-Werkzeugkorrekturen sind in der Betriebsart "Automatik" und bei aktivem Programm möglich.

3.2.5 Beispiele

3.2.5.1 Beispiel: Online-Werkzeugkorrektur schreiben, kontinuierlich



① Schleifscheibe
② Abrichtrolle
X: Pendelachse
Y: Zustellachse: Schleifscheibe
Z: Tischachse
V: Zustellachse: Abrichtrolle

Bild 3-9 Flachschleifmaschine

Vorgaben

- Werkzeugkorrektur
  - Bearbeitungsebene: Y/Z-Ebene (G19)
  - Werkzeugtyp: 401 (Länge 1 wirkt in Z, Länge 2 wirkt in Y)
- Kanal 1: Bearbeitungskanal, Achsen X, Y, Z
- Kanal 2: Abrichtkanal, Achse V

Nach dem Beginn der Schleifbebearbeitung bei Y100 soll die Schleifscheibe um den Betrag 0,05 (in V-Richtung) abgerichtet werden. Der Abrichtbetrag soll mit Online-Werkzeugkorrektur kontinuierlich kompensiert werden.

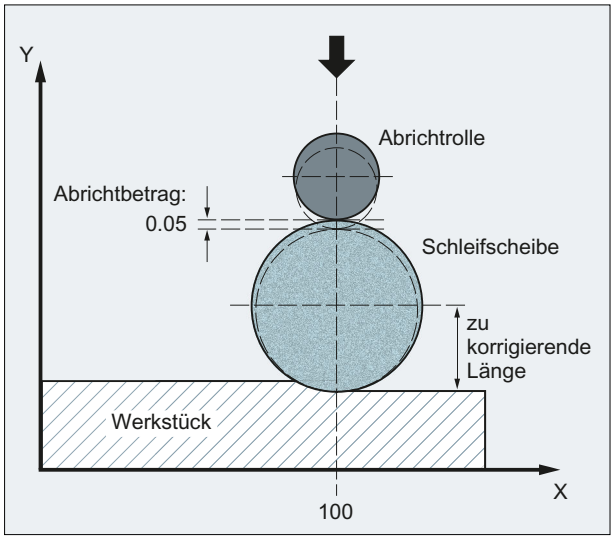


Bild 3-10 Werkzeugkorrektur

Programm (Ausschnitt) für Kanal 1: Bearbeitungskanal

Programmcode	Kommentar
...	
G1 G18 F10 G90	; Grundstellung
T1 D1	; Aktuelles Werkzeug anwählen
S100 M3 X100	; Spindel einschalten, X-Achse auf Ausgangsposition
INIT (2, "/_N_MPF_DIR/_N_ABRICHT_MPF", "S")	; Anwahl: Programm "ABRICHT" in Kanal 2
START (2)	; Start: Kanal 2
X200	; X-Achse auf Zielposition
FTOCON	; Online-Korrektur einschalten
...	

Programm für Kanal 2: Abrichtkanal

Die V-Achse verfährt parallel zur Y-Achse, d. h. die Länge 3 der Werkzeugkorrektur wirkt in Richtung der Y-Achse und muss somit kompensiert werden.

Programmcode	Kommentar
PROC ABRICHT	
FCTDEF(1, -1000, 1000, -$AA_IW[V], 1)	; Funktion definieren
PUTFTOCF(1, $AA_IW[V], 3, 1)	; Online-WZK schreiben kontinuierlich:
	; - Funktionsnummer: 1
	; - Bezugswert: Sollposition im WKS der V-Achse
	; - WZ-Parameter: Länge 3 der Schleifscheibe
	; - Kanal: 1
	; - Spindel: aktive Schleifscheibe

Programmcode	Kommentar
V-0.05 G1 F0.01 G91	; Zustellbewegung der V-Achse zum Abrichten
M30	

3.3 Online-Werkzeugradiuskorrektur

Funktion

Wenn die Werkzeuglängsachse und die Kontur senkrecht aufeinander stehen, dann kann die Korrekturgröße als Längenkorrektur auf eine der drei Geometrieachsen wirken (Online-Werkzeuglängenkorrektur).

Ist diese Bedingung nicht erfüllt, dann kann die Korrekturgröße als echte Radiuskorrektur eingegeben werden (Online-Werkzeugradiuskorrektur).

Freigabe

Für die Online-Werkzeugradiuskorrektur ist die Freigabe über ein Maschinendatum erforderlich:

MD20254 $MC_ONLINE_CUTCOM_ENABLE = 1

Aktivierung/Deaktivierung

Die Online-Werkzeugradiuskorrektur wird mit den G-Befehlen FTOCON und FTOCOF (Seite 264) ein-/ausgeschaltet (wie die Online-Werkzeuglängenkorrektur).

Parametrierung

Die Parametrierung der Online-Werkzeugkorrektur erfolgt mittels PUTFTOCF (Seite 262) bzw. PUTFTOC (Seite 263).

Der Parameter <ToolPar> (Verschleißparameter, in dem der Korrekturwert verrechnet werden soll), muss für die Online-Werkzeugradiuskorrektur wie folgt eingestellt werden:

<ToolPar> = 4

Randbedingungen

- Die Werkzeugradiuskorrektur und damit auch die Online-Werkzeugradiuskorrektur kann nur dann aktiviert werden, wenn das angewählte Werkzeug einen Radius ungleich "0" hat. Damit kann eine Bearbeitung ausschließlich mit der Online-Werkzeugradiuskorrektur nicht realisiert werden.
- Die Online-Korrekturwerte sollten klein gegenüber dem Originalradius sein, damit die Dynamikreserven bei den überlagerten Bewegungen nicht überschritten werden.
- Bei Schleif- und Drehwerkzeugen (Typ 400 - 599) wird der Korrekturwert entsprechend der Schneidenlage berücksichtigt, d. h. bei aktiver Werkzeugradiuskorrektur wirkt er als Radiuskorrektur, bei ausgeschalteter Werkzeugradiuskorrektur als Längenkorrektur in den durch die Schneidenlage spezifizierten Achsen.
  Bei allen anderen Werkzeugtypen wird der Korrekturwert nur dann berücksichtigt, wenn die Werkzeugradiuskorrektur mit G41 bzw. G42 aktiviert wurde. Beim Ausschalten der Werkzeugradiuskorrektur mit G40 wird der Korrekturwert herausgefahren.

3.4 Schleifspezifische Werkzeugüberwachung

3.4.1 Funktion

Die schleifspezifische Werkzeugüberwachung setzt sich aus Geometrie- und Drehzahlüberwachung zusammen. Sie kann für jedes Schleifwerkzeug (WZ-Typ: 400 bis 499) aktiviert werden.

Hinweis

Die Überwachung eines Werkzeugs wird durch die Anwahl der Überwachung eines anderen Werkzeugs nicht abgewählt, wenn sich die Werkzeuge auf unterschiedliche Spindeln beziehen.

Zu jedem Zeitpunkt kann für jede Spindel ein Werkzeug und somit **eine** Überwachung aktiv sein.

Aktive Überwachungen bleiben über Reset wirksam.

Geometrieüberwachung

Überwacht werden der aktuelle Scheibenradius und die aktuelle Scheibenbreite.

Als aktueller Scheibenradius wird derjenige Korrekturwert der ersten Schneide (D1) eines Schleifwerkzeugs verwendet, auf den im Parameter $TC_TPG9 verwiesen wird. Er wird bei

aktiver Überwachung mit dem im Parameter $TC_TPG3 stehenden Wert (minimaler Scheibenradius) verglichen.

Hinweis

Die Überwachung für den Scheibenradius wirkt auch bei aktiver Online-Werkzeugkorrektur:

- bei der Aktivierung der Überwachung
- beim Ändern des aktuellen Radius (Online-Werkzeugkorrektur, Verschleißparameter) bzw. der aktuellen Breite ($TC_TPG5)

Die aktuelle Scheibenbreite wird in der Regel durch den Abrichtzyklus ermittelt und kann im Parameter $TC_TPG5 eines Schleifwerkzeugs eingetragen werden. Dieser eingetragene Wert wird bei aktiver Überwachung mit dem im Parameter $TC_TPG4 stehenden Wert (minimale Scheibenbreite) verglichen.

Unterschreitet der aktuelle Scheibenradius den im Parameter $TC_TPG3 bzw. die aktuelle Scheibenbreite ($TC_TPG5) den im Parameter $TC_TPG4 festgelegten Wert, setzt die Steuerung das folgende NC/PLC-Nahtstellensignal auf "1":

DB31, ... DBX83.3 (Geometrieüberwachung) == 1

Hinweis

Eine steuerungsinterne Fehlerreaktion erfolgt nicht.

Drehzahlüberwachung

Überwacht werden die maximale Scheibenumfangsgeschwindigkeit in ms^{-1} (Parameter $TC_TPG7) bzw. die maximale Spindeldrehzahl in min^{-1} (Parameter $TC_TPG6).

Die Überwachung des Drehzahlsollwerts auf den Drehzahlgrenzwert erfolgt zyklisch unter Berücksichtigung des Spindel-Overrides.

Als Drehzahlgrenzwert gilt der kleinere Wert aus dem Vergleich zwischen der maximalen Spindeldrehzahl und der berechneten Drehzahl aus maximaler Scheibenumfangsgeschwindigkeit und aktuellem Scheibenradius.

Der Drehzahlgrenzwert wird neu bestimmt:

- beim Einschalten der Überwachung
- beim Ändern der Online-Korrekturwerte (Verschleißparameter)

Bei Erreichen des Drehzahlgrenzwerts wird auf den Drehzahlgrenzwert begrenzt und das folgende NC/PLC-Nahtstellensignal auf "1" gesetzt:

DB31, ... DBX83.6 (Drehzahlüberwachung) == 1

Hinweis

Eine steuerungsinterne Fehlerreaktion erfolgt nicht.

Aktivierung/Deaktivierung

Die Aktivierung/Deaktivierung der schleifspezifischen Werkzeugüberwachung erfolgt durch Programmierung von TMON(...)/TMOF(...) (Seite 271) im Teileprogramm.

3.4.2 Inbetriebnahme

Automatische Aktivierung

Soll bei Anwahl der Werkzeuglängenkorrektur eines Schleifwerkzeugs mit ungerader Werkzeugtypnummer die schleifspezifische Werkzeugüberwachung automatisch aktiviert werden, muss das folgende Maschinendatum auf "1" gesetzt werden:

MD20350 $MC_TOOL_GRIND_AUTO_TMON (Aktivierung der Werkzeugüberwachung) = 1

3.4.3 Programmierung

3.4.3.1 Schleifspezifische Werkzeugüberwachung ein-/ausschalten (TMON, TMOF)

Mit den vordefinierten Prozeduren TMON(...) und TMOF(...) wird die schleifspezifische Werkzeugüberwachung (Geometrie- und Drehzahlüberwachung) ein- bzw. ausgeschaltet.

Voraussetzung

Die werkzeugspezifischen Parameter $TC_TPG1 bis $TC_TPG9 müssen gesetzt sein.

Syntax

```
TMON(<TNr>)
...
TMOF(<TNr>)
```

Bedeutung

`TMON(...)`:	Schleifspezifische Werkzeugüberwachung einschalten Der Befehl muss in dem Kanal programmiert werden, in dem die schleifspezifische Werkzeugüberwachung eingeschaltet werden soll.
`TMOF(...)`:	Schleifspezifische Werkzeugüberwachung ausschalten Der Befehl muss in dem Kanal programmiert werden, in dem die schleifspezifische Werkzeugüberwachung ausgeschaltet werden soll.

<TNr>:	T-Nummer **Hinweis:** Nur erforderlich, wenn die Überwachung statt für das aktive, im Einsatz befindliche Werkzeug für eine nicht aktive Schleifscheibe ein- bzw. ausgeschaltet werden soll.
TMOF(0):	Überwachung für alle Werkzeuge ausschalten

3.5 Konstante Scheibenumfangsgeschwindigkeit (SUG)

3.5.1 Funktion

Für Schleifscheiben wird in der Regel statt der Spindeldrehzahl die Scheibenumfangsgeschwindigkeit verwendet. Der einzustellende Wert wird von technologischen Prozessparametern bestimmt (z. B. Schleifscheibenkenngröße, Materialpaarung).

Um im Teileprogramm die Scheibenumfangsgeschwindigkeit angeben zu können, müssen die werkzeugspezifischen Parameter $TC_TPG1 (Seite 250), $TC_TPG8 (Seite 252) und $TC_TPG9 (Seite 253) gesetzt sein und die Funktion "Konstante Scheibenumfangsgeschwindigkeit (SUG)" eingeschaltet werden.

Hinweis

Das Einschalten von SUG führt **nicht** zur automatischen Aktivierung der Werkzeuglängenkorrektur oder schleifspezifischen Werkzeugüberwachung!

Sobald die Funktion aktiv ist, wird jeder nachfolgende S-Wert für die Spindel, die über $TC_TPG1 der Schleifscheibe zugeordnet ist, als Scheibenumfangsgeschwindigkeit interpretiert. Die Drehzahl der Schleifscheibe wird dann so eingestellt, dass sich unter Berücksichtigung des aktuellen Scheibenradius eine gleichbleibende Scheibenumfangsgeschwindigkeit ergibt.

Die Formel für die Drehzahlberechnung lautet:

$$n[\mathrm{min}^{-1}] = \frac{\mathrm{SUG}\ [\mathrm{m*s}^{-1}] * 60}{2\pi * R[\mathrm{m}]}$$

Im werkzeugspezifischen Parameter $TC_TPG9 ist festgelegt, welcher Korrekturwert für die Berechnung verwendet werden soll. Zu diesem Korrekturwert wird der jeweilige Verschleiß und in Abhängigkeit vom Werkzeugtyp das Basismaß addiert. Die Summe wird bei positivem Wert des Parameters $TC_TPG8 (Winkel der schrägen Scheibe) durch "cos" ($TC_TPG8) und bei negativem Wert durch "sin" ($TC_TPG8) dividiert.

Die Drehzahl wird beim Auftreten folgender Ereignisse neu berechnet:

- Vorgabe einer neuen Scheibenumfangsgeschwindigkeit, z. B. durch:
  - Programmierung im Teileprogramm/Überspeichern
  - Zuweisung an die Adresse "S" in MDA
  - Spindelsteuerung über PLC (FC18)
- Ändern der Online-Korrekturwerte (Verschleißparameter)

Aktivierung/Deaktivierung

Das Ein-/Ausschalten von SUG erfolgt durch Programmierung von GWPSON(...)/ GWPSOF(...) im Teileprogramm (siehe "Programmierung (Seite 274)").

Status

Über das folgende Nahtstellensignal kann festgestellt werden, ob die SUG aktiv ist oder nicht:

DB31, ... DBX84.1 (SUG aktiv)

Neues Werkzeug

Soll für eine Spindel, für die SUG bereits aktiv ist, SUG mit einem neuen Werkzeug angewählt werden, muss zuerst die aktive SUG abgewählt werden (sonst erfolgt ein Alarm).

Verhalten bei Reset/Teileprogrammende

In der Gundeinstellung wird bei Reset oder Teileprogrammende SUG deaktiviert und die Spindel gestoppt. Die diesem Verhalten zugrunde liegenden Maschinendateneinstellungen können bei Bedarf verändert werden (siehe "Inbetriebnahme (Seite 273)").

Weitere Eigenschaften

- SUG kann für jedes Schleifwerkzeug (WZ-Typ: 400 bis 499) in jedem Kanal aktiviert werden.
- SUG kann für mehrere Spindeln eines Kanals mit jeweils unterschiedlichen Schleifwerkzeugen gleichzeitig aktiv sein.

3.5.2 Inbetriebnahme

Automatische Aktivierung

Soll die konstante Scheibenumfangsgeschwindigkeit (SUG) für eine Spindel nach Power On automatisch angewählt und über Betriebsartenwechsel, Reset oder Teileprogrammende angewählt bleiben, muss das folgende Maschinendatenbit auf "1" gesetzt werden:

MD35032 $MA_SPIND_FUNC_RESET_MODE Bit 0 = 1

Hinweis

Voraussetzung ist, dass es sich bei dem Wekzeug um ein Schleifwerkzeug (WZ-Typ 400 bis 499) handelt, die werkzeugspezifischen Parameter $TC_TPG1, $TC_TPG8 und $TC_TPG9 gesetzt sind und $TC_TPG1 auf eine gültige Spindel verweist.

Verhalten der Spindel nach Reset/Teileprogrammende

Soll die Spindel nach Reset oder Teileprogrammende nicht stoppen, sondern mit der aktuellen Drehzahl weiterdrehen, muss das folgende Maschinendatum auf "1" gesetzt werden:

MD35040 $MA_SPIND_ACTIVE_AFTER_RESET = 1

Hinweis

MD35040 ist nur wirksam in der Spindelbetriebsart Steuerbetrieb.

3.5.3 Programmierung

3.5.3.1 Konstante Scheibenumfangsgeschwindigkeit ein-/ausschalten (GWPSON, GWPSOF)

Mit den vordefinierten Prozeduren GWPSON(...) und GWPSOF(...) wird die konstante Scheibenumfangsgeschwindigkeit (SUG) für Schleifwerkzeuge (WZ-Typ: 400 bis 499) ein- bzw. ausgeschaltet.

Syntax

```
GWPSON(<TNr>)
S<n>=...
...
GWPSOF(<TNr>)
```

Bedeutung

GWPSON(...):	Konstante Scheibenumfangsgeschwindigkeit einschalten
GWPSOF(...):	Konstante Scheibenumfangsgeschwindigkeit ausschalten
<TNr>:	T-Nummer **Hinweis:** Nur erforderlich, wenn die konstante Scheibenumfangsgeschwindigkeit statt für das aktive, im Einsatz befindliche Werkzeug für eine nicht aktive Schleifscheibe ein- bzw. ausgeschaltet werden soll.

S<n>=…:	Scheibenumfangsgeschwindigkeit in m/s oder ft/s für Spindel <n>
S0=... bzw. S... :	Scheibenumfangsgeschwindigkeit für die Masterspindel

Status abfragen

Mit der folgenden Systemvariablen kann vom Teileprogramm aus abgefragt werden, ob die konstante Scheibenumfangsgeschwindigkeit für eine bestimmte Spindel aktiv ist:

$P_GWPS[<n>] ; mit <n> = Spindelnummer

Wert	Bedeutung
0 (= FALSE)	SUG ist **aus**geschaltet.
1 (= TRUE)	SUG ist **ein**geschaltet.

3.5.4 Beispiel

Für die Schleifwerkzeuge T1 und T5 soll konstante Scheibenumfangsgeschwindigkeit gelten. T1 ist das aktive Werkzeug.

Daten von Werkzeug T1 (Umfangsschleifscheibe)

```
$TC_DP1[1,1] = 403          ;Werkzeugtyp
$TC_DP3[1,1] = 300          ;Länge1
$TC_DP4[1,1] = 50           ;Länge2
$TC_DP12[1,1] = 0           ;Verschleiß Länge 1
$TC_DP13[1,1] = 0           ;Verschleiß Länge 2
$TC_DP21[1,1] = 300         ;Basis Länge 1
$TC_DP22[1,1] = 400         ;Basis Länge 2
$TC_TPG1[1] = 1             ;Spindelnummer
$TC_TPG8[1] = 0             ;Winkel der schrägen Scheibe
$TC_TPG9[1] = 3             ;Parameter-Nr. für Radiusberechnung
```

Daten von Werkzeug T5 (schräge Scheibe)

```
$TC_DP1[5,1] = 401          ;Werkzeugtyp
$TC_DP3[5,1] = 120          ;Länge1
$TC_DP4[5,1] = 30           ;Länge2
$TC_DP12[5,1] = 0           ;Verschleiß Länge 1
$TC_DP13[5,1] = 0           ;Verschleiß Länge 2
$TC_DP21[5,1] = 100         ;Basis Länge 1
$TC_DP22[5,1] = 150         ;Basis Länge 2
$TC_TPG1[5] = 2             ;Spindelnummer
$TC_TPG8[5] = 45            ;Winkel der schrägen Scheibe
$TC_TPG9[5] = 3             ;Parameter-Nr. für Radiusberechnung
```

Programmierung

Programmcode	Kommentar
N20 T1 D1	; T1 und D1 anwählen
N25 S1=1000 M1=3	; 1000 Umdr./min für Spindel 1
N30 S2=1500 M2=3	; 1500 Umdr./min für Spindel 2
...	
N40 GWPSON	; SUG für aktives Werkzeug T1 einschalten
N45 S[$P_ATPG[1]]=60	; SUG für aktives Werkzeug auf 60 m/s setzen → n=1909,85 min-1
...	
N50 GWPSON(5)	; SUG für Werkzeug 5 (Spindel 2) einschalten
N55 S[$TC_TPG1[5]]=40	; SUG für Spindel 2 auf 40 m/s setzen → n=1909,85 min-1
...	
N60 GWPSOF	; SUG für aktives Werkzeug ausschalten
N65 GWPSOF(5)	; SUG für Werkzeug 5 (Spindel 2) ausschalten
...	

3.6 Datenlisten

3.6.1 Maschinendaten

3.6.1.1 Allgemeine Maschinendaten

Nummer	Bezeichner: $MN_	Beschreibung
18075	MM_NUM_TOOLHOLDERS	Maximale Anzahl definierbarer Werkzeughalter
18094	MM_NUM_CC_TDA_PARAM	Anzahl der TDA
18096	MM_NUM_CC_TOA_PARAM	Anzahl der TOA
18100	MM_NUM_CUTTING_EDGES_IN_TOA	Werkzeugkorrekturen pro TOA

3.6.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: $MC_	Beschreibung
20254	ONLINE_CUTCOM_ENABLE	Freigabe Online-Werkzeugradiuskorrektur
20350	TOOL_GRIND_AUTO_TMON	WZ-Überwachung automatisch
20610	ADD_MOVE_ACCEL_RESERVE	Beschleunigungsreserve für überlagerte Bewegungen

3.6.1.3 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: $MA_	Beschreibung
32020	JOG_VELO	Konventionelle Achsgeschwindigkeit
35032	SPIND_FUNC_RESET_MODE	Reset-Verhalten einzelner Spindelfunktionen
35040	SPIND_ACTIVE_AFTER_RESET	Eigener Spindel-Reset

Anhang

A

A.1 Liste der Abkürzungen

A	
A	Ausgang
ADI4	Analog Drive Interface for 4 Axes
AC	Adaptive Control
ALM	Active Line Module
ARM	Asynchroner rotatorischer Motor
AS	Automatisierungssystem
ASCII	American Standard Code for Information Interchange: Amerikanische Code-Norm für den Informationsaustausch
ASIC	Application Specific Integrated Circuit: Anwender-Schaltkreis
ASUP	Asynchrones Unterprogramm
AUXFU	Auxiliary Function: Hilfsfunktion
AWL	Anweisungsliste
AWP	Anwenderprogramm

B	
BA	Betriebsart
BAG	Betriebsartengruppe
BCD	Binary Coded Decimals: Im Binärcode verschlüsselte Dezimalzahlen
BERO	Berührungsloser Näherungsschalter
BI	Binector Input
BICO	Binector Connector
BIN	Binary Files: Binärdateien
BIOS	Basic Input Output System
BKS	Basiskoordinatensystem
BO	Binector Output
BTSS	Bedientafelschnittstelle

C	
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CC	Compile Cycle: Compile-Zyklen
CEC	Cross Error Compensation
CI	Connector Input
CF-Card	Compact Flash-Card

C	
CNC	Computerized Numerical Control: Computerunterstützte numerische Steuerung
CO	Connector Output
CoL	Certificate of License
COM	Communication
CPA	Compiler Projecting Data: Projektierdaten des Compilers
CRT	Cathode Ray Tube: Bildröhre
CSB	Central Service Board: PLC-Baugruppe
CU	Control Unit
CP	Communication Processor
CPU	Central Processing Unit: Zentrale Rechnereinheit
CR	Carriage Return
CTS	Clear To Send: Meldung der Sendebereitschaft bei seriellen Daten-Schnittstellen
CUTCOM	Cutter Radius Compensation: Werkzeugradiuskorrektur

D	
DAU	Digital-Analog-Umwandler
DB	Datenbaustein (PLC)
DBB	Datenbaustein-Byte (PLC)
DBD	Datenbaustein-Doppelwort (PLC)
DBW	Datenbaustein-Wort (PLC)
DBX	Datenbaustein-Bit (PLC)
DDE	Dynamic Data Exchange
DDS	Drive Data Set: Antriebsdatensatz
DIN	Deutsche Industrie Norm
DIO	Data Input/Output: Datenübertragungs-Anzeige
DIR	Directory: Verzeichnis
DLL	Dynamic Link Library
DO	Drive Object
DPM	Dual Port Memory
DPR	Dual Port RAM
DRAM	Dynamischer Speicher (ungepuffert)
DRF	Differential Resolver Function: Differential-Drehmelder-Funktion (Handrad)
DRIVE-CLiQ	Drive Component Link with IQ
DRY	Dry Run: Probelaufvorschub
DSB	Decoding Single Block: Dekodierungseinzelsatz
DSC	Dynamic Servo Control / Dynamic Stiffness Control
DW	Datenwort
DWORD	Doppelwort (aktuell 32 Bit)

E	
E	Eingang
EES	Execution from External Storage
E/A	Ein-/Ausgabe
ENC	Encoder: Istwertgeber
EFP	Einfach Peripheriemodul (PLC–E/A–Baugruppe)
EGB	Elektronisch gefährdete Baugruppen/Bauelemente
EMV	Elektromagnetische Verträglichkeit
EN	Europäische Norm
ENC	Encoder: Istwertgeber
EnDat	Geberschnittstelle
EPROM	Erasable Programmable Read Only Memory: Löschbarer, elektrisch programmierbarer nur Lesespeicher
ePS Network Services	Dienste zur internetgestützten Maschinen-Fernwartung
EQN	Typbezeichnung eines Absolutwertgebers mit 2048 Sinussignalen/Umdrehung
ES	Engineering System
ESR	Erweitertes Stillsetzen und Rückziehen
ETC	ETC–Taste ">"; Erweiterung der Softkeyleiste im gleichen Menü

F	
FB	Funktionsbaustein (PLC)
FC	Function Call: Funktionsbaustein (PLC)
FEPROM	Flash–EPROM: Les– und schreibbarer Speicher
FIFO	First In First Out: Speicher, der ohne Adressangabe arbeitet und dessen Daten in derselben Reihenfolge gelesen werden, in der sie gespeichert wurden
FIPO	Feininterpolator
FPU	Floating Point Unit: Gleitpunkteinheit
FRK	Fräsradiuskorrektur
FST	Feed Stop: Vorschub Halt
FUP	Funktionsplan (Programmiermethode für PLC)
FW	Firmware

G	
GC	Global Control (PROFIBUS: Broadcast-Telegramm)
GDIR	Globaler Teileprogrammspeicher
GEO	Geometrie, z.B. Geometrieachse
GIA	Gear Interpolation Data: Getriebeinterpolationsdaten
GND	Signal Ground
GP	Grundprogramm (PLC)
GS	Getriebestufe
GSD	Gerätestammdatei zur Beschreibung eines PROFIBUS Slaves

G	
GSDML	Generic Station Description Markup Language: XML-basierte Beschreibungs-sprache zur Erstellung einer GSD-Datei
GUD	Global User Data: Globale Anwenderdaten

H	
HEX	Kurzbezeichnung für hexadezimale Zahl
HiFu	Hilfsfunktion
HLA	Hydraulischer Linearantrieb
HMI	Human Machine Interface: SINUMERIK-Bedienoberfläche
HSA	Hauptspindelantrieb
HW	Hardware

I	
IBN	Inbetriebnahme
IKA	Interpolatorische Kompensation
IM	Interface-Modul: Anschaltungsbaugruppe
IMR	Interface-Modul Receive: Anschaltungsbaugruppe für Empfangsbetrieb
IMS	Interface-Modul Send: Anschaltungsbaugruppe für Sendebetrieb
INC	Increment: Schrittmaß
INI	Initializing Data: Initialisierungsdaten
IPO	Interpolator
ISA	International Standard Architecture
ISO	International Standard Organization

J	
JOG	Jogging: Einrichtbetrieb

K	
K_V	Verstärkungsfaktor des Regelkreises
K_P	Proportionalverstärkung
$K_Ü$	Übersetzungsverhältnis
KOP	Kontaktplan (Programmiermethode für PLC)

L	
LAI	Logic Machine Axis Image: Logisches Maschinenachsen-Abbild
LAN	Local Area Network
LCD	Liquid-Crystal Display: Flüssigkristallanzeige
LED	Light Emitting Diode: Leuchtdiode
LF	Line Feed

L	
LMS	Lagemesssystem
LR	Lageregler
LSB	Least Significant Bit: Niederwertigstes Bit
LUD	Local User Data: Anwenderdaten (lokal)

M	
MAC	Media Access Control
MAIN	Main program: Hauptprogramm (OB1, PLC)
MB	Megabyte
MCI	Motion Control Interface
MCIS	Motion–Control–Information–System
MCP	Machine Control Panel: Maschinensteuertafel
MD	Maschinendatum bzw. Maschinendaten
MDA	Manual Data Automatic: Handeingabe
MDS	Motor Data Set: Motordatensatz
MELDW	Meldungswort
MKS	Maschinenkoordinatensystem
MM	Motor Module
MPF	Main Program File: Hauptprogramm (NC)
MSTT	Maschinensteuertafel

N	
NC	Numerical Control: Numerische Steuerung mit Satzaufbereitung, Verfahrbereich usw.
NCU	Numerical Control Unit: Hardware-Einheit des NC
NRK	Bezeichnung des Betriebssystems des NC
NST	Nahtstellensignal
NURBS	Non-Uniform Rational B-Spline
NV	Nullpunktverschiebung
NX	Numerical Extension: Achserweiterungsbaugruppe

O	
OB	Organisationsbaustein in der PLC
OEM	Original Equipment Manufacturer
OP	Operation Panel: Bedieneinrichtung
OPI	Operation Panel Interface: Bedientafel-Anschaltung
OPT	Options: Optionen
OLP	Optical Link Plug: Busstecker für Lichtleiter
OSI	Open Systems Interconnection: Normung für Rechnerkommunikation

P	
PAA	Prozessabbild der Ausgänge
PAE	Prozessabbild der Eingänge
PC	Personal Computer
PCIN	Name der SW für den Datenaustausch mit der Steuerung
PCMCIA	Personal Computer Memory Card International Association: Speichersteckkarten-Normierung
PCU	PC Unit: PC-Box (Rechnereinheit)
PG	Programmiergerät
PKE	Parameterkennung: Teil eines PKW
PKW	Parameterkennung: Wert (Parametrierteil eines PPO)
PLC	Programmable Logic Control: Anpass-Steuerung
PN	PROFINET
PNO	PROFIBUS-Nutzerorganisation
PO	POWER ON
POE	Programmorganisationseinheit
POS	Position/Positionieren
POSMO A	Positioning Motor Actuator: Positioniermotor
POSMO CA	Positioning Motor Compact AC: Komplette Antriebseinheit mit integrierter Leistungs– und Reglungsbaugruppe sowie Positioniereinheit und Programmspeicher; Wechselstrom–Einspeisung
POSMO CD	Positioning Motor Compact DC: wie CA, jedoch Gleichstromeinspeisung
POSMO SI	Positioning Motor Servo Integrated: Positioniermotor; Gleichstromeinspeisung
PPO	Parameter Prozessdaten Objekt ; Zyklisches Datentelegramm bei der Übertragung mit PROFIBUS–DP und Profil "Drehzahlveränderbare Antriebe"
PPU	Panel Processing Unit (zentrale Hardware einer Panel-basierten CNC-Steuerung z.B. SINUMERIK 828D)
PROFIBUS	Process Field Bus: Serieller Datenbus
PRT	Programmtest
PSW	Programmsteuerwort
PTP	Point to Point: Punkt zu Punkt
PUD	Program Global User Data: Programmglobale Anwendervariable
PZD	Prozessdaten: Prozessdatenteil eines PPO

Q	
QFK	Quadrantenfehler Kompensation

R	
RAM	Random Access Memory: Schreib-/Lese-Speicher
REF	Funktion Referenzpunkt anfahren
REPOS	Funktion Repositionieren
RISC	Reduced Instruction Set Computer: Prozessortyp mit kleinem Befehlssatz und schnellem Befehlsdurchsatz

R	
ROV	Rapid Override: Eingangskorrektur
RP	R-Parameter, Rechenparameter, vordefinierte Anwendervariable
RPA	R-Parameter Active: Speicherbereich in NC für R-Parameternummern
RPY	Roll Pitch Yaw: Drehungsart eines Koordinatensystems
RTLI	Rapid Traverse Linear Interpolation: Lineare Interpolation bei Eilgangbewegung
RTS	Request To Send: Sendeteil einschalten, Steuersignal von seriellen Daten-Schnittstellen
RTCP	Real Time Control Protocol

S	
SA	Synchronaktion
SBC	Safe Break Control: Sichere Bremsenansteuerung
SBL	Single Block: Einzelsatz
SBR	Subroutine: Unterprogramm (PLC)
SD	Settingdatum bzw. Settingdaten
SDB	System Datenbaustein
SEA	Setting Data Active: Kennzeichnung (Dateityp) für Settingdaten
SERUPRO	Search-Run by Program Test: Satzsuchlauf via Programmtest
SFB	System Funktionsbaustein
SFC	System Function Call
SGE	Sicherheitsgerichteter Eingang
SGA	Sicherheitsgerichteter Ausgang
SH	Sicherer Halt
SIM	Single in Line Module
SK	Softkey
SKP	Skip: Funktion zum Ausblenden eines Teileprogrammsatzes
SLM	Synchroner Linearmotor
SM	Schrittmotor
SMC	Sensor Module Cabinet Mounted
SME	Sensor Module Externally Mounted
SMI	Sensor Module Integrated
SPF	Sub Program File: Unterprogramm (NC)
SPS	Speicherprogrammierbare Steuerung = PLC
SRAM	Statischer Speicher (gepuffert)
SRK	Schneidenradiuskorrektur
SRM	Synchron rotatorischer Motor
SSFK	Spindelsteigungsfehlerkompensation
SSI	Serial Synchron Interface: Serielle synchrone Schnittstelle
SSL	Satzsuchlauf
STW	Steuerwort
SUG	Scheibenumfangsgeschwindigkeit
SW	Software

S	
SYF	System Files: Systemdateien
SYNACT	Synchronized Action: Synchronaktion

T	
TB	Terminal Board (SINAMICS)
TCP	Tool Center Point: Werkzeugspitze
TCP/IP	Transport Control Protocol / Internet Protocol
TCU	Thin Client Unit
TEA	Testing Data Active: Kennung für Maschinendaten
TIA	Totally Integrated Automation
TM	Terminal Module (SINAMICS)
TO	Tool Offset: Werkzeugkorrektur
TOA	Tool Offset Active: Kennzeichnung (Dateityp) für Werkzeugkorrekturen
TRANSMIT	Transform Milling Into Turning: Koordinatentransformation für Fräsbearbeitungen an einer Drehmaschine
TTL	Transistor–Transistor–Logik (Schnittstellen–Typ)
TZ	Technologiezyklus

U	
UFR	User Frame: Nullpunktverschiebung
UP	Unterprogramm
USB	Universal Serial Bus
USV	Unterbrechungsfreie Stromversorgung

V	
VDI	Interne Kommunikationsschnittstelle zwischen NC und PLC
VDI	Verein Deutscher Ingenieure
VDE	Verband Deutscher Elektrotechniker
VI	Voltage Input
VO	Voltage Output
VSA	Vorschubantrieb

W	
WAB	Funktion Weiches An- und Abfahren
WKS	Werkstückkoordinatensystem
WKZ	Werkzeug
WLK	Werkzeuglängenkorrektur
WOP	Werkstatt-orientierte Programmierung
WPD	Work Piece Directory: Werkstückverzeichnis
WRK	Werkzeug-Radius-Korrektur

W	
WZ	Werkzeug
WZK	Werkzeugkorrektur
WZV	Werkzeugverwaltung
WZW	Werkzeugwechsel

X	
XML	Extensible Markup Language

Z	
ZOA	Zero Offset Active: Kennung für Nullpunktverschiebungen
ZSW	Zustandswort (des Antriebs)

Index

$

$AC_CUT_INV, 169
$AC_CUTMOD, 168
$AC_CUTMOD_ANG, 148, 168
$AC_CUTMODK, 168
$NT_ROT_AX_NAME, 164
$P_AD, 168
$P_CUT_INV, 169
$P_CUTMOD, 168
$P_CUTMOD_ANG, 148, 168
$P_CUTMOD_ERR, 170
$P_CUTMODK, 168
$P_GWPS, 275
$P_ORI_DIFF, 159
$P_ORI_POS, 159
$P_ORI_SOL, 160
$P_ORI_STAT, 163
$P_TOOLBIN, 150
$P_TOOLENV, 218
$P_TOOLENVN, 218
$TC_CARR_CORR_ELEM, 120
$TC_DP1, 144, 248
$TC_DP10, 145
$TC_DP11, 145
$TC_DP2, 144, 150
$TC_DP24, 145
$TC_DPC1...10, 248
$TC_DPCE[t,d], 29
$TC_DPROT, 150
$TC_TPC1...10, 254
$TC_TPG_DRSPATH, 254
$TC_TPG_DRSPROG, 254
$TC_TPG3, 252
$TC_TPG4, 252
$TC_TPG5, 252
$TC_TPG8, 252
$TC_TPG9, 253

A

Ablage von Winkeln in den Werkzeugträgerdaten, 97
Abrichten, 258
Adaptermaß, 47
Alarm bei Vorlaufstopp, 82

B

Basismaß, 47
Bearbeitung in Richtung der Werkzeugorientierung, 174
Beschreibung einer Drehung, 106
Bewertung einzelner Verschleißkomponenten, 211

C

CDOF, 78
CDOF2, 78
CDON, 78
CORRTC, 117
CUT2D, 87
CUT2DD, 87
CUT2DF, 87
CUT2DFD, 87
CUTMOD, 165
CUTMODK, 165

D

DB31, ...
    DBX83.3, 270
    DBX83.6, 270
    DBX84.1, 273
DELTOOLENV, 216
D-Funktion, 23
DISC, 72
D-Nummern
    Vergabe von freien ..., 28

E

Eingeschränkte Orientierung des Werkzeuträgers, 97

F

FCTDEF, 261
Freischneidwinkel, 50
Freiwinkel, 145
FTOCOF, 264
FTOCON, 264

G

G40, 52
G41, 52
G42, 52
G450/G451, 71
G451, 73
G460, 84
G461, 84
G462, 85
G91 Erweiterung, 173
    Nullpunktverschiebung, 174
GETTCOR, 218
GETTENV, 217
GWPSOF, 274
GWPSON, 274

H

Halterwinkel, 145

I

Inkrementell programmierte Korrekturwerte, 173

K

Kinematik der Maschine, 97
Kinematiktyp, 98
Kollisionsüberwachung, 77, 78
KONT, 53
KONTC, 53
KONTT, 53
Korrektur
    -ebene, 90
    Werkzeuglängen-, 19
    Werkzeugradius-, 21
Korrekturnummer, 29
krümmungsstetig, 54

L

Längsbearbeitung, 51
LENTOAX, 237

M

Maschinen mit aufgelöster Kinematik, 104
Maschinen mit drehbarem Werkstück, 103
Maschinen mit drehbarem Werkzeug, 102
MD11346, 37
MD11410, 78
MD18075, 260
MD18080, 195
MD18088, 99, 125, 130
MD18096, 248, 254
MD18100, 24, 249
MD18104, 215
MD18105, 28, 29, 30, 32
MD18106, 23, 28, 29
MD18108, 199
MD18112, 200
MD18114, 181
MD18116, 216
MD20096, 26, 27, 28
MD20110, 132, 174, 204
MD20112, 132
MD20120, 132
MD20121, 23
MD20125, 147, 151
MD20126, 132, 152
MD20127, 152
MD20180, 122
MD20184, 124
MD20188, 106
MD20190, 106
MD20202, 68
MD20204, 61
MD20210, 74
MD20220, 72
MD20230, 74
MD20240, 76, 78
MD20250, 53
MD20252, 93
MD20254, 260, 268
MD20256, 83
MD20270, 23, 204, 205
MD20272, 199, 204, 205
MD20350, 271
MD20360, 36, 37, 147, 153, 223
MD20390, 190, 192, 193
MD20392, 191
MD20610, 265
MD22530, 132
MD22550, 21, 34, 35
MD22560, 22
MD22562, 35, 36
MD24100, 192
MD24110, 192
MD24120, 192

MD24558, 225
MD24570, 192
MD24572, 193
MD24658, 225
MD25574, 193
MD28082, 124
MD28085, 24, 126, 130
MD32750, 190, 192, 193
MD35032, 274
MD35040, 274

N

NORM, 53
Nullvektoren, 101
Nutsäge, 41

O

Online-Werkzeugkorrektur, 258
Online-Werkzeugradiuskorrektur, 268
Orientierbare Werkzeugträger, 94
  Beispiele, 134
  Berechnung der aktiven Werkzeuglänge, 122
  Kinematische Kette, 101
  Neuanlegen, 125
ORISOLH, 154

P

Parametersatz
  orientierbare Werkzeugträger, 97
Parametrierung der Drehachsen, 98
Planbearbeitung, 51
Polynome, Schnittpunktverfahren, 83
PUTFTOC, 263
PUTFTOCF, 262

S

Scheibenumfangsgeschwindigkeit, 272
Schneiden
  -anzahl von Konturwerkzeugen, 89
  -bezugspunkt, 148
  -lage, 144, 150
  -mittelpunkt, 148
Schneidenlage, 42
  relevante, 87
Schneidennummer, 29
Schnittrichtung, 144
SD42440, 174
SD42442, 173
SD42480, 82
SD42496, 82
SD42900, 186, 206
SD42910, 187, 206, 211, 212
SD42920, 187, 206, 211, 212
SD42930, 188, 206, 211
SD42935, 139, 207, 208, 212, 240
SD42940, 176, 206, 212
SD42950, 144, 177, 206, 212
SD42954, 178
SD42956, 178
SD42960, 191, 192
SD42974, 105
SD42984, 167
SD42998, 146, 154
SD42999, 154
SETTCOR, 225
SUG, 272, 274

T

tangentenstetig, 54
TCARR, 123
TCOABS, 123
TCOFR, 123
T-Funktion, 21
TMOF, 271
TMON, 271
TOA
  -Daten, 39
  -Einheit, 24
TOOLENV, 213
TOWBCS, 210
TOWKCS, 211
TOWMCS, 210
TOWSTD, 209
TOWTCS, 211
TOWWCS, 210

V

Vorausschauende Konturberechnung, 76

W

Wechsel des Frames, 96
Weiches An- und Abfahren
  Bedeutung, 58
  Teilbewegungen, 58

Werkstück-Kette, 120
Werkzeug
  Basismaß/Adaptermaß, 47
  DISC, 72
  -form, 45
  -form, aktive, 46
  -größe, aktive, 46
  -gruppe, 39
  -korrekturdaten, 39
  -länge, 44
  -längenkorrektur, 19
  -parameter, 37
  -radiuskorrektur, 21
  Schneide, 37
  -typ, 39
  -verschleiß, 46
  Werkzeuganwahl, 21
  Werkzeugradiuskorrektur-2D (WRK), 52
  Werkzeugschneide, 23
  Werkzeugwechsel, 21
Werkzeug-Kette, 120
Werkzeugkorrektur
  Typen, 96
  Verrechnung in der NC, 26
Werkzeuglängenkorrektur
  Geometrie-, 44
  Verschleiß-, 46
  werkstückspezifisch einrechnen, 205
  werkstückspezifisch einrechnen, Beispiel, 137
Werkzeugradiuskorrektur
  CUT2DF, 90
  konstant halten, 90
  Verschleiß-, 46
Werkzeugradiuskorrektur-2D, 51
  Abwahl, 71
  An - und Abfahrverhalten, 53
  Anwahl, 52
  geändertes Alarmverhalten, 82
  Geometrie-, 45
  Korrektur an Außenecken, 71
  Schnittpunkt G451, 73
  Übergangskreis, 72
  veränderlicher Korrekturwert, 81
  Weiches An- und Abfahren, 58
Werkzeugträger, orientierbare, 94
  drehbarer Werktisch, 122
  Neuanlegen, 125
  Programmierung, 130
  Randbedingungen, 131
  Schrägbearbeitung, 121
  Steuerungsverhalten bei Reset, Programmstart, Repos, 131
Werkzeugträgeranwahl, 94
Werkzeugtyp, 248
Werkzeugüberwachung
  schleifspezifische, 269
Werkzeugwechsel
  Korrekturspeicher, 24
Winkel
  Frei-, 145
  Halter-, 145

Z

Zuordnung Werkzeug / Werkzeugträger, 94