

K#\$ Persönliches Vorwort

Ab 2009 wird sich der Befehlsumfang der CNC-Prüfung nach PAL erweitern (Programmteilwiederholungen, Eckenverrundung, polare Koordinateneingabe, neue Zyklen, ...). Da PALturn als kostenlose Lernsoftware bei Berufsschülern eine überraschend hohe Verwendung findet, bleibt mir fast keine andere Wahl, als das Programm aufzurüsten, obwohl ich selbst in den letzten Jahren alles andere unterrichte, als CNC-Technik. Alle wichtigen Neuerungen, die ab 2009 Prüfungsbestandteile sein können, werde ich implementieren. (vgl. [Befehlsübersicht](#))

Allerdings: die für 2012 vorgesehenen Erweiterungen, nämlich das Arbeiten mit mehreren Achsen, bzw. angetriebene Achsen, werden mit (dem pixelorientertem) PALturn nicht mehr zu machen sein. (Also keine C-, Y- und B-Achse; Wer schreibt dann eine neue, vektororientierte Freeware?) PALturn wird dann „out“ sein (und ich auch).

Bis dahin, viel Spaß beim Lernen.




H. Klinkner

Idar-Oberstein im Frühling 2009

Hinweis:

Mit **F1** öffnet die Hilfe zum gerade editieren G-Gefehl. (Cursor muss in der jew. Zeile stehen.)

Mit **F11** kommst du immer zur Übersicht der Befehle von PAL-CNC.

P.S.: Obwohl bei PAL2009 kaum mehr gerechnet werden muss, vergesst mir den Pythagoras nicht. 

Unter <http://www.hubertklinkner.de/palmill> können immer die aktuellsten Versionen von PALmill und PALturn herunter geladen werden.

Mir bitte keine Mails an die Schule schicken, sondern nur an ... palmill@freenet.de

Nachdem sich mit den alten PALmill- und PALturn- Versionen von 1995 immer öfters lästige [Grafikprobleme](#) häuften (denn kein Grafikerhersteller hält sich mehr an die schönen, alten Konventionen und kein neues Betriebssystem erlaubt mehr schnelle, direkte, problemlose Zugriffe auf den Grafikspeicher), sah ich mich bedrängt, eine neue [CNC-Simulations-Software](#) zu schreiben, die problemlos unter WINDOWS (insb. NT) laufen sollte.

Nach einer Reihe von Um- und Irrwegen ist es mir doch gelungen, eine einigermaßen flotte Grafik zu programmieren. Wem's immer noch zu langsam ist, kann beim Simulieren die S-Taste drücken bzw. den Schieberegler „aufdrehen“: der Drehmeißel hopst dann in größeren Schritten über den Bildschirm. Zeitgewinn bringt auch die Option „Simulieren ab Cursor“. (Strg+F7) In der Hoffnung, dass innerhalb der nächsten Jahre diese Software störungsfrei laufen wird und keiner mir wieder jammern wird, er könne die CNC-Hausaufgaben nicht adäquat bearbeiten, wünsche ich viel Spaß und hoffentlich auch einen Lerngewinn beim Umgang mit PALturn

H. Klinkner

Idar-Oberstein, im Frühsommer 2002

Was ist neu ab Version 2.05 ?

- Nach dem Simulieren werden (Maus-)Messpunkte auf die Skalen projiziert.
- Entgegen der PAL-Konvention können nun Kreisinterpolationen alternativ auch mit R statt I+J programmiert werden.
- Exakte Treffsicherheit der falschen Programmzeile auch bei G41/42

^K Vorwort

[#] IHD_Vorwort

[§] Persönliches Vorwort

- Die Entf-Taste macht endlich, was sie soll.
- Parametrisierter Aufruf von PAturnl (durch Doppelklick auf die jew. .NC3-Datei). (Die Hilfedatei wird sofort gefunden.)
- Offset mit G48 (und G49; vgl. Hilfe)
- Verfeinerte Skaleneinteilung

Was ist neu ab Version 2.07 ?


- Bedienerfreundlicheres, betriebsicheres Speichern der NC-Dateien
- Schrift-Formatierung (← „Schnickschnack“)
- rot markierte Fehlerzeile
- simuliert im Eilgang schneller

Was ist neu ab Version 2.09 „PALturn+“ ?

- Erweiterte NC-Funktionen laut „PAL 2007“:
Koordinateneingabe im Absolut- oder Kettenmaß, sowie auch gemischt (XA, YA, ZA oder XI, YI, ZI),
Eckenverrundung, polare Koordinaten, Programmierung mit Anfangs- und Öffnungswinkel sowie Längen
und Radien, Programmsprünge, Spiegeln, ...
- mächtigere Zyklen
- Editor mit größerer Toleranz bezüglich Reihenfolge und Anzahl der Befehlswoorte
- größerer Simulationsbildschirm

Was ist neu ab Version PALturn+ 2.10?

- Parameterprogrammierung


[zur nächsten Seite](#)

^{\$K#} **Was das Programm tut** (Kurzbeschreibung von PALturn)

PALturn ist eine kostenlose Software zur Simulation einer CNC-Drehmaschine.

PALturn simuliert die Bearbeitung eines Drehteils auf einer 2-achsigen CNC-Schrägbett-Drehmaschine mit Bahnsteuerung.

Welchen CNC-Code verwendet es?

Die CNC-Programme müssen der DIN 66025 entsprechen.

Benutzt also Euer Tabellenbuch!

Zyklen können nur nach PAL verwendet werden. (s. *Tabellenbuch*, bzw. *Hilfe F1*)

(PAL heißt: Prüfungsaufgaben- und Lehrmittelentwicklungsstelle)

Wie arbeitet es?

Die Bearbeitung wird zweidimensional dargestellt; das Werkstück ist im Halbschnitt gezeichnet.

Es kontrolliert die wichtigsten (von Euch oft gemachten) Fehler.

Was muss ich als Bediener beachten?

Damit simuliert werden kann, muss sich ein CNC-Programm auf dem Desktop (Arbeitsoberfläche) befinden. Entweder wird es mit **Strg+N** neu geschrieben oder mit **Strg+O** geladen (bzw. *Pull-Down-Menü "Programm" anklicken*).

Auf einfache Weise können CNC-Programme nach PAL erstellt werden.


PALturn beachtet beim Abspeichern Eurer Programme folgende Konvention über die Erweiterung der Dateinamen, damit ihr eure Programme später mühelos wiederfindet:

Hauptprogramme: _____.NC3

Unterprogramme: _____.NC4 (erledigt PALturn automatisch)

Werden Unterprogramme verwendet, so müssen sie sich vor der Simulation (**F7**) bzw. vor der Syntaxprüfung (STRG+F7) auf dem Desktop befinden (im Gegensatz zu "richtigen" Maschinen).

Treten bei der Simulation Probleme mit der Grafik auf, so informiere dich unter Grafikprobleme !


zur nächsten Seite

^{\$} Programmbeschreibung

^K Programmaufbau; Programmstruktur; Aufgabe von PALturn

[#] IDH_ProgrammBeschreibung

\$#K **Wie schreibe ich ein CNC-Programm?**

Zuerst muss eine Editor-Fenster geöffnet werden ...

- neues leeres Fenster kreieren mit **Strg+N**, oder
- bereits gespeicherte Datei öffnen mit **Strg + O**

Der Editor (Software zur Texterstellung) wurde als vorgefertigtes Programmpaket in PALTURN übernommen; er unterstützt die gängigsten Editier-Befehle mit Hilfe der Maus oder auch der Sondertasten.

(vgl. [Editorfunktionen](#))

Es ist unwichtig, ob die Befehle groß- oder kleingeschrieben werden; auch sind Leerstellen nicht erforderlich.

Die Reihenfolge der Adressen muss ebenfalls nicht eingehalten werden; die nach DIN 66025 bzw. nach PAL erforderlichen Adressen müssen natürlich vollständig sein!

Zeilennummern brauchen nicht unbedingt geschrieben zu werden; das erleichtert auch das Löschen oder Kopieren von Programmsequenzen.

(Ästheten können mit "Format" diese Verschönerungen oder Tabellierungen nachträglich noch hineinzaubern. Selbst die Schriftart, -farbe und -größe können verändert werden.)

Kann man Programmzeilen ausblenden?

Beginnt die Programmzeile mit einem ";", einem "[", einem "%" oder mit einem "{", so wird diese Zeile beim Simulieren nicht interpretiert. Auch hinter einer Befehlszeile kann so ein *Kommentar* stehen.

Hinweis: Ein „(,“ oder ein“/“ ist ab Version 2.10 nur noch für die *Parameter*programmierung verwendbar, nicht mehr für die Kennzeichnung von Kommentaren.

Was kann ich während der Simulation tun?

Falls Dein Rechner nicht zu langsam ist (mind. 486er, 133 MHz), kann die Simulation oft kaum nachvollzogen werden. Deshalb kann man mit "e" auf Einzelsatz umschalten und den nächsten Befehl erst ablaufen lassen, wenn die RETURN-Taste gedrückt wird!

Auch lässt sich mit "h" zu jeder Zeit die Simulation anhalten und mit jeder folgenden Taste wieder fortfahren.

Die Simulationsgeschwindigkeit kann mit "S" gesteigert und mit "L" vermindert werden.

"m" stellt die maximale, „o“ die optimale Simulationsgeschwindigkeit ein.

(Das geht auch alles mit dem **Kontext-Menü**, das mit der rechten Maustaste aufgerufen werden kann.)

Mit ESC kann von jeder Situation aus die Simulation abgebrochen werden und dann wieder zum Editor gewechselt werden.

Treten Fehler auf, so wird anschließend die fehlerhafte Zeile im entsprechenden Fenster markiert.

„**Messen**“ im Sinne einer Qualitätsprüfung ist nur bedingt möglich: drücke im Werkstückbereich die linke Maustaste und die Koordinaten sind auf der Skala leichter bestimmbar (nur ±1 Pixel genau).

Welche nc-Befehle sind verwendbar?

siehe: [Übersicht der G- und M-Befehle](#)



[zur nächsten Seite](#)

^S Programmfunktionen

[#] IDH_ErsteSchritte


^K erste Schritte; CNC-Programm schreiben; Umgang mit dem Programm; messen; Kommentar;

K\$# [Detaillierte Funktionsbeschreibung von PALturn](#)

Die meisten Funktionen sind selbsterklärend. Falls dir doch was unklar ist, klicke auf eines der folgenden Detailinformationen:

- [neues CNC-Programm erstellen](#)
- [CNC-Programm speichern](#)
- [CNC-Programm öffnen](#)
- [CNC-Programm drucken](#)
- [CNC-Programm bearbeiten / editieren](#)
- [CNC-Programm formatieren / tabellieren](#)
- [Drehbearbeitung simulieren](#)
- [Hilfe benutzen](#)

Hast du dich mal irgendwo vertan, so drücke einfach die „war alles nur Spaß-Taste“ (ESC) oben links und du kommst wieder zurück, ohne was böses angestellt zu haben.


[zur nächsten Seite](#)

^K Bedienungsanleitung; Funktionsbeschreibung; Details zu PALturn


^S Detail_Funktionsbeschreibung

[#] IDH_DetailFunktionsbeschreibung

K\$# Neues CNC-Programm erstellen

Ein CNC-Programm, das du später testen und simulieren willst, muss zuerst auf den Bildschirm gebracht werden. Jedes Programm befindet sich also in einem eigenen Editor-Fenster.


Ein neues **Hauptprogramm** (-Fenster) wird erstellt mit einem der folgenden Befehle:

- mit **Menü: P**rogramm → **N**eu ... → **H**auptprogramm oder
- mit **Strg+N** oder
- Symbol  klicken

Ein neues **Unterprogramm** (-Fenster) wird erstellt mit einem der folgenden Befehle

- mit **Menü: P**rogramm → **N**eu ... → **U**nterprogramm .oder
- mit **Strg+Alt+N**

Anschließend verlangt PALturn die Eingabe der Unterprogramm-Nummer, die laut PAL 2-stellig sein muss. (Der später beim Abspeichern des Unterprogramms verwendete Dateiname ist für PAL-CNC uninteressant und darf beliebig lang sein; „richtige“ Steuerungen sind da pingelig.)


zur nächsten Seite

^K^{\$}# .. **Gespeichertes CNC-Programm öffnen**

Ein CNC-Programm, das du später testen und simulieren willst, muss zuerst auf den Bildschirm gebracht werden. Jedes Programm befindet sich also in einem eigenen Editor-Fenster.

Ein bereits gespeichertes **Hauptprogramm** (-Fenster) wird ins Desktop gebracht mit einem der folgenden Befehle:


- mit **Menü: P**rogramm → **Ö**ffnen ... → **H**auptprogramm oder
- mit **Strg+O** oder
- Symbol  klicken

(Der dann erscheinende Datei-Öffnen-Dialog ist auf die Extension :CN1 voreingestellt.)

Ein bereits gespeichertes **Unterprogramm** (-Fenster) wird erstellt mit einem der folgenden Befehle

- mit **Menü: P**rogramm → **Ö**ffnen ... → **H**auptprogramm.oder
- mit **Strg+Alt+O**

(Der dann erscheinende Datei-Öffnen-Dialog ist auf die Extension :CN2 voreingestellt.)


zur nächsten Seite


^K Programm öffnen; Öffnen, Laden

^{\$} Programm_oeffnen

[#] IDH_Programm_oeffnen

K\$# CNC-Programm speichern

Ein von dir geschriebenes (und getestetes) CNC-Programm kann auf externe Datenträger gespeichert werden mit ...


- mit **Menü: Programm → Speichern** .oder
- mit **Strg+S** .oder
- Symbol  klicken

Wenn das Programm vorher noch nicht gespeichert war, wird automatisch zu

- mit **Menü: Programm → Speichern unter**

Hinweis: Die Vergabe der Dateinamen ist beliebig (im Gegensatz zu „richtigen“ Maschinen, meist nur das Format „P001234“ zulassen). Die von PALturn gewünschte Extension :NC1 bzw. .NC2 wird automatisch zugefügt. → Bitte keine Dateierweiterung eingeben!

(Wer den Programmtext mit einem Standardeditor bearbeiten will, stellt erschreckt fest, dass die erste Zeile ganz wirres Zeug enthält. Das erklärt sich dadurch, dass die von dir gemachten Werkstück- und Werkzeugeinstellungen hier mit abgespeichert wurden. Ähnlich ist's auch bei „richtigen“ Steuerungen; sie speichern auch nicht sichtbar interne Kenngrößen ab, die nicht verändert werden dürfen.)


[zur nächsten Seite](#)

^K Programm speichern; Speichern

^S Programm Speichern

[#] IHD_ProgrammSpeichern

^{K\$#} CNC-Programm exportieren

Wenn ein von dir geschriebenes (und getestetes) CNC-Programm auf externe Datenträger gespeichert wird, erhält es automatisch die Endung *NC3 und wird so im sog. RichText-Format abgelegt und kann damit aber nicht von einer richtigen CNC-Maschine gelesen werden.


Wenn du dein Programm für eine CNC-Maschine speichern willst (Achtung: die PAL-Zyklen passen dann aber nicht für die Steuerung!), so kannst du es mit EXPORTIEREN im sog. ASCII-Code abspeichern. Solche Dateien können von jeder CNC-Steuerung gelesen werden.

mit ...

Menü: Programm → Exportieren .

Es erscheint zuerst eine Sicherheitsabfrage. Drücke JA, dann kannst du im folgenden Speichern-Dialogfeld den Dateinamen eingeben.

Passt die Extension *.CNC nicht für deine CNC-Steuerung, dann schreibst du den Dateinamen mit samt der gewünschten Extension. (Statt *.CNC ist dann deine Extension voreingestellt.)


zur nächsten Seite

^K Programm exportieren; Exportieren; MaschinenProgramm

^{\$} Programm Exportieren

[#] IHD_ProgrammExportieren

AK\$# **CNC-Programm ausdrucken**

- mit **Menü: P**rogramm → **D**rucken.oder
- mit **Strg+P**

Tipp: Wer einen noch schöneren Ausdruck seines CNC-Programms haben will, kann

1. den ganzen Programmtext mit Strg+A in den Zwischenspeicher von WINDOWS kopieren und dann
2. eine Textverarbeitungssoftware starten und mit Strg+V den Programmtext in ein (leeres) Dokument einfügen.
3. Formatierungen nach eigenem Gusto durchführen
4. evtl. den Simulations-Bildschirm mit Alt+Druck zwischenspeichern, dann z.B. in Paint einfügen, begrenzen, bearbeiten und dann mit Strg+C und Strg+V (bzw. Alt+e, g) in die Textverarbeitung einfügen
5. drucken



vgl. auch Werkstück ausdrucken

^A Drucken

^K Programm ausdrucken; ausdrucken; drucken

^S ProgrammAusdrucken

[#] IHD_ProgrammAusdrucken

AK\$# [Werkstück ausdrucken](#)

Geschieht mit **Menü: Programm → Werkstück drucken**
Ist aber sehr spartanisch. (Wozu soll's gut sein?)

→ vgl. besser den Grafik-Kopier-Tipp bei CNC-[Programm ausdrucken](#) !



[zur nächsten Seite](#)

^A Drucken

^K Werkstück ausdrucken; ausdrucken; drucken


^S WerkstückAusdrucken

[#] IHD_WerkstückAusdrucken

K\$# Editorfunktionen

Der verwendete Programmeditor (Standard-Textverarbeitungsprogramm) arbeitet ähnlich wie herkömmliche Editoren. Richtige CNC-Maschinen haben bis heute (mir unverständlicherweise) umständliche „bedienerunfreundliche“ Möglichkeiten, Programmzeilen zu schreiben.

Cursor bewegen	Cursortasten	linke Maustaste
" " wortweise	" + Strg	
" " an Zeilenanfang	Pos1	
" " an Zeilenende	Ende	
" " Seite nach oben	Bild-auf	oder Bildlaufleisten benutzen
" " Seite nach unten	Bild-ab	" " "
" " an Programmanfang	Strg + Pos1	" " "
" " ans Programmende	Strg + Ende	" " "
Einfügemodus Ein/Aus	Einfg	
Block markieren	Shift + Cursor	Maustaste gedrückt
" zwischenspeichern	Strg + C	Menü "Bearbeiten"
" kopieren	Strg + V	" "
" ausschneiden	Strg + X	" "
" löschen	Entf " "	
" verschieben	Strg + X dann Strg + V	oder „drag & drop“
Zeichen löschen	Rücktaste	
Letzten Schritt rückgängig machen: Strg + Z		
Textstelle suchen	Strg + F dann: F3	Menü "Bearbeiten"
Fenster schließen	Alt + F4	[X] "anklicken"
Fenstergröße verändern	Cursor Rahmen "ziehen"	
maximale Fenstergröße bzw. „kacheln“		Strg + F5 Menü "Fenster"
Fenster in Ausgangsgröße bzw. „überlappen“		Umsch + F5 [□] "anklicken"
nächstes Fenster nach oben	F5	bzw. Rahmen "anklicken"
<u>Formatieren</u>	F9	


zur nächsten Seite

^K Editorfunktionen; Umgang mit dem Programm-Editor; F3-Taste

^S Editorfunktionen

[#] IHD_Programmeditor

K\$# **Programmtext formatieren, tabellieren, nummerieren**

a)

Durch ein Tabellieren und Nummerieren kann der Programmtext übersichtlich dargestellt werden ...

- mit **Menü: Bearbeiten → Tab_Formatieren** .oder
- Symbol  klicken oder
- F9 drücken
- Kontext-Menü

Es folgt dann eine Dialogfeld, in dem man zusätzlich (zu: Reihenfolge und Abstand der Adressen) alle Adressen in Großbuchstaben, eine Zeilennummerierung und eine Tabellierung wünscht.


Die Einstellung der Schrittweite zur Nummerierung wird nur aktiv, wenn Zeilennummerierung angeklickt wurde.


Zeilennummerierung und Tabellierung können später wieder rückgängig gemacht werden, wenn man bei erneutem Formatieren diese Optionsfelder frei lässt.

(Der CNC-Steuerung ist es natürlich egal, in welcher Reihenfolge und Form die Befehle erscheinen, aber durch die Formatierung sind sie leichter lesbar. Zeilen-Nr. machen nur Sinn, wenn eine Steuerung mittels Hochsprache Sprung- und Wiederholbefehle akzeptiert.)

b)

Will man unbedingt Textstellen des Programms besonders herausstellen, also den Programmtext in **Schrift, Größe, Farbe** usw. formatieren, so muss man...

- mit **Menü: Bearbeiten → Schrift_Formatieren** .oder
- das Symbol  klicken.


zur nächsten Seite

^K Programm formatieren; Tabellieren; F9-Taste; formatieren

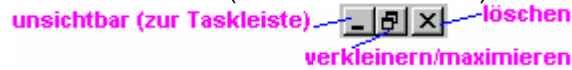
^S TextFormatieren

[#] IHD_TextFormatieren

K\$# Umgang mit mehreren Fenstern


Der Menüpunkt **Fenster** -- ... bezieht sich ausschließlich auf die Fenster innerhalb des Hauptfensters von PALturn, also alle Haupt- und Unterprogramme, die sich auf dem Desktop befinden. Das Simulationsfenster (**F8**) oder der Hilfetext sind damit nicht gemeint.


Max. können 10 Programmfenster offen sein. (Das müsste reichen.)



- Mit **F5** kann man zu nächsten Fenster schalten.
- Mit **Umsch+F5** können sie überlappt dargestellt werden.
- Mit **Strg+F5** können sie nebeneinander (gekachelt) werden.
- **F4** schließt das momentan aktive Fenster (Vorsicht nicht Alt+F4, denn das schließt PALturn !!) Wurde das Fenster nach dem letzten Speichern verändert, so erscheint eine Warnung mit letzter Chance, es noch zu speichern.

Tipp: Soll das Hilfe-Fenster im Zugriff bleiben, so verkleinere das PALturn-Hauptfenster und überlappe beide (mittels Maus am oberen blauen Kopf greifen und schieben).

Das Simulationfenster kann mit **F8** bzw. mit  aufgerufen werden, falls schon vorher min. einmal eine Simulation durchgeführt wurde. Mit F8 kann auch zwischen Simulation- und Editorfenster hin- und her geschaltet werden (eine evtl. arbeitende Simulation wird kurzerhand abgebrochen).


zur nächsten Seite

^K Fenster; Kacheln; F5-Taste; F8-Taste; F4-Taste


^S Fenster

[#] IHD_Fenster

K\$# CNC-Programm simulieren

Um festzustellen, ob ein CNC-Programm richtig läuft, kann die Drehbearbeitung simuliert werden. Dies geschieht ...

- mit **Menü: Ausführen → Simulieren**.oder
- mit **F7** .oder
- Symbol  klicken
- Kontext-Menü

Um (Syntax-)Fehler im Programm schon vorher zu eliminieren, ist es ratsam, mit **STRG+F7** bez. mit der -Taste evtl. Fehler auszumerzen. Im Falle eines Fehlers erscheint eine Meldung und die fehlerhafte Zeile im Programmeditor wird durch kurzzeitiges Blinken angezeigt.

Während des Simulierens kann das Programm mit der Taste **h** bzw. mit dem Button „anhalten“ angehalten werden. Weiter geht's analog mit dem gleichen Button bzw. mit der **w**-Taste.


Auch kann mit den Tasten **f** bzw. **e** Folge- oder Einzelsatz eingestellt werden.

Die Simulationsgeschwindigkeit kann verändert werden:

- (grob) mit dem Schieberegler oder
- (fein) mit den Tasten **s** für schneller und **L** für Langsamer
- Mit der Taste **m** kann die maximale Simulationsgeschwindigkeit eingestellt werden.
- Mit der Taste **o** kann die optimale Simulationsgeschwindigkeit (ohne Pixelsprünge) eingestellt werden.

Tipp: **WINDOWS-Task-Leiste verkleinern oder ganz nach oben setzen!**

Wurde nach der Simulation weiter programmiert, kann man die bisherigen Simulationsschritte überspringen, in dem man den Cursor auf die Programmzeile stellt, ab der erst simuliert werden soll. Dies geschieht ...


- mit **Menü: Ausführen → Simulation ab Cursor**.oder
- mit **Umsch + F7** .oder
- Symbol  klicken
- Kontext-Menü

Mit ESC kann man (wie überall) die Simulation abbrechen. (Mit der F8-Taste, die zwischen Editor und Simulationsfenster hin und her schaltet, geht's ohne Halt noch schneller.)

Mit der linken Maustaste kann man orthogonale Linien zu den X- und Z-Skalen erzeugen um evtl. **Maße zu überprüfen**. (Die Messgenauigkeit von ± 1 Pixel ist allerdings sehr dürftig!)

Besonderheit in der Befehlszeile links unten:

PALturn lässt auch „Nicht-PAL-Befehle“, wie z.B. G03 X.. Y.. R.. zu. Wenn man beim Simulieren die Umwandlung in PAL-Format sehen will, muss man vorher im Menü „Ausführen“ die Option „BefehlsZeile in PAL“ aktivieren. Mit einem Doppelklick auf die Zeile kann man sogar deren Inhalt in den Zwischenspeicher ablegen.


[zur nächsten Seite](#)

^K Programm simulieren; Simulation; F7-Taste; STRG+F7-Taste; messen

^S ProgrammSimulieren

[#] IHD_ProgrammSimulieren

K\$# Hilfe benutzen

Die Arbeitsweise von PALturn erklärt sich weitgehend selbst. Das mitgelieferte Hilfesystem (Datei: „PALturnH.hlp“) ermöglicht den Zugriff auf Informationen auf zwei Wegen:

a) **Direkthilfe**

Mit der **F1**-Taste gelangt man (je nach momentaner Situation, Menüpunkt oder Fenster) direkt in den Hilfetext.

Tipp: Wenn man sich im Editor befindet und z.B. (mit einem Doppelklick) einen Befehl z.B. G41 markiert hat und dann die F1-Taste drückt, erhält man sofort Zusatzinformationen zu diesem Befehl. (So verwöhnt man Schüler, die zu faul sind, in ihr Tabellenbuch zu schauen.)

Falls du während des Programmschreiben nichts markiert hast und F1 drückst, so sucht PALturn nach dem ersten Befehlsword in der Cursorzeile und springt in die hoffentlich richtige Hilfe hinein. (Auf den Trick bin ich mächtig stolz, obwohl ich fast 2 Stunden herumprogrammieren musste, bis er endlich richtig klappte!)

b) Hilfe mittels **Inhalte**, **Index** oder **Suchen**

Mit **Umsch + F1** öffnet sich eine Dialogfenster mit drei Registern:

- Inhalte: eine Art Inhaltsverzeichnis der Hilfetexte
- Index: Zugriff auf Hilfeseiten über „Stichwörter“
- Suchen: mittels Textpassagen

Wem diese Hilfehilfe nicht reicht, der drücke jetzt noch mal F1 und gelangt so in den Hilfe-Hilfe-Text von Windows!



zur nächsten Seite

^K Hilfe; Direkthilfe; F1-Taste

^{\$} HilfeBenutzen

[#] IDH_HilfeBenutzen

^{§#K} Grafikprobleme

... gab's u.U. bei der alten DOS-Version. Mit dieser Version dürfte es (hoffentlich) keine größeren Probleme mehr geben. Dennoch sollte man einige Voraussetzungen beachten, damit die Simulationsfreude ungetrübt bleibt:

- Bildschirmpfarben sollten möglichst auf 256 Farben eingestellt sein
- Bildschirmauflösung optimal: 800x600
- Die Taskleiste sollte verschwunden sein (vgl. WINDOWS – Einstellungen -...) oder notfalls ganz nach oben verschoben werden

Falls es immer noch Probleme gibt, H. Klinkner Bescheid sagen.



zur nächsten Seite

[§] Grafikprobleme

[#] IDH_Grafikprobleme

^K Grafikprobleme; Bildschirmeinstellungen

AK\$# **Details zum Programmieren von CNC-Maschinen nach PAL (Syntaxregeln)**

CNC heißt **c**omputerizes **n**umerical **C**ontrol, als die computerunterstützte zahlenmäßige Steuerung von Maschinen.

Die Maschine, die über einen Computer, Messsysteme u.a.m. verfügt, wird über Befehlsätze gesteuert. Ein CNC-Programm besteht aus vielen solcher Sätze.
In einem CNC-Befehlsatz erscheinen Wegbefehle (Geometrie) und/oder Schaltbefehle (Technologie)


- N** dient zur Nummerierung der Programmzeilen (nicht unbedingt erforderlich)
- G** (engl. GO) beschreibt die verschiedenen Wegbefehle
- X,Y,Z,I,J und K** beschreiben die verschiedenen Koordinaten
- F** (engl. Feed) steht für den Vorschub bzw. Vorschubgeschwindigkeit
- S** (engl. Speed) steht für die Drehzahl bez. für die Schnittgeschwindigkeit
- T** (engl. Tool) steht für das Werkzeug
- M** (engl. miscellaneous = vermischt) steht für verschiedenste Schaltbefehle (an die SPS)
- L** steht für den Aufruf eines Unterprogramms

Hol' dir dein Tabellenbuch zur Hand!

Den jew. Buchstabe nennt man Adresse; er benötigt immer einen numerischen Wert (z.B. T01).
Beides zusammen nennt man ein Wort.

Die Reihenfolge der Befehls Worte ist beliebig, der G-Befehl steht sinnvoller Weise am Anfang. Auf Groß- und Kleinschreibung braucht nicht geachtet zu werden.

Die PAL-CNC ist pingelig und „einfach gestrickt“: Es darf ein Befehl immer nur einmal pro Befehlszeile erscheinen. Andere Steuerungen erlauben problemlos mehrere G-Befehle; z.B. G90 G41 G01 X20 Y4
...


[zur nächsten Seite](#)

^A M-Befehle

^K CNC-Befehle; CNC-Grundlagen; Adresse; Befehl

^S DetailsCNCProgrammierung

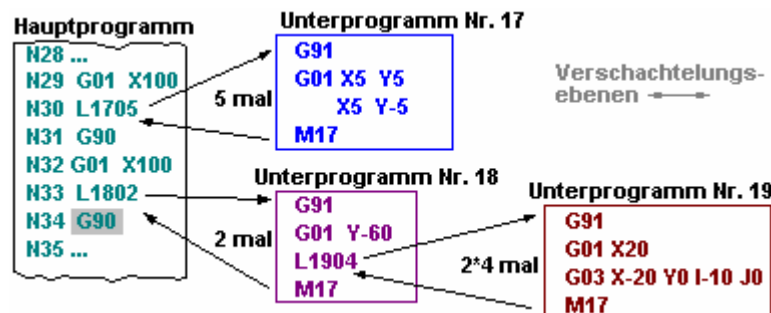
[#] IDH_DetailsCNCProgrammierung

#SK Unterprogramme

Sie dienen der **übersichtlicheren** Programmierung und werden oft eingesetzt, wenn Programmteile öfters **wiederholt** werden.

(Apropos: Wiederholungen können auch in Hauptprogrammen vorkommen, indem man Zeilen in Hochsprache programmiert. Die CNC nach PAL kennt so was nicht!)

Sie sind keine eigenständigen Programme und können nur von einem Hauptprogramm aus gestartet werden. Sie können selbst wieder andere Unterprogramme aufrufen (Die Verschachtelungsebene ist bei den meisten CNC-Steuerungen auf ca. 10 Ebenen begrenzt. PALturn tut's unbegrenzt (solange der Stack = Speicherbereich nicht überläuft). Unter PALturn ist sogar ein Selbstaufruf (= rekursives Programmieren) möglich.



Unterprogramme werden bei PAL mit **L** aufgerufen, gefolgt von der zweistelligen Programm-Nr. und der zweistelligen Anzahl ihrer Aufrufe. z.B.:

z.B.: **L1302** ruft das Unterprogramm Nr. 13 auf und arbeitet es 2x ab (1 Wiederholung).

Tipp: Da im Unterprogramm meist inkremental programmiert wird (Ausnahme: G54-G59), darf nicht vergessen werden, mit **G90** die Absolutprogrammierung wieder einzuschalten!!

Bei PALturn muss allerdings beim Programmablauf (hier: Simulation) das Unterprogramm auf dem Bildschirm sein. Der eigentliche Dateiname des Unterprogramms wird von PALturn dabei ignoriert; wichtig ist nur die 2-stellige Nr., die beim Erzeugen des Unterprogramms mit Strg + Alt + N vergeben werden musste.

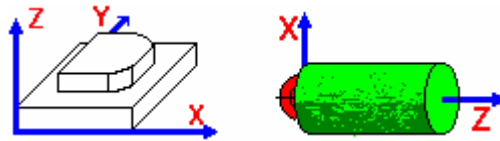
(Will man diese Nr. ändern, so bleibt nur als einzige Möglichkeit, ein neues Unterprogramm mit der neuen Nr. zu kreieren und dann aus dem alten Unterprogramm mit Strg+A und Strg+C den gesamten Programmtext in das neue Unterprogramm (mit Strg+V) zu kopieren.) Bei „richtigen“ Steuerungen zählt nur der Dateiname.)

Am Ende eines jeden Unterprogramms muss der Befehl M17 (=Rücksprung zum aufrufenden Programm) stehen.

➡
zur nächsten Seite

^K^S# Die Koordinaten X, Y, Z

...kennzeichnen die Lage eines Punktes (z.B. des Wkz-Mittelpunktes) im Raum.
Die Koordinatenwerte werden in mm angegeben (z.B. Y12,345)



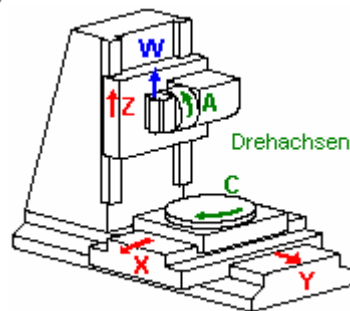
Laut DIN 66217 ist die **Z immer parallel zur Hauptspindelachse** der Maschine;
X,Y und Z stehen nach der Drei-Finger-Regel der rechten Hand senkrecht zueinander.
X ist dabei immer die Hauptbewegungsachse der Maschine

Vgl. auch absolute und relative Koordinaten!

Vgl. auch die verschiedenen Koordinaten-Nullpunkte!


Nicht in PALturn:

Zusätzliche Werkzeugschlitten, die sich parallel zu den Achsen X, Y und Z bewegen werden
entsprechend mit U,V und W bezeichnet. (z.B. eine zusätzliche Pinolen-Bewegung in Richtung
Richtung Z erhält die Koordinate W)



Befinden sich zusätzlich noch Schwenkachsen an der Maschine, so erhalten diese entsprechend den
Achsen X,Y, Z die Koordinaten A,B und C. (Drehrichtung: vgl. „Korkenzieher-Regel“)

Steuerungen lassen auch **Polare Koordinaten** (Radius und Winkel) zu.


zur nächsten Seite

^K Koordinaten; X;Y;Z; U; V; W; A; B; C

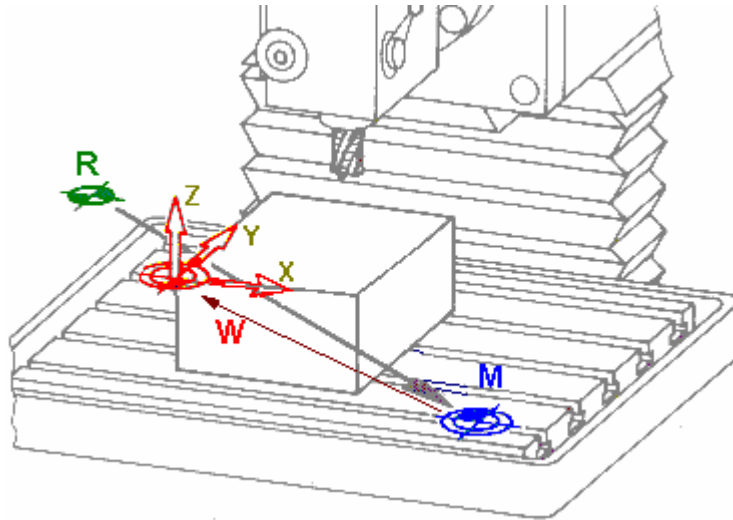
^S Koordinaten

IDH_Koordinaten

K§# **Bezugspunkte, Nullpunkte**


Die in einem CNC-Befehl vorhandenen Koordinatenangaben beziehen sich (meistens) auf den sog. **Werkstücknullpunkt**. Dieser wurde vom Maschinenbediener definiert; es kann aber auch während des Programmablaufes verändert werden. vgl. G54-G59

Es gibt verschiedene Nullpunkte:



Der **Maschinennullpunkt** ist vom Hersteller festgelegt und unveränderlich (und zum Programmieren uninteressant; lediglich Werkzeugwechselforgänge müssen darauf zugreifen).

Damit aber beim Starten der Maschine das Wegmesssystem „weiß“ an welcher Stelle es steht, ist eine sog. Referenzfahrt zu machen. Damit ist der **Referenzpunkt** festgelegt. Zum Simulieren mit PALturn ist lediglich der Werkstücknullpunkt interessant. Dieser liegt links unten auf der Werkstückoberfläche und kann mit dem Menü-Punkt Ausführen --> Voreinstellungen bzw. mit Strg+W verändert werden.


[zur nächsten Seite](#)

^K Bezugspunkte; Nullpunkte; Referenzpunkt

[§] Bezugspunkte

[#] IHD_Bezugspunkte

<u>G0</u>	Verfahren im Eilgang
<u>G1</u>	Linearinterpolation im Arbeitsgang
<u>G2</u>	Kreisinterpolation im Uhrzeigersinn
<u>G3</u>	Kreisinterpolation entgegen dem Uhrzeigersinn
<u>G04</u>	Verweildauer
<u>G09</u>	Genauhalt
<u>G14</u>	Konfigurierten Werkzeugwechsellpunkt anfahren
<u>G18</u>	Drehebeneanwahl
<u>G17</u>	Stirnseitenbearbeitungsebenen
<u>G19</u>	Mantelflächen/Sehnenflächenbearbeitungsebenen
<u>G22</u>	Unterprogrammaufruf
<u>G23</u>	Programmteilwiederholung
<u>G29</u>	Bedingte Programmsprünge
<u>G30</u>	Umspannen/Gegenspindelübernahme
<u>G40</u>	Abwahl der Schneidenradiuskorrektur SRK
<u>G41/42</u>	Schneidenradiuskorrektur SRK
<u>G50</u>	Aufheben von inkrementellen Nullpunkt-Verschiebungen und Drehungen
<u>G53</u>	Alle Nullpunktverschiebungen und Drehungen aufheben
<u>G54-57</u>	Einstellbare absolute Nullpunkte
<u>G59</u>	Inkrementelle Nullpunkt-Verschiebung kartesisch und Drehung
<u>G61</u>	Linearinterpolation für Konturzüge
<u>G62</u>	Kreisinterpolation im Uhrzeigersinn für Konturzüge
<u>G63</u>	Kreisinterpolation entgegen dem Uhrzeigersinn für Konturzüge
<u>G70</u>	Umschaltung auf Maßeinheit Zoll (Inch)
<u>G71</u>	Umschaltung auf Maßeinheit Millimeter (mm)
<u>G90</u>	Absolutmaßangabe einschalten
<u>G91</u>	Kettenmaßangabe einschalten
<u>G92</u>	Drehzahlbegrenzung
<u>G94</u>	Vorschub in Millimeter pro Minute
<u>G95</u>	Vorschub in Millimeter pro Umdrehung
<u>G96</u>	Konstante Schnittgeschwindigkeit
<u>G97</u>	Konstante Drehzahl

PAL Bearbeitungszyklen

<u>G31</u>	Gewindezyklus
<u>G32</u>	Gewindebohrzyklus
<u>G33</u>	Gewindestreihgang
<u>G80</u>	Abschluss einer Bearbeitungszyklus-Konturbeschreibung
<u>G81</u>	Längsschruppzyklus
<u>G82</u>	Planschruppzyklus
<u>G83</u>	Konturparalleler Schruppzyklus
<u>G84</u>	Bohrzyklus
<u>G85</u>	Freistichzyklus
<u>G86</u>	Radialer Stechzyklus
<u>G87</u>	Radialer Konturstechzyklus
<u>G88</u>	Axialer Stechzyklus
<u>G89</u>	Axialer Konturstechzyklus

In einer Befehlszeile kann nun (bei PALturn+) ein G- oder auch ein M-Befehl mehrmals vorkommen!

siehe auch: Übersicht der M-Befehle: (=Ein- oder Ausschaltbefehle für die SPS)

^A M-Befehle

^S GundMBefehle

^K G; Befehle im Überblick; M; PAL-Zyklen

[#] IHD_GundMBefehle



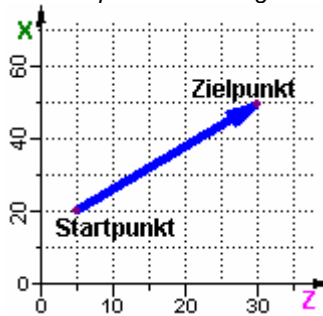
zur nächsten Seite

^{K#}Der G00-Befehl

Werkzeug verfährt im Eilgang.

= = > „Punktsteuerung“ (aber: Maschinen verfahren heutzutage meist linear interpoliert.)

Punktsteuerung heißt: Kein Werkzeug im Eingriff; X- und Y-Bewegung sind nicht miteinander abgestimmt; zur der Zielpunkt ist wichtig.



G00 X.. Y.. Z..

Mindestens ein Koordinatenwert (Ziel) muss angegeben sein.

z.B.: G00 X50 Z30



zur nächsten Seite

^K G00;G0; Eilgang

[#] IDH_Eilgang

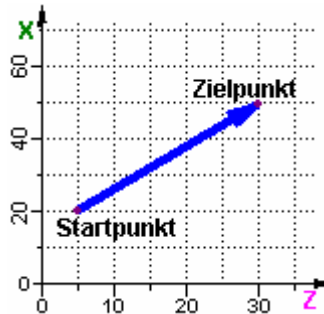
^S G00-Befehl

K# Der G01-Befehl (Linearinterpolation)

Werkzeug verfährt mit Vorschubgeschwindigkeit (und ist im Eingriff).

==>

„Bahnsteuerung“ (; X-, Y- und evtl. die Z-Bewegung sind miteinander abgestimmt, „interpoliert“)

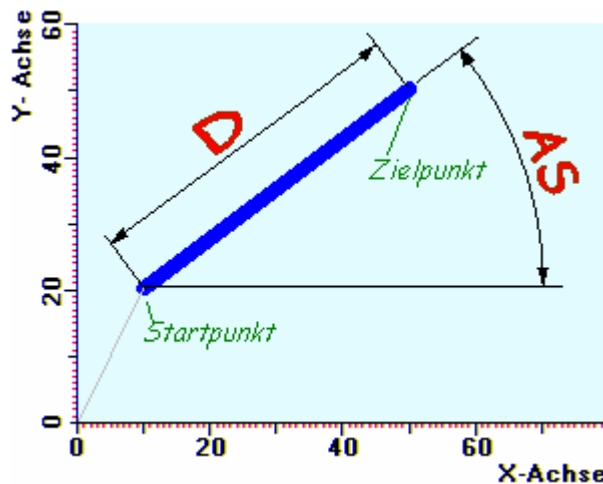


G01 X.. Y.. Z..

Mindestens ein Koordinaten wert (Zielpunkt) muss angegeben sein.

z.B.: G01 X50 Z30

Ab 2009 sind weitere Eingabemöglichkeiten erlaubt :



Statt G01 X50 Y50 bzw. G01 XI40 YI30

G01 D50 AS36,87

Auch Kombinationen sind möglich:

G01 X50 D50 H1 oder **G01 YI30 D50 H1** oder **G01 X50 AS36,87**

In den Fällen, wo es zwei Lösungen gibt, muss mit der Adresse H die Linie eindeutig festlegen:

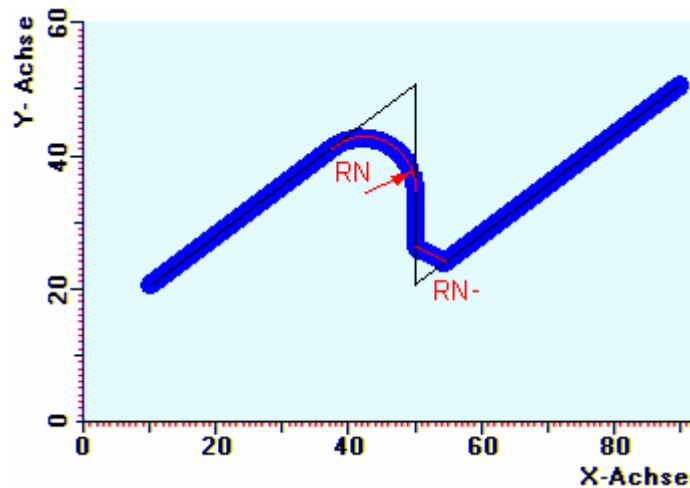
H1 bedeutet: kleiner Anstiegswinkel zur X-Achse

H2 bedeutet: großer Anstiegswinkel

Zwei zusammenhängende Geraden können durch ein Anhängen einer RN-Adresse miteinander verrundet werden bzw. (bei neg. RN-Wert) eine Fase erhalten; es entsteht ein Übergangselement :

^K G01;G1; Linearinterpolation; Verrundung; RN-Befehl; Fase; D-Adresse; AS-Adresse; Übergangselement
IDH_G01

^S G01-Befehl



G01 X50 Y50 **RN8**
 G01 Y20 **RN-5**

← Radius mit tangentialem Übergang

← Radius neg.: Fasenbreite

G01 X90 Y50

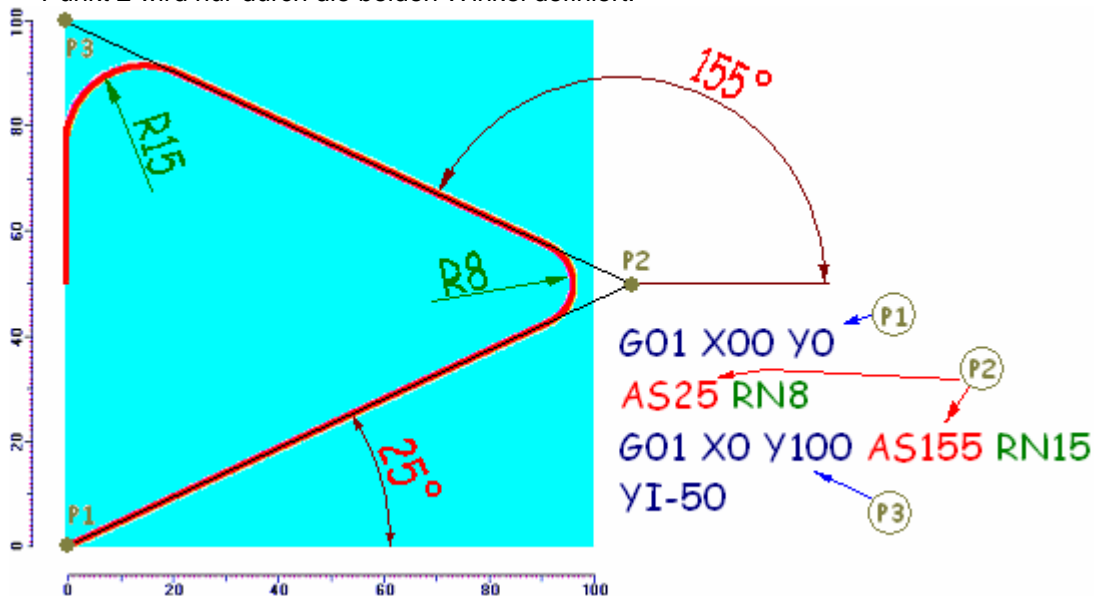
alternativ:

G01 D50 AS36,87 **RN8**

Y20 **RN-5** X90 YI30 (Streng nach PAL müsste X90 YI30 in eine neue Zeile.)

noch'n Beispiel:

Punkt 2 wird nur durch die beiden Winkel definiert.



Tip: Die Fehlermeldungen im Zusammenhang mit RN beziehen sich u.U. auf die Folgezeilen. →
 Schalte bei Interpretationsproblemen vorübergehend den RN-Befehl mit ; aus.

Achtung: RN muss immer am Ende der Befehlszeile stehen, weil PALturn auch die Verrundung von zwei Linearwegen in einer Zeile erlaubt. Deshalb kann's leider zu Fehlermeldungen kommen, wenn man die originale Reihenfolge von manchen PAL-Lösungen verwendet. (PAL und PALturn vertragen sich hier nicht ganz 100%-zig; ich will's aber nicht mehr umprogrammieren.)



zur nächsten Seite

K#§ **Der G02- bzw. G03-Befehl** (Kreisinterpolation)

Werkzeug verfährt auf einem Kreisbogen mit Vorschubgeschwindigkeit (und ist im Eingriff).
 == >

„Bahnsteuerung“ (X-, Z- und evtl. die Z-Bewegung sind miteinander abgestimmt,
 „interpoliert“)

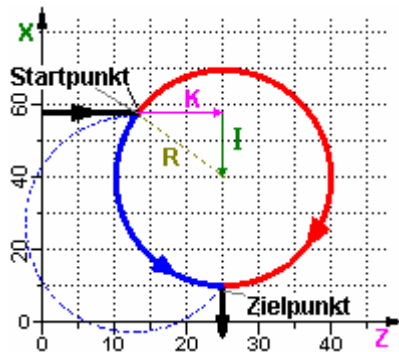
G02: Bewegung im Uhrzeigersinn

G02 X.. Z..I..K

G03: Bewegung entgegen dem Uhrzeigersinn

G03 X.. Z..I..K

Benötigt werden die Zielkoordinaten X und Z sowie die Mittelpunktskoordinaten des Kreisbogens.
 Dieser wird inkremental eingegeben, also ausgehend vom Startpunkt.
 (Einige Steuerungen – bes. bei Waagrecht-Fräsmaschinen – erlauben eine absolute Eingabe von I, und J.)

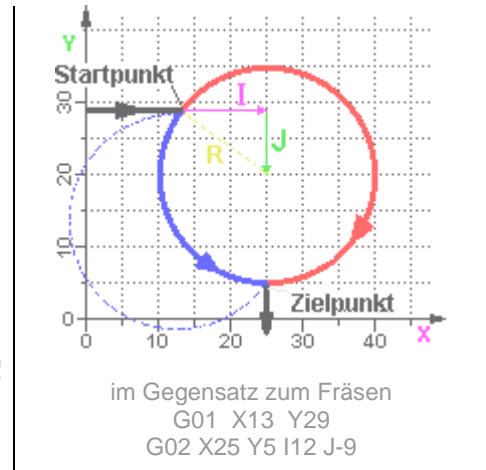


Beachte beim Drehen: X-Wert = doppelter Werkstückradius!

G01 X58 Z13
G02 X10 Z25 I-9 K12

oder :

G01 X58 Z13
G03 X10 Z25 I-9 K12



im Gegensatz zum Fräsen
 G01 X13 Y29
 G02 X25 Y5 I12 J-9

Drehen mit G91: X = Werkstückradius
 G02 X-24 Z12 I-9 K12

Im Sonderfall, dass der Zielpunkt identisch mit dem Startpunkt ist, verfährt PALturn (aber nicht jede Steuerung) einen Vollkreis. Schlimmstenfalls 2 Halbkreise programmieren!

Die Werte von I und K in dem obigen Beispiel sind nach Herrn Pythagoras zufällig ganzzahlig. Sie müssen bis auf die 3. Kommastelle genau ausgerechnet werden. PALturn ist da genau so rabiat wie dein „Gesellenprüfungsauswerter“: bei Abweichungen von mehr als 3 Mikrometer bringt es `ne Fehlermeldung. Andere Steuerungen sind da z.T. phlegmatischer: sie machen irgendwie das Beste `draus und verfahren am Schluss linear an den Zielpunkt !?!)

Natürlich gibt's bei „richtigen“ Steuerungen bedienerfreundlichere Möglichkeiten, diese obige Bahn zu programmieren, ohne mühevoll rechtwinklige Dreiecke zu suchen: einfach durch die Eingabe der Radius (und des Zielpunktes natürlich). Aber so gibt's zwei Möglichkeiten, wie du aus obiger Zeichnung siehst. Bei der blau gestrichelten Variante (und bei der roten Bahn) ist der Innenwinkel des Bogens größer als 180 Grad; für diese Bahn müsste man zur eindeutigen Unterscheidung den Radius negativ eingeben. (Wer CAD kann, kennt diese Möglichkeiten.)

z.B. G03 X25 Z10 **R15** u. für die blau gestrichelte Variante G03 X25 Z10 **R -15**

~~Dies' geht leider nicht bei PAL. Peinliches Pech für Pythagorasphobisten!~~

Dieses ist ab der PALturn-Version 2.05-(3) möglich, sollte aber aus pädagogischen Gründen nicht in der Schule verwendet werden.

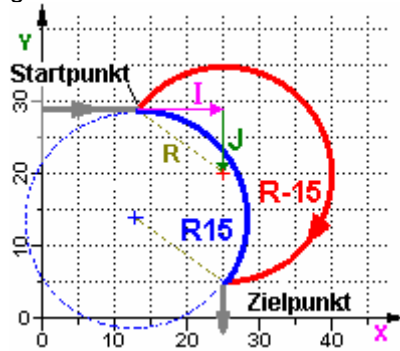
Selbst die Angabe des Zielpunktes und eines Punktes auf dem Bogen reicht vielen Steuerungen aus, daraus den Bogen zu ermitteln. Ganz verschwiegen werden soll hier, dass es auch so schöne, mächtige Funktionen wie „tangentiale Übergänge“ oder die „Eckenverrundung“ gibt. Muss man nicht in der Schule haben.

^K G02; G03;G2;G3 Kreisinterpolation; Radien; R; Befehlszeile in PAL; RN-Befehl; AO; Übergangselement

IDH_G02

[§] G02-Befehl

PAL2009 erlaubt nun „bedienerfreundlichere“ Koordinateneingabe:
 Statt I und J wird nur der Radius **R** bzw. nur der Öffnungswinkel **AO** eingegeben.
 Da es 2 Lösungsmöglichkeiten gibt, muss die Variante, die über 180° geht, mit einem Minus-Zeichen gekennzeichnet werden. Also R- bzw. AO-



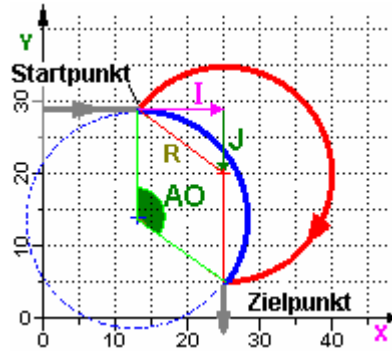
G01 X13 Y29

G02 X25 Y5 R-15

bzw. G02 X25 Y5 R15 O2

bzw..

G02 X25 Y5 R 15



G01 X13 Y29

G02 X25 Y5 AO -126,87

bzw. G02 X25 Y5 AO 233,13

bzw. G02 X25 Y5 AO 126,87 O2

G02 X25 Y5 AO 126,87

Alternativ kann man auch das Bogenlängenkriterium „O“ benutzen: wenn es auf 2, statt auf 1 steht, dann ist (trotz pos. R oder pos. AO) der größere Bogen gemeint.

Hinweis: Genau wie der G01-Befehl können auch die G2- bzw. G3-Befehle mit **RN** (Verrundung) oder RN-Übergangselement erweitert werden.

Tipp: Die Fehlermeldungen im Zusammenhang mit RN beziehen sich u.U. auf die Folgezeilen. →
 Schalte bei Interpretationsproblemen vorübergehend den RN-Befehl mit ; aus.



zur nächsten Seite

K#§ G04 Verweilzeit


... sorgt für eine Verweilzeit, die in Sekunden angegeben wird.

G04 U... O...

- U: Verweildauer (in Sekunden bzw. Umdrehungen) *(voreingestellt: XS=X)*
- O: Auswahl der Verweilzeit *(voreingestellt: O1)*
 - O1: ... in Sekunden
 - O2: ... in Umdrehungen (O2 wird von der PALturn-Simulation nicht beachtet)

Fräsen: G04 ist nur wichtig beim Bohren und bei exakten Ecken. Durch den Schleppwert zwischen Soll- und Istwert (vgl. Regeltechnik in deinem Fachbuch) ist immer eine leichte Verrundung festzustellen. Mit G04 *(oder dafür eigens vorgesehene Befehle → G09)* kann man dies vermeiden.

Drehen: G04 ist zu verwenden, wenn man eine Nut einsticht. Das Werkzeug muss dann mindestens so lange stehen bleiben, bis die Spindel einmal umgelaufen ist, sonst dreht man „ein Ei“.


zur nächsten Seite


^K G04; Verweilzeit
[#] IDH_G04
[§] G04-Befehl

^{K#} **G09** Genauhalt

... sorgt in Verbindung mit den Verfahrbefehle G01-G03 dafür, dass die Vorschubgeschwindigkeit auf Null reduziert wird.

G09 (ohne zusätzliche Parameter)

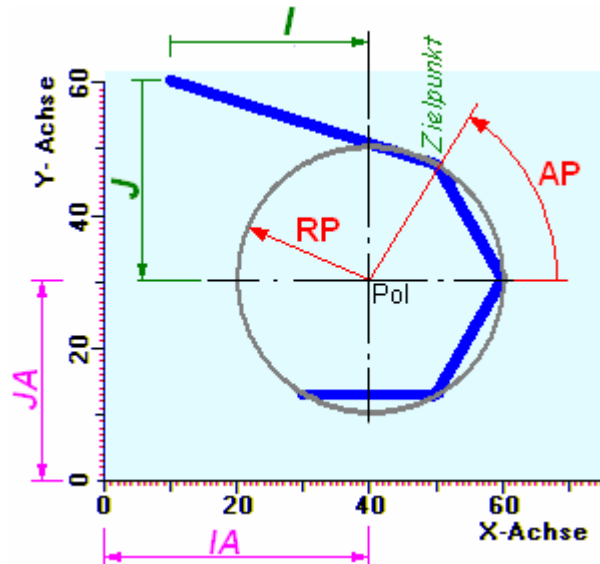
Dadurch wird ein „Verrunden“ von Kanten, das durch den sog. Schleppfehler der Steuerung bedingt ist, vermieden.


zur nächsten Seite

^{K#} Der G10- und G11-Befehl (Linearinterpolation mit Polarkoordinaten)

Bei Bohrbildern oder Bei Vielecken geht die Bemaßung oft von einem zentralen Punkt aus. Damit man nicht mit Pythagoras oder Winkelfunktionen rechnen muss, kann man die Zielpunkte auch mit Hilfe von polaren Koordinaten eingeben:

G11 I.... J.... RP... AP...



G11 I30 J30 RP20 AP60

weiter mit:

G11 IA40 JA30 RP20 AP0

G11 I-20 J0 RP20 AP-60

G11 IA40 JA30 RP20 AP-120

Will man das Gleiche im Eilgang tun, so verwendet man G10 statt G11.

Hinweis: Genau wie der G01-Befehl kann auch der G11-Befehl mit **RN (Verrundung)** erweitert werden.

Tipp: Die Fehlermeldungen im Zusammenhang mit RN beziehen sich u.U. auf die Folgezeilen. →
 Schalte bei Interpretationsproblemen vorübergehend den RN-Befehl mit // aus.

→
zur nächsten Seite

^K G10; G11; Polarkoordinaten; AP; RP

[#] IDH_G11

^S G11-Befehl (Vorschub über Polarkoordinaten)

K#\$ **G12 G13** Kreisinterpolation über Polarkoordinaten

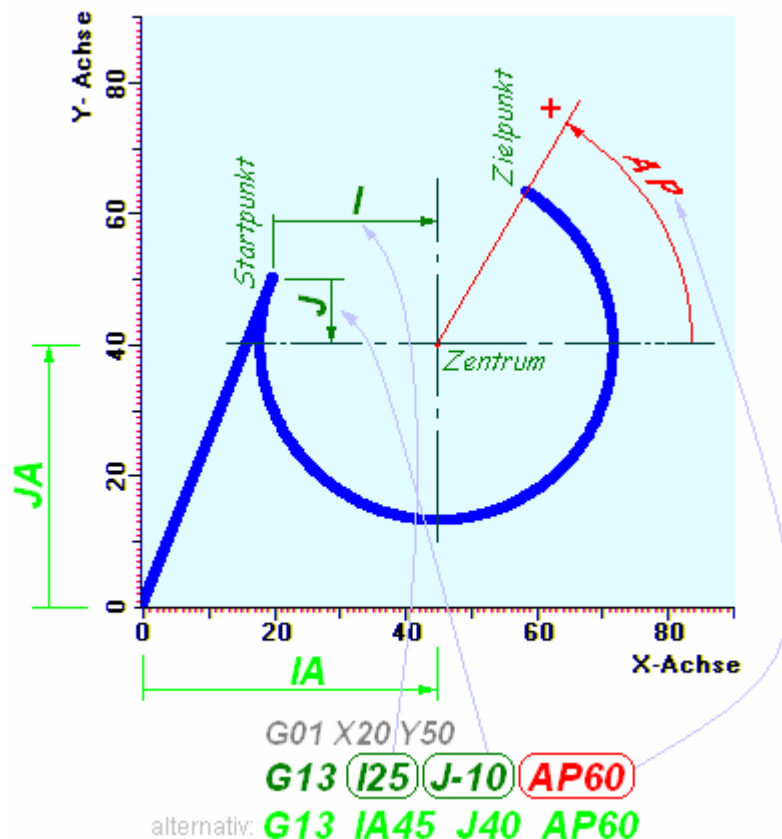
Ist das Zentrum des Bogens bekannt, so kann man statt mit Hilfe von G02 bzw. G03 den Bogen nur mittels „Mittelpunkts“koordinaten I und J, sowie mittels Polwinkel AP angeben

Syntax: **G12 IA ... JA... AP...** Bogen im Urzeigersinn
G13 IA ... JA... AP... Bogen entgegen dem Urzeigersinn

I und J sind immer inkremental (egal, ob G90 oder G91 aktiv ist).

IA und JA sind immer absolut.

Der Polwinkel AP bezieht sich (im math. Sinne) auf die pos. Abszisse. Er kann auch neg. eingetragen werden (z.B. -60 Grad statt 300 Grad)



Hinweis: Genau wie der G01-Befehl können auch die G12- bzw. G13-Befehle mit **RN (Verrundung)** erweitert werden.

Tipp: Die Fehlermeldungen im Zusammenhang mit RN beziehen sich u.U. auf die Folgezeilen. →
Schalte bei Interpretationsproblemen vorübergehend den RN-Befehl mit // aus.

➡
zur nächsten Seite

^K G12; G13; Polarkoordinaten; AP

[#] IDH_G12

^S G12-Befehl

K#\$ **G14** Werkzeugwechsellpunkt anfahren

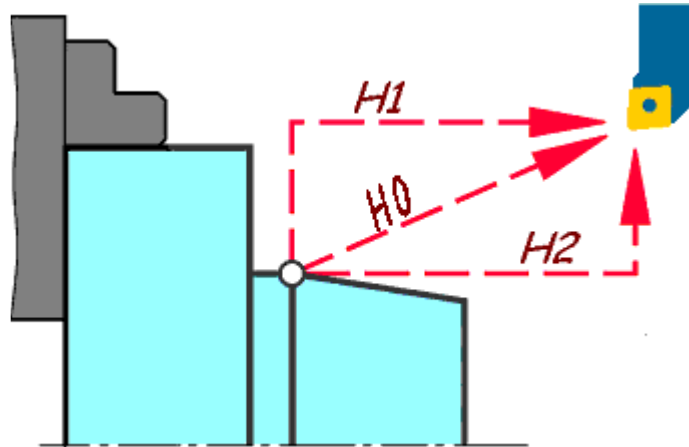
Damit wird der (an der Maschinen festgelegte) Werkzeugwechsellpunkt im Eilgang angefahren. Die Art und Weise des Anfahrens geschieht mit Hilfe des optionalen Parameters H.

Syntax: **G14 H...**


H0: in allen Achsen gleichzeitig (voreingestellt H0)

H1: zuerst in X-Achse, dann in Z-Achse

H2: zuerst in Z-Achse, dann in X-Achse



(Anmerkung: Der Werkzeugwechsellpunkt ist bei PALturn fest vorgegeben.)


zur nächsten Seite

^K G14; Werkzeugwechsellpunkt anfahren

[#] IDH_G14

[§] G14-Befehl

^{K#§}Der **G22**-Befehl **Unterprogrammaufruf**

... sorgt dafür, dass ein Unterprogramm, das sich in einer eigenen Datei befindet, (evtl. mehrmals) abgearbeitet wird.

Syntax: **G22 L ... H ...**

L: ist die Nummer des Unterprogramms

H: (optional) ist die Anzahl der Wiederholungen

Z.B. **G22 L666 H3** bedeutet: Programm mit der Nr. 666 wird 4x durchlaufen

Besonderheit bei PALturn: die NC-Datei muss sich auf dem Desktop befinden. Ist sie nicht ins PALturnFenster geladen, so erfolgt während des Programmablaufs eine entsprechende Aufforderung, das Programm zu öffnen. Außerdem muss diese Datei als Unterprogramm erstellt worden sein (Menü: Datei → Neu → Unterprogramm → Unterprogr.-Nr. eingeben z.B. 666 und später unter beliebigem Namen abspeichern)

Tipp: ist das Unterprogramm schon als externe Datei vorhanden, so erzeuge wie oben beschrieben ein neues Unterprogramm und kopiere den Dateiinhalt hinein.



zur nächsten Seite

^K G22; Unterprogramm-Aufruf

[#] IDH_G22

[§] G22-Befehl

^{K#}\$ Der **G23**-Befehl Programmabschnittswiederholung

... sorgt dafür, dass das Programm an einer anderen Stelle im Programmtext (evtl. mehrmals) abgearbeitet wird, bevor es wieder an der Absprungstelle fortgeführt wird.

Syntax: **G23 N ... N ... H ...**

N: ist die Zeilennummer am Beginn des Programmabschnitts

N: ist die Zeilennummer am Ende des Programmabschnitts

H: (optional) ist die Anzahl der Wiederholungen

Z.B. **G23 N117 N121 H2** bedeutet: Programm durchläuft 3x die Zeilen 117 bis 121



zur nächsten Seite

^K G23; Programmabschnittswiederholung; Wiederholung

[#] IDH_G23

^{\$} G23-Befehl

^{K#}\$ Der G29-Befehl **bedingter Programmsprung** & Parameter-Programmierung

... sorgt dafür, dass das Programm bei bestimmten Bedingungen an einer anderen Stelle im Programmtext fortgeführt wird. (kein Rücksprung wie bei G23)

Syntax: **G29 P... O... P... N ...**

N: ist die Zeilennummer der „anzuspringenden“ Programmzeile

Vergleichsparameter:

1. P: erste Variable

O: Vergleichsoperator

O1 bedeutet: gleich

O2 bedeutet: ungleich

O3 bedeutet: größer

O4 bedeutet: kleiner

2. P: zweite Variable

Z.B. **G29 N139** (ohne Parameter: „**unbedingter Sprung**“ in die Zeile 139)

Z.B. **G29 N117 P13 O3 P29** (mit Parameter: „**bedingter Sprung**“ in die Zeile 117, wenn der Wert von P13 gleich dem Wert von P29 hat. Wenn diese Bedingung nicht erfüllt ist, wird die dem Befehl folgende Zeile abgearbeitet.)

Parameterprogrammierung

Sie erlaubt es, viele mathematisch zu beschreibende Geometrien an der Steuerung zu programmieren. Mittels Parameter und Unterprogrammtechnik können auch eigene Zyklen nach Bedarf programmiert werden (um das fehlende Angebot der Steuerungshersteller zu ergänzen).

Mit P-Parametern kann man mit einem Bearbeitungs-Programm eine ganze Teilefamilie definieren. Dazu gibt man anstelle von Zahlenwerten Platzhalter ein:

PALmill kennt nur Benutzerparameter. Sie werden mit der Adresse P und dem ganzzahligen Adresswert von 0 bis 9999 programmiert. Ein Wertzuweisung erfolgt durch ein Gleichheitszeichen:

z.B.: P456 = -45,5

Systemparameter, z.B. Nullpunkte u.ä. werden nicht berücksichtigt. Lediglich kann auf die aktuellen abs. Position der Achsen zugegriffen werden.

z.B.: P123 = X

Einer P-Adresse kann mit Gleichheitszeichen der Wert eines arithmetischen Ausdruckes zugeordnet werden.

z.B.: P145 = 6*0,46 – (P14 * SIN(26) + 0.5)

Leider können bei PALmill (vorerst ?) nicht mehrere Parameterzuweisungen in einem NC-Satz programmiert werden.

Sorry, → neue Zeile verwenden. ☹

Einer NC-Adresse kann der Wert eines Benutzerparameters zugewiesen werden, in dem der Parameter mit vorangestelltem Gleichheitszeichen an die Adressbuchstabenkombination angehängt wird.

z.B.: Y = P145

Leider ist es bei PALmill (vorerst ?) nicht möglich, einer Adresse mit Gleichheitszeichen den Wert eines arithmetischen Ausdruckes zuzuordnen..

^K G29; Programm-Sprung; Parameter; Parameter-Programmierung

IDH_G29

^S G29-Befehl und Parameter

Sorry, → in vorhergehender Zeile den arithmetischen Ausdruck einem bestimmten Parameter zuordnen. ☹

z.B.: P555 = P33 + P17 -0,6
X = P555

Ein arithmetischer Ausdruck wird aus Parametern, Zahlenwerten und Funktionswerten zusammen mit den arithmetischen Operationen +, -, *, / und Klammerungen () gebildet, wobei die üblichen Algebraregeln über die Klammerung und „Punkt-vor-Strich-Rechnung“ gültig sind. Innerhalb eines arithmetischen Ausdruckes wird ein Parameter oder eine Funktion wie eine Zahl behandelt.

Ein arithmetischer Ausdruck muss mit dem Zeichen "(" (Klammer auf) eröffnet und mit dem Zeichen ")" (Klammer zu) abgeschlossen werden. Arithmetische Teilausdrücke eines arithmetischen Ausdruckes werden ebenfalls durch runde Klammern eingeschlossen.

Natürlich muss der Benutzerparameter P vor seiner Verwendung in einem arithmetischen Ausdruck bereits im NC-Programm vorher schon definiert sein.

Rechenoperationen

Eine **Addition** wird mit dem Zeichen "+" (Plus) programmiert.

Eine **Subtraktion** wird mit dem Zeichen "-" (Minus) programmiert.

Eine **Multiplikation** wird mit dem Zeichen "*" (Stern) programmiert.

Eine **Division** wird mit dem Zeichen "/" (Schrägstrich) programmiert.

Nicht bei PAL, aber bei PALmill: Eine **Potenz** wird mit dem Zeichen "^" programmiert.

Für die Reihenfolge der Ausführungen gilt die Punkt-vor-Strich-Regel (* / vor + -), die festlegt, dass zuerst die Multiplikation und Division vor Addition oder Subtraktion durchgeführt werden.

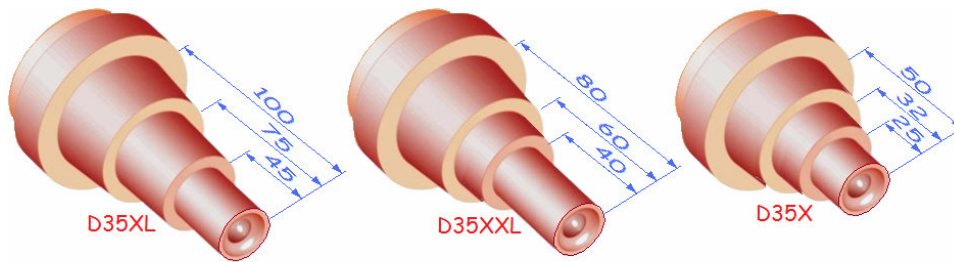
Funktionen

Eine Funktion ordnet ihrem in Klammern stehenden Argument, das auch ein arithmetischer Ausdruck sein kann, einen Funktionswert zu. Eine Funktion kann auch zwei durch Komma getrennte Argumente haben (vgl. MOD u. ATAN2). Der Funktionswert wird mit dem Funktionsnamen und den angehängten Argumenten in Klammern programmiert.

z.B.: P456 = -SIN(360/7)

ABS(F)	Berechnet den Absolutbetrag des Argumentes als Funktionswert.
INT(F)	Schneidet die Nachkommastellen des Argumentes ab und hat als Funktionswert eine ganze Zahl.
MOD(I1,I2)	Restbetrag IR einer ganzzahligen Division. Der Rest $IR = I1 - (I1 / I2) * I2$ ist betragsmäßig kleiner als I2.
SIN(F)	Sinusfunktion mit der Argumenteinheit Winkelgrad.
COS(F)	Kosinusfunktion mit der Argumenteinheit Winkelgrad.
TAN(F)	Tangensfunktion mit der Argumenteinheit Winkelgrad.
ASIN(F)	Arcussinusfunktion mit dem Funktionswert Winkelgrad.
ACOS(F)	Arcuskosinusfunktion mit dem Funktionswert Winkelgrad.
ATAN(F)	Arcustangensfunktion mit dem Funktionswert Winkelgrad.
ATAN2(F1,F2)	Die Arcustangensfunktion mit dem Funktionswert Winkelgrad. Die Funktion entspricht der Tangensfunktion $A1/A2$ für $A2 \neq 0$. und $ATAN2(F1,0)=90^\circ$ für $F1 > 0$ $ATAN2(F1,0)=-90^\circ$ für $F1 < 0$. Quadratwurzelfunktion (eng. „square root“)
SQRT(F)	Die Exponentialfunktion ist die Potenz e^F und wird mit der Eulerschen Konstanten $e = 2,71828$ berechnet.
EXP(F)	
LN(F)	Als Umkehrfunktion zur obigen Exponentialfunktion kann mit "LN" der Logarithmus zur Basiszahl e berechnet werden.
^	Potenz {nicht bei PAL erlaubt!! Alternative: $EXP(LN(\text{Basis} * \text{Hochzahl}))$ }

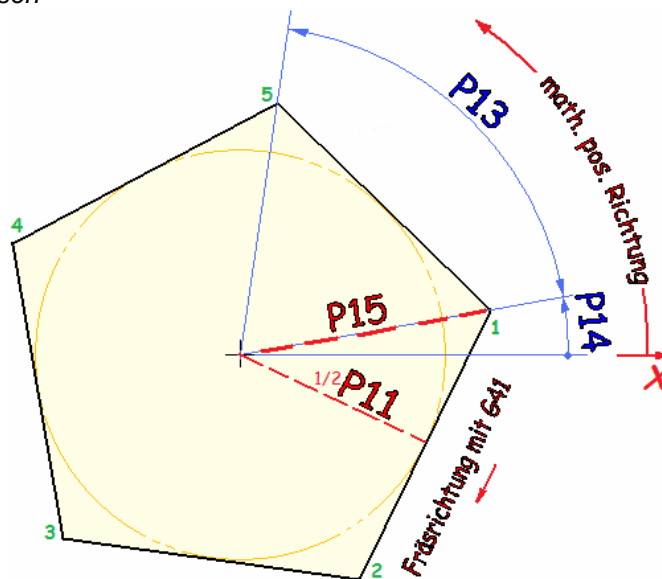
1. Beispiel: *Teilefamilie*



	P21	P22	P23
D35XL	45	75	100
D35XXL	40	60	80
D35X	25	32	50

...
 N004 P22 = -60
 N005 P23 = -80
 ...
 N040 G01 Z=P22
 N041 G01 X55
 N042 G01 Z = P23
 ...

2. Beispiel: Vieleck fräsen



P11=19 [Schlüsselweite = Durchmesser des Innenkreises]
 P12=5 [Anzahl der Flaechen]
 P13=(360/ P12) [Winkelteilung (360 : Eckenzahl)]
 P14=10 [Startwinkel der 1. Ecke (in pos. math. Richtung) wird später jedes Mal um P13 größer]
 P15=((P11/2)/C OS(P13/2)) [Radius des Außenkreises]
 P16=20 [Fräserdurchmesser; wichtig für Anfahrpunkt vor G41]
 P17=-10 [Frästiefe]

G00 Z100 F100 S1 T3 M03

P112=((P15+ P16)*C OS(P14)) [X-Anfahr-Koordinate vor der 1. Ecke]
 P113=((P15+ P16)* SIN(P14)) [Y-Anfahr-Koordinate vor der 1. Ecke]

G00 X= P112 Y= P113

G00 Z= P17

P20= P16/2;legt R des Anfahrbogens bei G47 fest

P112=(P15*C OS(P14)) [X-Koordinate der jew. Ecke]
P113=(P15* SIN(P14)) [Y-Koordinate der jew. Ecke]
G41 G47 X= P112 Y= P113 R= P20

; jetzt wird "P12"mal wiederholt

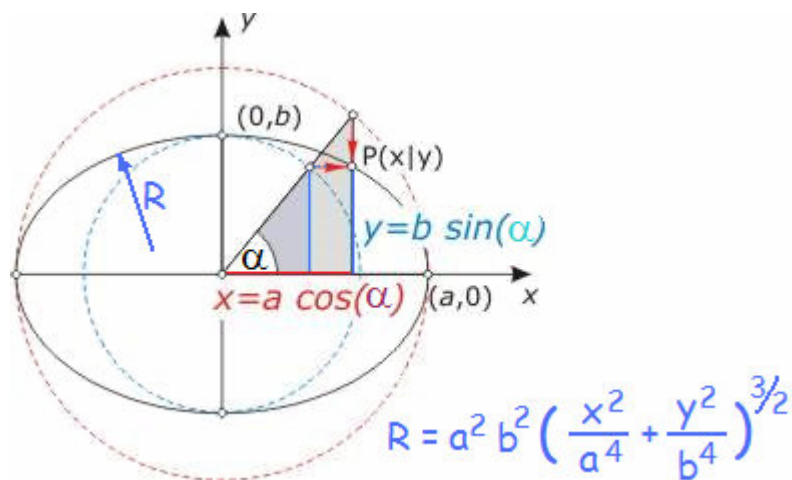
N25 P14= P14- P13 [Winkel der jew. Ecke wird erhöht
(bez. auf Mitte, entgegen dem math. Drehsinn)]

P112=(P15*C OS(P14)) [X-Koordinate der jew. Ecke]
P113=(P15* SIN(P14)) [Y-Koordinate der jew. Ecke]
G01 X= P112 Y= P113 [zur berechneten Ecke fahren]
P12= P12-1

G29 (P12 O3 (0)) N25 [Springt zurück in N25, solange Eckenzahl P12 > 0 ist.

G40 G48 R= P20
M30

3. Beispiel: Ellipse fräsen



;Eingabe:

P11 = 60 [Außendurchmesser über lange Achse]
P12 = 40 [Außendurchmesser über kurze Achse]
P13 = -7 [Tiefe, absolut]
P14 = 20 [Fräserdurchmesser; wichtig für Anfahrpunkt vor G41]
P15 = 20 [Anzahl der Bogenschritte pro Konturumrundung (min. 20)]

;daraus berechnet:

P21 = P11/2 [a]
P22 = P12/2 [b]
P23 = P21*P21*P22*P22 [a*b zum Quadrat]
P25 = 360/P15 [Winkelzunahme in Grad]
P26 = P21*P21*P21*P21 [a hoch 4 bzw. P21^4]
P27 = P22*P22*P22*P22 [b hoch 4 bzw. P22^4]

G00 z100 F100 S1 T3 M03

P34 = P14*0.75 ;legt R des Anfahrbogens bei G47 fest

P31 = P21 + P34 [X-Anfahr-Koordinate auf X-Achse]

P32 = P34/3 [Y-Anfahr-Koordinate]

G00 X=P31 Y=P32

G00 Z=P13

P31 = P21 [X-Koordinate des 1. Konturpunktes]

P32 = 0 [Y-Koordinate der jew. Ecke]

G41 G47 X=P31 Y=P32 R=P34

P124 = 0 [Lauf-Winkel Alpha setzen; muss 0 Grad sein! vgl. Anfahren!]

; Jetzt wird P15 mal wiederholt, bis Kontur 1x umrundet ist.

N10 P124 = P124 - P25/2 [Winkel Alpha wird um halben Schritt "erhöht" (bezogen auf Mitte, entgegen dem math. Drehsinn)-->G41]


P31 = P21*COS(P124) [X-Koordinate aus a und Alpha ermittelt. Wird nicht angefahren, sondern nur zur R-Ermittlung benötigt
P32 = P22*SIN(P124) [Y-Koordinate aus b und Alpha ermittelt " " Liegt auf Mitte Bogen
P33 = P23*(P31*P31/P26 + P32*P32/P27)^1.5 ; ^ gibt's bei PAL nicht: erlaubt wäre => EXP(LN("Klammerwert" * 1,5))

N10 P124 = P124 - P25/2 [Winkel Alpha wird weiter "erhöht"
P31 = P21*COS(P124) [X-Koordinate aus a und Alpha ermittelt.
P32 = P22*SIN(P124) [Y-Koordinate aus b und Alpha ermittelt

G02 X=P31 Y=P32 R=P33[Bogenschnitt fahren m.E. schneller, sauberer, als tausend kleine G01-Wege zu fahren

P15 = P15-1 [Zähler herunterzählen
G29 (P15 O3 (0)) **N10** [Springt zurück in N10, solange Bogenzahl P15 > 0 ist.

P32 = P32 -1
g01 Y=P32 [End-Tangente ist sonst evtl. ungenau bei kleinen Schrittzahlen von P15
g40 g48 r = P34
m30


zur nächsten Seite

^{K#}\$ Der **G30**-Befehl **Werkstück umspannen**

... sorgt dafür, dass das Programm bei bestimmten Bedingungen an einer anderen Stelle im Programmtext fortgeführt wird. (kein Rücksprung wie bei G23)

Syntax: **G30**

Q.. DE.... u.v.a.m..

Ist in PALturn bisher nur ohne Parameter aufrufbar.

PAL-CNC sieht weitere Optionen vor. z.B. Q2: Gegenspindel positionieren und Spannen



zur nächsten Seite

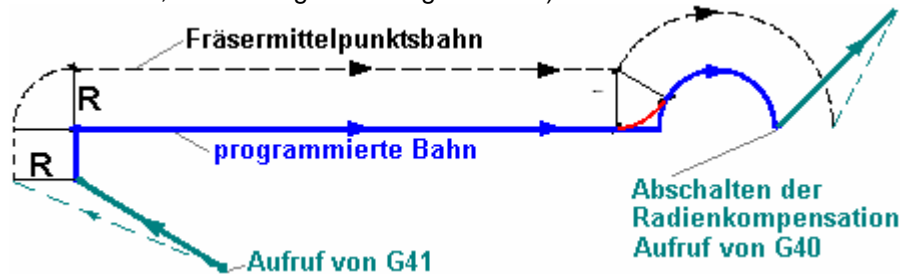
^K G30; Unterprogramm-Aufruf

[#] IDH_G30

^{\$} G30-Befehl

^{K#S} Der G41- bzw. G42-Befehl (Werkzeugbahnkorrektur)

Der Übergang von den NC-Maschinen der 70er-Jahre zu den computergesteuerten Maschinen ermöglichte es, auf die umständliche Berechnung (der Äquidistanten) der Wkz.-mittelpunktsbahn zu verzichten. (Früher musste das Programm umgeschrieben werden, wenn man einen anderen Werkzeugradius verwandte, als im Programm vorgesehen!!!)



Man braucht also nur die Kontur des Werkstücks zu programmieren, der Computer bestimmt aus dem modal (also solange geltend, bis er wieder mit G40 abgeschaltet wird) wirkenden Befehl

G41: Werkzeugradienkompensation **links** von der Werkstückkontur

G42: Werkzeugradienkompensation **rechts** von der Werkstückkontur

die zu verfahrenende Mittelpunktsbahn.

Selbstverständlich benötigt die Steuerung die erforderlichen Maße des Werkzeuges. Sie erhält sie beim Aufruf (Werkzeugwechsel) des Werkzeugs mit dem Befehlswort T. (Bei einigen wenigen Steuerungen werden mit dem Befehlswort D solche Werte aus einer Tabelle gelesen, die der Bediener bei jedem Nachschliff des Werkzeuges eingeben muss.)

Je einfacher heutzutage das Programmieren geworden ist, desto sorgloser verwenden viele Schüler die G41/G42-Funktionen: Sie fahren oft bei eingeschalteter Wkz-Radienkompensation „wild in der Gegend herum“ oder machen sich keine Gedanken zum An- oder Wegfahren zu oder von der Kontur!! Es ist ja logisch, dass beim Zurückfahren der Fräser plötzlich auf der anderen Seite steht.

**Tipps: Beim Verfahren mit G00 nie die Wkz-Radienkompensation eingeschaltet lassen!
Beim Schlichten der Drehteilkontur niemals G42 vergessen!**

G40 schaltet die Werkzeugradienkompensation wieder ab

Das unmittelbare **Anfahren im Bogen** kann PALturn leider nicht, ohne dass vorher ein Linearschritt mit G01 oder auch G00 erfolgt. (Der Programmierungsaufwand war mir zu hoch, denn der Anfahrbogen wäre eh nicht identisch mit dem programmierten Bogen.)



= > Nach G41 immer zuerst einmal (ein wenig) linear verfahren!!!!

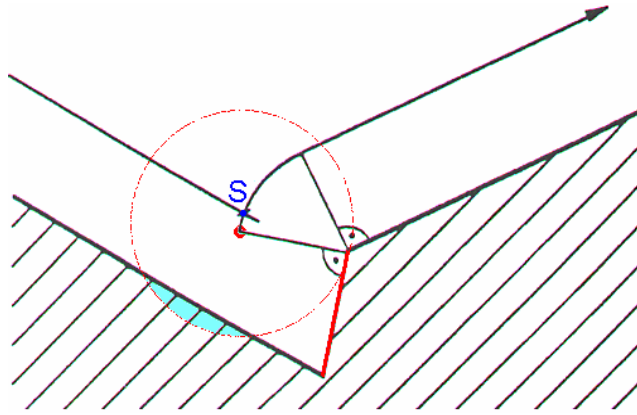
Abschließend sei auf einen typischen Programmierfehler hingewiesen, den meines Wissens auch keine andere CNC-Steuerung auffängt:

Bei der vorausschauenden Berechnung der korrigierten Werkzeugmittelpunktsbahn wird nur der Übergangswinkel zum Konturelement des anschließenden Bewegungssatzes berücksichtigt, nicht aber die Länge der Konturelemente!!!!

^K G41; Bahnkorrektur; G42; G40

[#] IDH_G41

^S G41-Befehl



Die Steuerung verrechnet korrekt das rote Konturelement: der Werkzeugmittelpunkt fährt also zuerst an den blauen Punkt **S**, aber beim nächsten Schritt fährt er zum Ende der roten Kontur, also auf den roten Punkt !!! Es entsteht eine Konturverletzung.



zur nächsten Seite

K#§

G45 G47 lineares oder bogenförmiges Anfahren an die Kontur
G46 G48 lineares oder bogenförmiges Wegfahren von der Kontur

Will man tangential zum 1. Konturelement (G01, G02,...) anfahren, so bietet

- a) G45 eine Linearbewegung und
- b) G47 ein Bewegung im Viertelkreis an.

sorry: Skizzen gibt's erst in der offiziellen Version 09

Syntax: **G45 X ... Z ... D ...** z.B.: G41 G45 X10 Z10 D15

Syntax: **G47 X ... Z ... R ...** z.B.: G41 G47 X10 Z10 R15

X und Y ist der Startpunkt der Kontur, D die Gerade bzw. der Radius des Viertelkreises.
(Zwischen dem G41/G42-Befehl und dem Anfahrbefehl darf kein anderer Verfahrbefehl stehen.)

Will man vom Konturelement (G01, G02,...) tangential wegfahren, so bietet


- c) G46 eine Linearbewegung und
- d) G48 ein Bewegung im Viertelkreis an.

sorry: Skizzen gibt's erst in der offiziellen Version 09

Syntax: **G46 D ...** z.B.: G40 G46 D15

Syntax: **G48 R ...** z.B.: G40 G48 R15

D ist die Länge der Geraden bzw. der Radius des Viertelkreises.
(Zwischen dem G40-Befehl und dem Wegfahrbefehl darf kein anderer Verfahrbefehl stehen.)


zur nächsten Seite

^K G48; G45; G46; G47;

[#] IDH_G48

[§] G48-Befehl

K#§ **G50, G53, G54 – G57 , G58 und G59** (Verschiebung)

G50: inkrementale Verschiebung AUS


G53: inkrementale und absolute Verschiebung AUS

G54 bis G57: absolute Verschiebung EIN

G58: programmierbare, inkrementale Verschiebung (mittels Polarkoordinaten) EIN

G59: „ „ „ („ karthesischer Koordinaten) EIN

Die Koordinatenwerte in einem Programm beziehen sich nicht immer auf den Werkstücknullpunkt (vgl. Bezugspunkte). Oft ist es hilfreich diesen Nullpunkt vorübergehend zu verschieben (und zu verdrehen). Dazu dienen die oben genannten Befehle.

Die Koordinaten der **absoluten Nullpunktverschiebungen** werden in die Maschine eingegeben. (Bei PALturn werden sie (X,Y,Z und Verdrehwinkel) mittels Button  in eine Tabelle eingetragen.) Im Programm können sie jew. mit den „nackten“ Befehlen G54, G55, G56 oder G57 eingeschaltet werden. Sie gelten immer absolut (auch wenn sie nach G91 aufgerufen werden). Mit G53 werden sie ausgeschaltet. Auch wird eine absolute Verschiebung dadurch ausgeschaltet, wenn eine andere eingeschaltet wird.

Für den Programmier sind die inkrementalen Nullpunktverschiebungen flexibler zu handhaben, da sie (ständig) neu verändert werden können. Zudem wirken sie additiv; d.h. wenn eine absolute Verschiebung (z.B. G55) schon eingeschaltet ist und G58 oder G59 aufgerufen wird, so addieren sich die Verschiebewerte (bei G90 nur einmal, bei G91 immer weiter). Mit G50 werden dann die inkrementalen Verschiebungen ausgeschaltet; die absolute bleibt erhalten. Mit G53 werden alle Einstellungen abgeschaltet.

sorry: Skizzen gibt's erst in der offiziellen Version 09

Syntax: **G58 RP ... AP ... Z ... AR ...** z.B.: G58 RP50 AP20 AR30
Verschiebung: aus Radius RP und Winkel AP , Koordinatendrehung: Winkel AR

sorry: Skizzen gibt's erst in der offiziellen Version 09

Syntax: **G59 X ... Y ... Z ... AR ...** z.B.: G58 RP50 AP20 AR30
Verschiebung: aus Radius RP und Winkel AP , Koordinatendrehung: Winkel AR

Z ... AR ... sind nur optional

(Werden die G58-, G59-Befehle innerhalb des G91-Modus gesetzt, dann addiert sich die Verschiebung auf die vorhergehende. Ob das so in allen Steuerungen läuft, und ob die PAL-CNC-Strategen sich das so gedacht haben, weiß ich nicht. Wer mehr weiß, maile mir bitte.)


[zur nächsten Seite](#)

^K G50; G53; G54; G55; G56; G57; G58; G59; Nullpunkt-Verschiebung


[#] IDH_G53

[§] G53-Befehl

K#S G60 konstantes Aufmaß in Z und äquidistantes Aufmaß (Offset) bei der Werkzeug-Radien-Korrektur und bei Zyklen

Will man beim Fräsen einer Kontur mit G41 bzw. G42 z.B. ein Schlichtaufmaß lassen (und später noch einmal den gleichen Unter-Programmteil ohne Schlichtaufmaß erneut aufrufen), so stellt man mit G60 dieses Distanzmaß ein. ==> Der Fräser fräst eine Äquidistante zur programmierten Kontur.

Beim Drehen nicht definiert :
einige benutzen G60 nur als Verschiebung Syntax: z.B.: G60 DX 0.2 DZ 0.5
(Mit G60 R 0 D 0 wird diese Einstellung rückgängig gemacht.)


zur nächsten Seite

^K G60; Offset; Aufmaß;
[#] IDH_G60
^S G60-Befehl

\$K# **Die Befehle G90 und G91 (Absolut- und Schrittmaßprogrammierung)**

Beide Befehle wirken modal, d.h. sie gelten solange, bis sie jew. von dem anderen ausgeschaltete werden. Z.B. G91 in einem Unterprogramm gilt solange, bis irgendwann wieder G90 erscheint.

G90 (=Absolutbemaßung) besagt, dass sich die Koordinatenangaben auf das momentan verwendete Bezugssystem (vgl. Nullpunkte) beziehen.

G91 (Schrittmaßprogrammierung) besagt, dass sich die Koordinatenangaben immer („inkremental“) jew. auf den letzten Zielpunkt beziehen.



^S G90-Befehl

^K G90; Absolutbemaßung; G91; Inkrementalbemaßung


IDH_G90

^{K#}\$ **Der G92-Befehl** (Speicher setzen oder aber Spindeldrehzahlbegrenzung)

An Drehmaschinen wird damit die max. zulässige Drehzahl begrenzt, die ja z.B. beim Plandrehen unter G96 bei kleiner werdendem Radius immer größer würde! Fliehkräfte!!

G92 S... (Drehzahl in 1/min)

Bei Fräsmaschinen-Steuerungen wird der Befehl G92 dazu benutzt, den Werkstück-Nullpunkt zu setzen. Nicht so bei PAL.


zur nächsten Seite

^{K#§} **Die Befehle G94 und G95** (Vorschub und Vorschubgeschwindigkeit)

Beide Befehle wirken **modal**, d.h. sie werden jew. durch den anderen ausgeschaltet.
Also noch mal: G94 bleibt solange aktiv, bis G95 vorkommt.

Wurde **G94** programmiert, so wird E als **Vorschubgeschwindigkeit in mm/min** interpretiert.
(Drehachsen : Grad/min)

z.B.: G94 F150

Hier bewegt sich das Werkzeug mit 150 mm/min. (Bewegen sich mehrere Achsen gleichzeitig, so werden laut Pythagoras die Geschwindigkeiten aufgeteilt, so dass der Vorschub an einer Raumkurve immer konstant bleibt.)

An Fräsmaschinen ist (meist) G94 (mittels Maschinenparameter = Konstanten, die man auch verändern kann) voreingestellt. An Drehmaschinen hingegen ist meist G95 voreingestellt !!

Wurde **G95** programmiert, so wird E als **Vorschubgeschwindigkeit in mm/Umdrehung** interpretiert.

G94 F0,5

Hier bewegt sich das Werkzeug so, dass bei jeder Umdrehung 0,5 mm zurückgelegt werden.
Macht mehr Sinn an Drehmaschinen.



zur nächsten Seite

^K G94; G95

[#] IDH_G94

[§] G94-Befehl


^{K#§}Die Befehle G96 und G97 (Konst. Schnittgeschwindigkeit)

Beide Befehle wirken **modal**, d.h. sie werden jew. durch den anderen ausgeschaltet.
Also noch mal: G96 bleibt solange aktiv, bis G97 vorkommt.

Bei **Drehmaschinen** bedeutet **G96**, dass die S-Werte als **Schnittgeschwindigkeit in mm/min** angesehen werden; die **G97**-Einstellung hingegen bedeutet, dass die S-Werte als **Drehzahlen** in 1/min angesehen werden.

Beim Bohren ($x=0$, dann würde die Drehzahl unendlich groß), **aber auch beim Gewindezyklus G83 muss vorher G97 eingeschaltet sein!!!**

Bei **Fräsmaschinensteuerungen** heißt G96, dass die F-Werte (Vorschub) sich nicht wie bei G97 auf die Fräsermittelpunktsbahn beziehen, sondern auf die Werkstückkontur. Dies bewirkt eine gleichmäßige Oberfläche an Wölbungen.

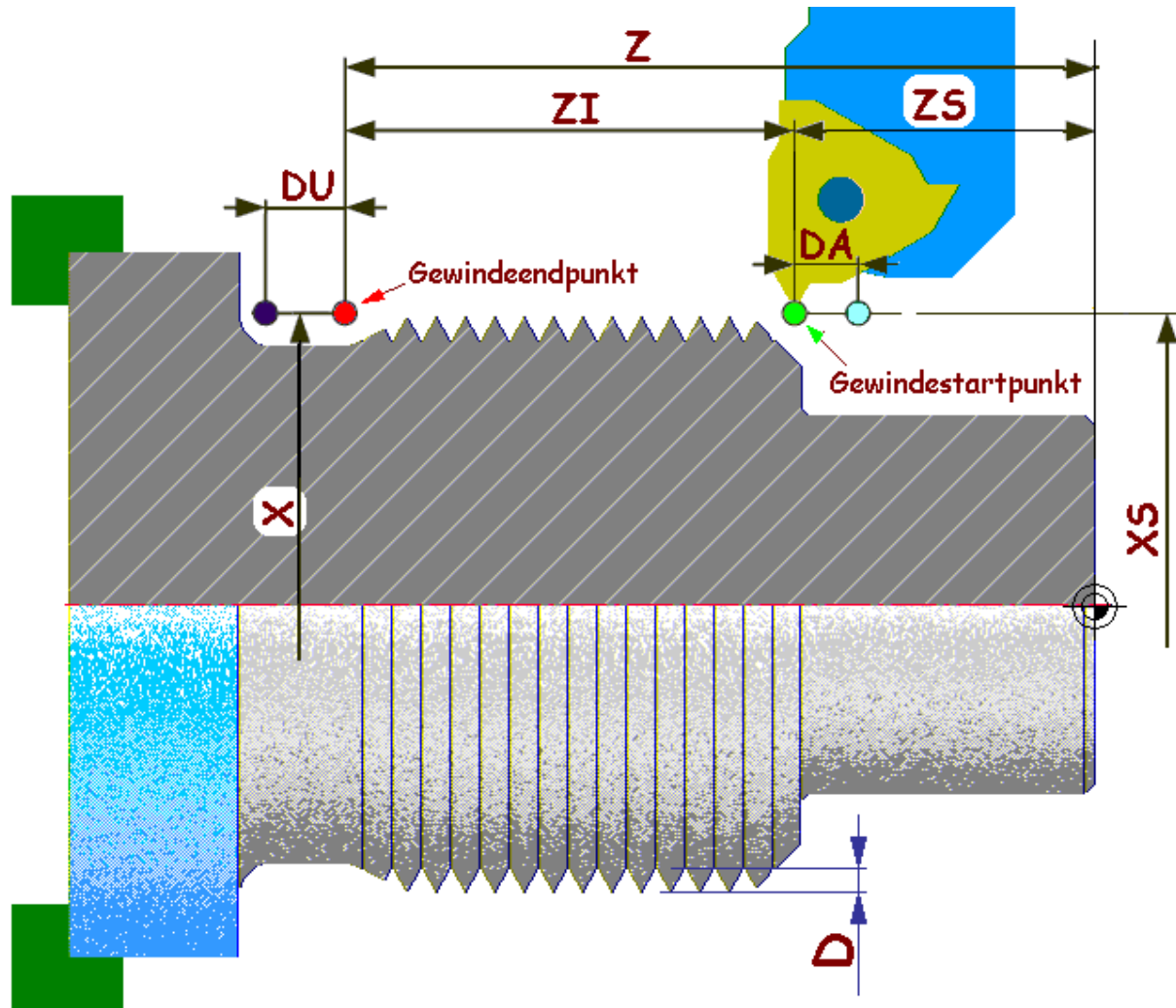

zur nächsten Seite

^K G96; G97
[#] IDH_G96
[§] G96-Befehl

K# \$ G31 – Gewindezyklus für Außen- und Innenbearbeitung

Er fertigt (mit Meißel T5 bzw. T11) ein (evtl. auch konisches) Gewinde in mehreren Schritten.

G31 X... (XI) Z... (ZI) F... D..... XS... ZS... DA... DU... Q... O... AE... H... S... M...



G31 XA24 ZA-50 F3 D1.84 ZS-20 XS24 Q12 O13

- X: Gewindeendpunkt
- XA: bezogen auf Werkstücknullpunkt (absolut)
- XI: bezogen auf den Gewindestartpunkt
- Z: Gewindeendpunkt
- ZA: bezogen auf Werkstücknullpunkt (absolut)
- ZI: bezogen auf den Gewindestartpunkt
- F: Steigung (in Z-Achse)
- D: Gewindetiefe
- XS: Gewindestartpunkt in X-Richtung absolut (voreingestellt: XS=X)
- ZS: Gewindestartpunkt (in Z-Richtung) absolut (voreingestellt: ZS=Z)
- DA: Gewindeanfahrstrecke (in Richtung Z) (voreingestellt auf DA 0)

^K G31; Gewindezyklus

IDH_G31

^S G31-Zyklus


DU: Gewindeauslaufstrecke (in Richtung Z) *(voreingestellt auf DU 0)*
Q: Zahl der Schnitte) *(voreingestellt auf Q1)*
O: Zahl der Leerschnitte) *(voreingestellt auf O0)*
AE: Eintauchwinkel zur X-Achse für Zustellung der rechten und linken Flanke
(voreingestellt: AE 29)
H: Zustellart und Restschnittauswahl
H1: ohne Versatz Restschnitt aus
H2: linke Flanke Restschnitt aus
H3: rechte Flanke Restschnitt aus
H4: Versatz R/L wechselweise Restschnitt aus


H1: ohne Versatz Restschnitt ein
H2: linke Flanke Restschnitt ein
H3: rechte Flanke Restschnitt ein
H4: Versatz R/L wechselweise Restschnitt ein
Restschnitt: $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{8}$, ... * (D/Q)

S: *(voreingestellt auf aktuelle Spindeldrehzahl / Schnittgeschwindigkeit)*
M: *(voreingestellt auf aktuelle Drehrichtung / Spülmittelschaltung)*

Hinweis:

Bei der PALturn-Simulation sind die Einstellungen für die Zustellungsart „H“ nicht erkennbar.


zur nächsten Seite

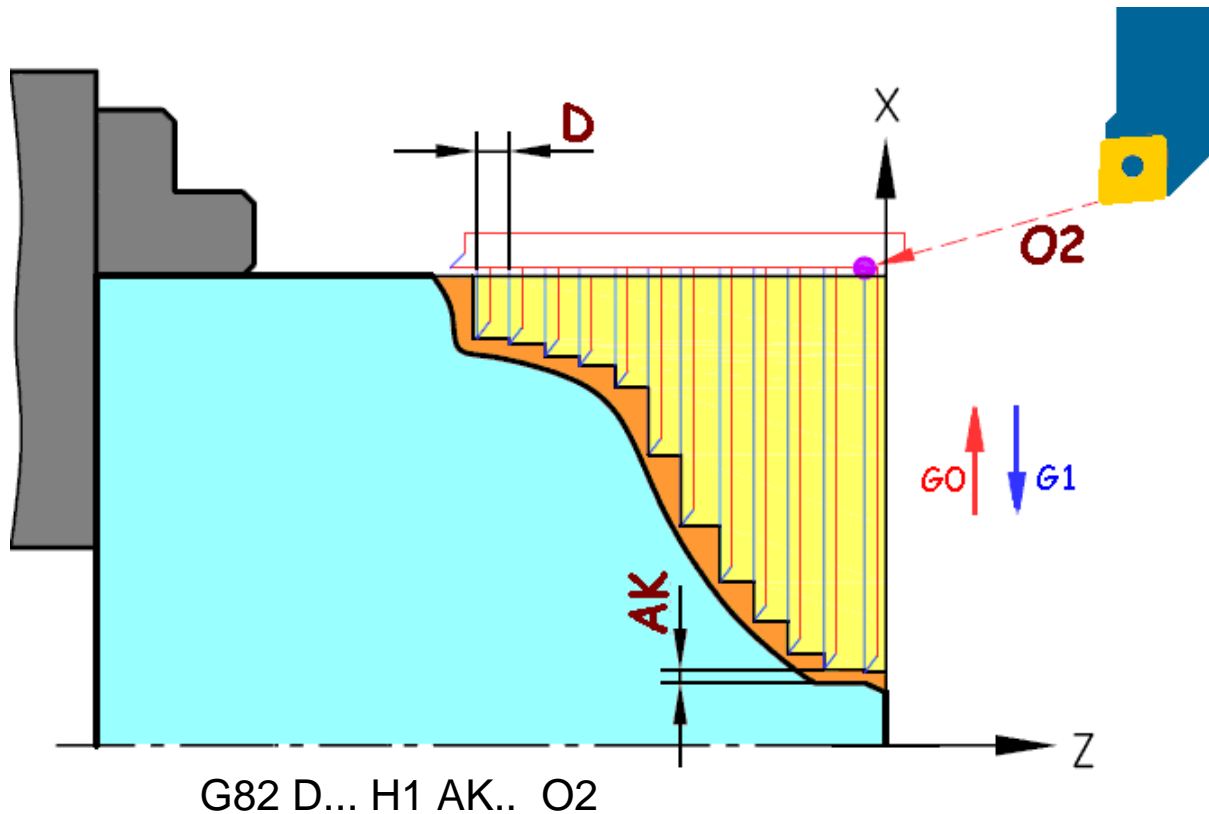

zur nächsten Seite

K# \$ G82 – Abspannzyklus Plan

= Planschruppzyklus an eine programmierte Kontur

G82 D... H... AK... AZ... AX... AE... AS... AV... O... Q... V...

Es folgt dann die Beschreibung der Kontur, deren Ende mit dem Befehl **G80** gekennzeichnet ist.



- D:** Zustellung (Pflichtparameter)
H: Bearbeitungsart
H1 nur Schruppen, 1x45 Grad abheben
H2 stufenweises Auswinkeln entlang der Kontur
H3 wie H2 mit zusätzlichem Konturschnitt am Ende
H24 Schruppen mit H2 und anschließendes Schlichten
AK: konturparalleles Aufmaß auf die zu bearbeitende Kontur (voreingestellt auf AK 0)
AZ: Aufmaß in Richtung Z (voreingestellt auf AZ 0)
AX: Aufmaß in Richtung X (voreingestellt auf AX 0)
AE: Eintauchwinkel (voreingestellt: aus dem Wkz-Konturspeicher) (noch) nicht bei PALturn
AS: Austauschwinkel (voreingestellt: aus dem Wkz-Konturspeicher) (noch) nicht bei PALturn
AV: Sicherheitsabschlag für AE und AS (voreingestellt auf AV 1)
O: Def. Bearbeitungsstartpunkt (voreingestellt auf O 1)
O1 = aktuelle Werkzeugposition
O2 = aus der Kontur berechnet
Q: Leerschnittoptimierung (voreingestellt auf Q 1)
Q1 = Optimierung AUS
Q1 = Optimierung EIN (noch) nicht bei PALturn
V: Sicherheitsabstand (voreingestellt auf V 1)

^K G82; Abspannzyklus plan; Planschruppzyklus

IDH_G82

^S G82-Zyklus

- F: *(voreingestellt auf aktueller Vorschub)*
- S: *(voreingestellt auf aktuelle Spindeldrehzahl / Schnittgeschwindigkeit)*
- M: *(voreingestellt auf aktuelle Drehrichtung / Spülmittelschaltung)*

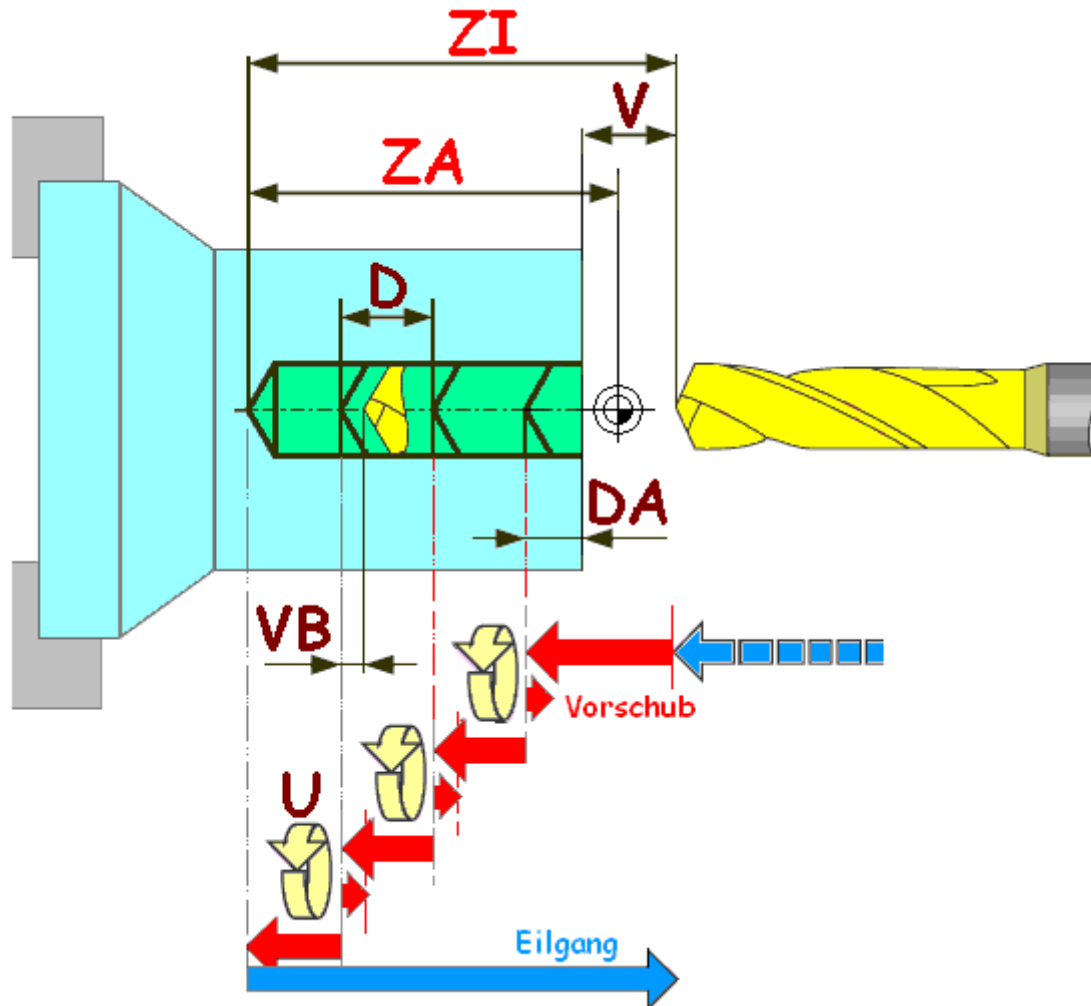


zur nächsten Seite

K#\$ G84 – Bohrzyklus

= (Tief-) Bohren mit Spanbruch und/oder Spanentleerung

G84 Z... D... V... VB... DR... DM... R... DA... U... O... FR... E... F... S... M...



G84 ZI-100 D30 V4 VB1 DR2 U0.5

Pflichtparameter:

Z: Tiefe der Bohrung

ZI: inkremental zur aktuellen Werkzeugposition

ZA: absolut

optional:

D: Zustelltiefe (voreingestellt auf Endbohrtiefe)

V: Sicherheitsabstand (voreingestellt auf V 1)

VB: Sicherheitsabstand vor Bohrgrund (voreingestellt auf VB 1)

DR: Reduzierwert der Eintauchtiefe (voreingestellt: DR 0)

DM: Mindestzustellung (ohne Vorzeichen) (voreingestellt: DM = DR {Falls DR=0, dann DM=D})

R: Rückzugsabstand (voreingestellt: auf Startpunkt)

DA: Anbohrtiefe (voreingestellt auf DA 0)

U: Verweilzeit am Bohrgrund (voreingestellt auf U 0)

^K G84; Bohrzyklus

[#] IDH_G84

^S G84-Zyklus

- O: Auswahl der Verweilzeiteinheit (voreingestellt: 0 1)
1: in Sekunden
2: in Umdrehung
- FR: Eilgangreduzierung in % (voreingestellt auf FR 100, also: nicht reduziert)
- E: Anbohrvorschub (voreingestellt: $E = F$)
- F: (voreingestellt auf aktueller Vorschub)
- S: (voreingestellt auf aktuelle Spindeldrehzahl / Schnittgeschwindigkeit)
- M: (voreingestellt auf aktuelle Drehrichtung / Spülmittelschaltung)

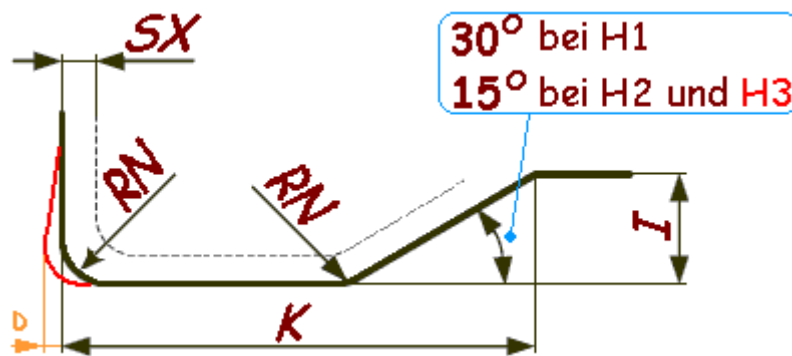
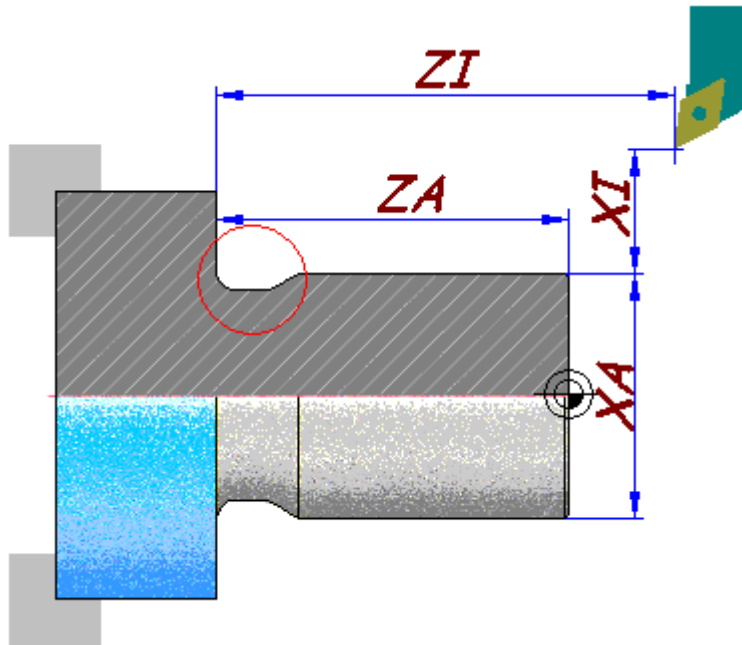


zur nächsten Seite

K#§ G85 – Freistichzyklus

zur Drehen von (Außen- oder Innen-) Freistichen nach DIN 76 und DIN 509 E/F

G85 X... Z... I... K... RN... SX... H... D... E... F... S... M...



Tabellenbuch benutzen!

z.B.: G85 XA30 ZA-50 I1,6 K5.2 RN1 SX0.3 H1 E0.15

Das aktive Werkzeug bestimmt, ob jeweils ein Außen- oder Innenfreistich gedreht wird.

Pflichtparameter:

- X, XA, XI: Freistichposition in X-Richtung
- Z, ZA, ZI: Freistichposition in Z-Richtung
- I: Freistichtiefe
- K: Freistichbreite

H: Bearbeitungsart

H1 Gewindefreistich DIN 76 (voreingestellt auf H1)

^K G85; Freistichzyklus

IDH_G85

§ G85-Zyklus

H2 Freistich DIN 509 Form E
H3 Freistich DIN 509 Form F

RN: Eckenradius (voreingestellt auf RN 0.3)
D: Tiefe in der Stirnseite (Maß t_2 bei DIN 509 F) (voreingestellt auf RN 0.2)
SX: Bearbeitungszugabe (Schleifaufmaß) (voreingestellt auf SX 0)
E: Eintauchvorschub (voreingestellt auf $0.25 * F$)

F: (voreingestellt auf aktueller Vorschub)
S: (voreingestellt auf aktuelle Spindeldrehzahl / Schnittgeschwindigkeit)
M: (voreingestellt auf aktuelle Drehrichtung / Spülmittelschaltung)

Hinweis:

Zur korrekten Beschreibung ist neben den Pflichtparametern (rot gekennzeichnet) auch „RN“ und „D“ nötig. (Wurde wahrscheinlich bei PAL vergessen oder man erwartet eine Steuerung, die alle Tabellenwerte von DIN 76 und DIN 509 gespeichert hat ???)
Die Bearbeitungszugabe SX ist bei der Simulation nicht sichtbar. (Das habe ich mir „geschenkt“; die Zehntel mm sieht man eh' nicht.)

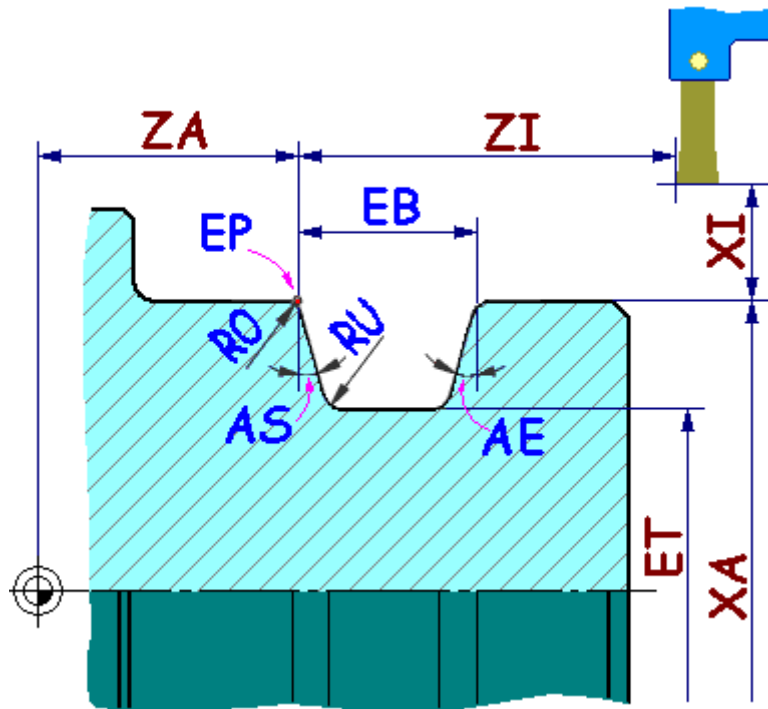


zur nächsten Seite

K# \$ G86 – Einstechzyklus radial

... zur Herstellung von Einstichen in X-Richtung

G86 Z... X... ET... EB... D... AS... AE... RO... RU... AK... AX... EP... H... DB...
V..... E... F... S... M...



G86 Z-40 X80 ET50 EB20 D4 AS10 AE10 RO-2.5 RU2 H14

Pflichtparameter:

- Z:** Einstichsetzposition P in Z-Richtung (gesteuert von G90/G91)
 - Zi: inkremental zur aktuellen Werkzeugposition
 - ZA: absolute Werkstückkoordinate
- X:** Einstichsetzposition P in X-Richtung (gesteuert von G90/G91)
 - XI: inkremental zur aktuellen Werkzeugposition
 - XA: absolut
- ET:** absoluter Durchmesser der Einstichtiefe

optional:

- EB:** Einstichbreite (Betrag) und Einstichlage (Vorzeichen) (voreingestellt auf Einstichmeißelbreite)
 - EB+ Einstich in Richtung Z+ von der programmierten Einstichposition P
 - EB- Einstich in Richtung Z- von der programmierten Einstichposition P
- D:** Zustelltiefe (voreingestellt auf $D = ET$)
- AS:** Flankenwinkel des Einstichs am Startpunkt bezogen auf die Stechrichtung (voreingestellt auf $AS = AE$)
- AE:** Flankenwinkel des Einstichs am Endpunkt bezogen auf die Stechrichtung (voreingestellt auf $AS = AS$)
 - ergo: Bei symmetrischen Nuten braucht man nur AS oder AE einzugeben. Keine Eingaben: Winkel ist Null.

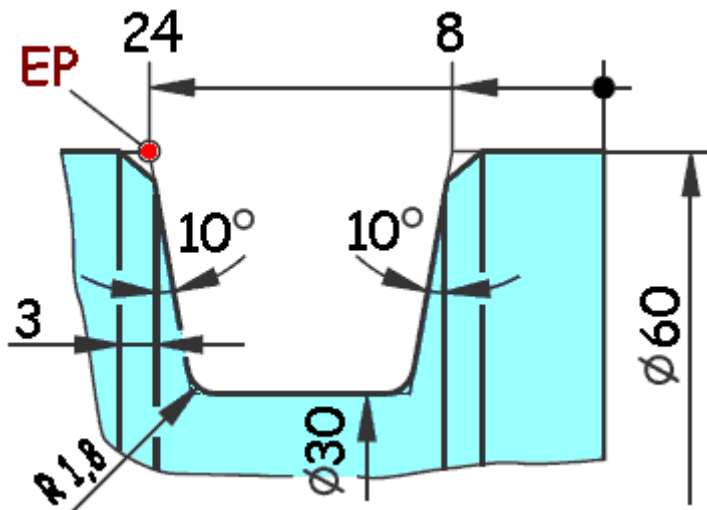
^K G86; Einstechzyklus radial

IDH_G86


^S G86-Zyklus

- RO: Verrundung oder Fase der oberen Ecken (voreingestellt auf RO 0)
 RO+ Verrundung
 RO- Fasenbreite (als Winkelhalbierende)
- RU: Verrundung oder Fase der unteren Ecken (voreingestellt auf RU 0)
 RU+ Verrundung
 RU- Fasenbreite (als Winkelhalbierende)
- AK: Konturparalleles (äquidistantes) Aufmaß auf die Kontur (voreingestellt auf AK 0)
- AX: Aufmaß in X auf Kontur (voreingestellt auf AX 0)
- EP: Setzpunktfestlegung für den Einstich (Position P) (voreingestellt auf EP 1)
 EP1: Setzpunkt in einer Ecke der Einstichöffnung
 EP2: Setzpunkt in einer Ecke des Einstichbodens
- H: Bearbeitungsart (voreingestellt auf H 14)
 H1 Vorstechen
 H14 Vorstechen und Schlichten
 H2 Stechdrehen
 H24 Stechdrehen und Schlichten
 H4 Schlichten
- DB: Zustellung in % der Meißelbreite beim Stechen (voreingestellt auf DB 75)
- F: (voreingestellt auf aktueller Vorschub)
 S: (voreingestellt auf aktuelle Spindeldrehzahl / Schnittgeschwindigkeit)
 M: (voreingestellt auf aktuelle Drehrichtung / Spülmittelschaltung)
- V: Sicherheitsabstand über der Einstichöffnung (voreingestellt auf V 1)
- E: Vollmaterial-Einstechvorschub (voreingestellt auf E = F)

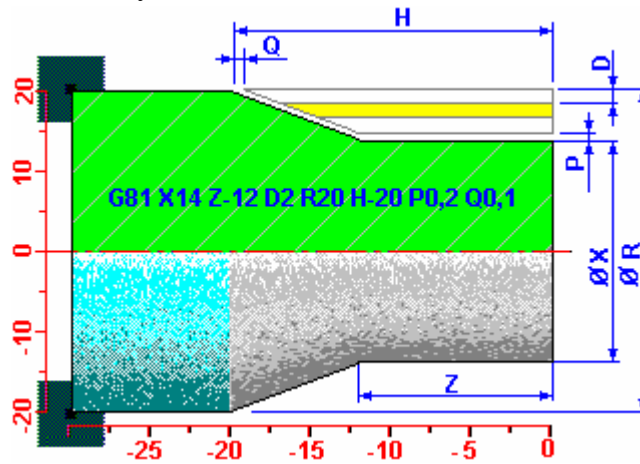
z.B.:



G86 Z-24 X60 ET30 EB16 D4 AS10 AE10 RO-2.5 RU1.8 H14


zur nächsten Seite

Veraltet! (ehem. G81-Zyklus von PAL 1992)



G981 X... Z... D... R... H... P... Q...


R: Start- und Enddurchmesser des Zyklus

P: Bearbeitungszugabe in X

Q: Bearbeitungszugabe in Z

Ist $H = Z$, dann wird ein rechteckiger Zapfen gedreht

Der gleiche Zyklus gilt auch für die **Innen**bearbeitung. Dann ist natürlich $X > R$.

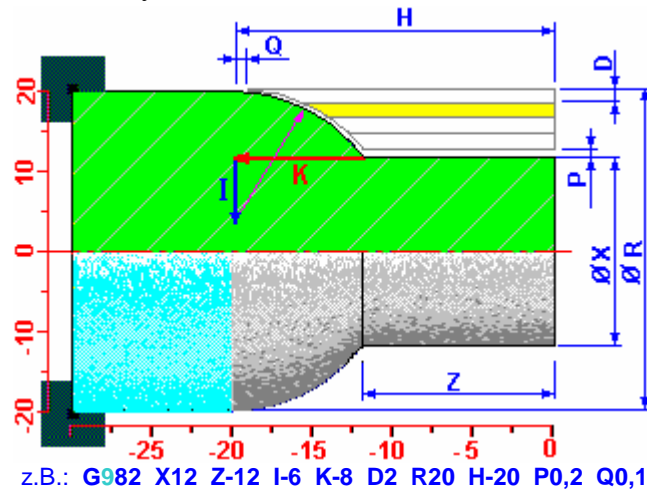

zur nächsten Seite

^K G981; Abspannzyklus längs

IDH_G981

§ G981-Zyklus

Veraltet! (ehem. G82-Zyklus von PAL 1992)



G982 X... Z... I... K... D... R... H... P... Q...

- R: Start- und Enddurchmesser des Zyklus
- I und K: inkrementale Angabe des Mittelpunktes
- P: Bearbeitungszugabe in X
- Q: Bearbeitungszugabe in Z
(auch konkave Radien möglich)

Der gleiche Zyklus gilt auch für die **Innen**bearbeitung. Dann ist natürlich $X > R$.
(R = Enddurchmesser des Bogens, aber PALturn toleriert bis zu 2mm Aufmaß.)

→
zur nächsten Seite

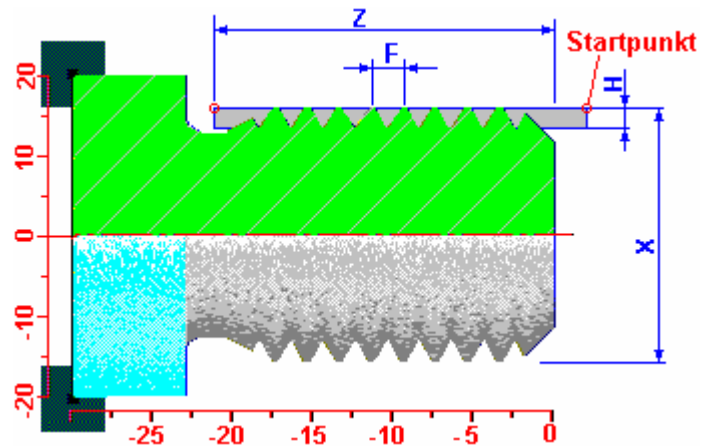
^K G982; Abspannzyklus längs mit Radius

IDH_G982

§ G982-Zyklus

K#§ G983 – Gewindezyklus längs für Außen- und Innenbearbeitung

Veraltet! (chem. G83-Zyklus von PAL 1992)



z.B.

G97 M03 G96 darf nicht eingeschaltet sein; M03 für Rechtsgewinde


G00 X16 Z4 für genügend großen Anlaufweg sorgen!

G983 X16 Z-20,5 D9 H1,23 F2

G983 X... Z... F... D... H...

- X: Außendurchmesser des Gewindes
- Z: Länge einschließlich Gewindeauslauf
- F: Gewindesteigung
- D: Anzahl der Schnitte
- H: Gewindetiefe

Der gleiche Zyklus gilt auch für die **Innenbearbeitung**

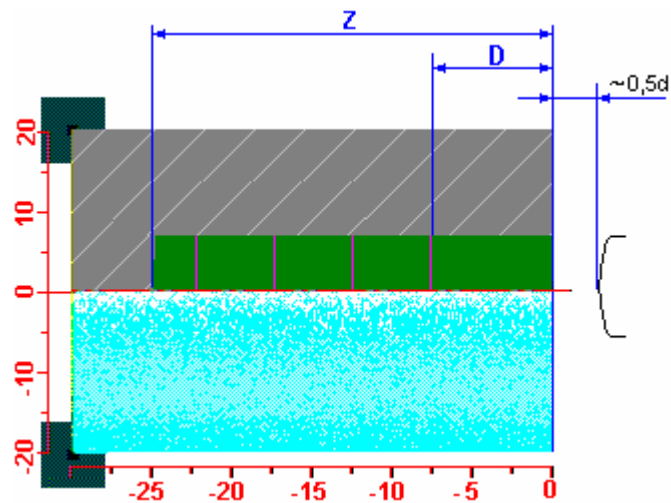

zur nächsten Seite

^K G983; Gewindezyklus

[#] IDH_G983

[§] G983-Zyklus

Veraltet! (ehem. G84-Zyklus von PAL 1992)




z.B.

G97 M03 G96 darf nicht eingeschaltet sein;
G00 X0 Z5
G84 Z-25 F0,04 D-10 H4

G984 Z... F... D... H...

Z: Gesamt-Bohrtiefe
F: Vorschub
D: erste Bohrtiefe
H: Anzahl der Spanentleerungen


[zur nächsten Seite](#)


^{K#}\$ **Die M-Befehle** (miscellaneous=vermischt; Schaltbefehle)

... steuern die SPS der CNC-Maschine. Sie schalten irgendwelche Motoren, Spindeln, Zangen etc. ein- oder aus.

M-Befehle: (Ein- oder Ausschaltbefehle für die SPS)

M03	Spindel im Uhrzeigersinn
M04	" " Gegenuhrzeigersinn (z.B. bei Schrägbett-Drehmaschinen)
M06	Werkzeugwechsel
M07 und M08	Kühlschmierung Ein
M09	" Aus
M13	Spindeldrehung rechts
M14	Spindeldrehung links
M15	Spindel und Kühlmittel aus
M17	Unterprogramm Ende
M30	Programm Ende
M99	Werkstück umspannen (nicht genormt)

In einer Befehlszeile darf ein M-Befehl nun mehrmals vorkommen!!


zur nächsten Seite

^{K#§} **Die F-Adresse** (FEED; Vorschub)

...ist erforderlich, damit die CNC-Maschine weiß, mit welcher Vorschubgeschwindigkeit sie verfahren soll.

Ein F-Befehl muss spätestens in der Programmzeile aufgetreten sein, in der G01, G02 o.ä. zum ersten Mal erscheinen.

Je nach Art der CNC-Maschine bzw. je nach ihrer Voreinstellung oder der Programmierung von G94 bzw. G95 wird der F-Wert unterschiedlich interpretiert:

Mit G94 bedeutet der F-Wert mm pro Minute	(Bei Fräsmasch. voreingestellt.)
Mit G95 bedeutet der F-Wert mm pro Umdrehung	(Bei Drehmasch. voreingestellt.)



zur nächsten Seite

^K F; Vorschub

[#] IDH_F

[§] F-Befehl

^{K#}\$ **Die S-Adresse** (SPEED; Drehzahl o. Schnittgeschwindigkeit)

...ist erforderlich, damit die CNC-Maschine weiß, mit welcher Schnittgeschwindigkeit sie das Werkstück bearbeiten soll.

Ein S-Befehl muss spätestens in der Programmzeile aufgetreten sein, in der M03 bzw. M04 (Spindelorientierung). zum ersten Mal erscheint.

Je nach Art der CNC-Maschine bzw. je nach ihrer Voreinstellung oder der Programmierung von G96 bzw. G97 wird der S-Wert unterschiedlich interpretiert:

- Mit G96 bedeutet der S-Wert konstante **Schnittgeschwindigkeit** in m pro Minute (Bei Fräsmaschinen voreingestellt.)
- Mit G97 bedeutet der S-Wert die **Spindeldrehzahl** in Umdrehungen pro Minute (Bei Drehmaschinen voreingestellt.)



zur nächsten Seite

^K S; Drehzahl

[#] IDH_S

^{\$} S-Befehl

K#§ Die T-Adresse (TOOL Werkzeug)

...ist erforderlich, damit die CNC-Maschine weiß, welches Werkzeug eingespannt ist. Wenn T in der Befehlszeile erscheint, wird, bevor sie abgearbeitet wird, der **Werkzeugwechsel** eingeleitet. Geschieht dieser nicht, wie in einem Fräszentrum automatisch, so muss laut PAL der Befehl M06 eingegeben werden. Dann hält die Spindel u.v.a.m. an und das nächste Werkzeug kann manuell gewechselt werden.

Mit dem Werkzeugwechsel alleine ist es noch nicht getan, denn die veränderten Längen und Radien des neuen Werkzeuges müssen von der CNC-Steuerung verrechnet werden. Deshalb entnimmt sie aus dem **Werkzeug(korrektur)speicher** diese Werte (+ Verschleißkorrekturwerte, Standzeiten u.ä.). (Bei einigen Steuerungen muss dieser Aufruf durch die Adresse D programmiert werden. Z.B.T04 D04 ...)

Bei ganz pfiffigen Steuerung wird D optional eingesetzt d.h. man kann zusätzliche Korrekturwerte aufrufen. Ist z.B. der eingelesene Wkz-Radius größer als der wirkliche, hat man auf einfache Art eine Schlichtzugabe erzielt. Nun kann man die gleiche Kontur z.B. mit G41 noch einmal abfahren nachdem dann die realen Korrekturwerte eingelesen wurden. Noch eleganter wird diese Möglichkeit durch Angabe von sog. Offset-Werten gelöst. (vgl. z.B. Deckel: G48)


PALturn reicht alleine der T-Aufruf zum Werkzeugwechsel. (Vergiss dennoch M06 nicht; zumindest nicht in deiner Prüfung!)

Zur Verfügung stehen 20 verschiedene Drehwerkzeuge, die bei der Voreinstellung mit **Strg + W** eingesehen (und z.T. verändert) werden können. Wenn man mit der Maus auf eines der Werkzeuge klickt, dann wird sein Name in den Zwischenspeicher abgelegt: im Editor kann er dann mit Strg+V ins Programm eingefügt werden.

(Fünf weitere „mutierte“ Werkzeuge zum Experimentieren gibt's noch:



T21 mit ganz großem Schneidenradius von 3mm,
T22 mit Schneidenradius Null und „langgezogener Nase“
T23 als 12 mm langer Einstechmeißel
T24 als runder „Längs“Einstechmeißel
T25 als kreisrunde Schneide von 1 mm, ohne Schaft)


[zur nächsten Seite](#)

^K T; Werkzeug

IDH_T

[§] T-Befehl


^{K#§} **Die L-Adresse** (Unterprogramm-Aufruf)

...ist erforderlich, um ein Unterprogramm aufzurufen.

Vgl. die Beschreibungen bei: Unterprogramme !

Unterprogramme werden bei PAL mit **L** aufgerufen, gefolgt von der zweistelligen Programm-Nr. und der zweistelligen Anzahl ihrer Aufrufe. z.B.:

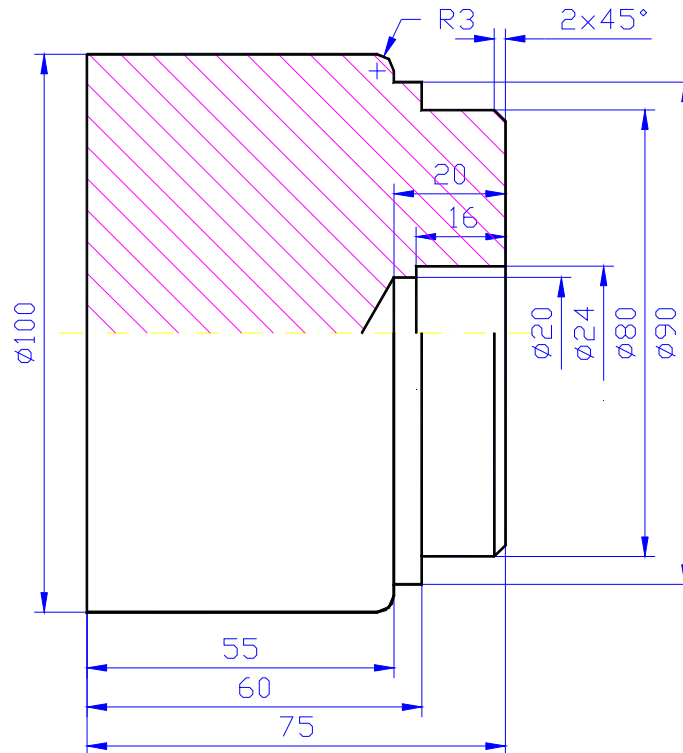
z.B.: **L1302** ruft das Unterprogramm Nr. 13 auf und arbeitet es 2x ab (1 Wiederholung).



zur nächsten Seite

^K L; Unterprogramm-Aufruf
[#] IDH_L
[§] L-Befehl

K\$# Übungsaufgabe Nr. 1

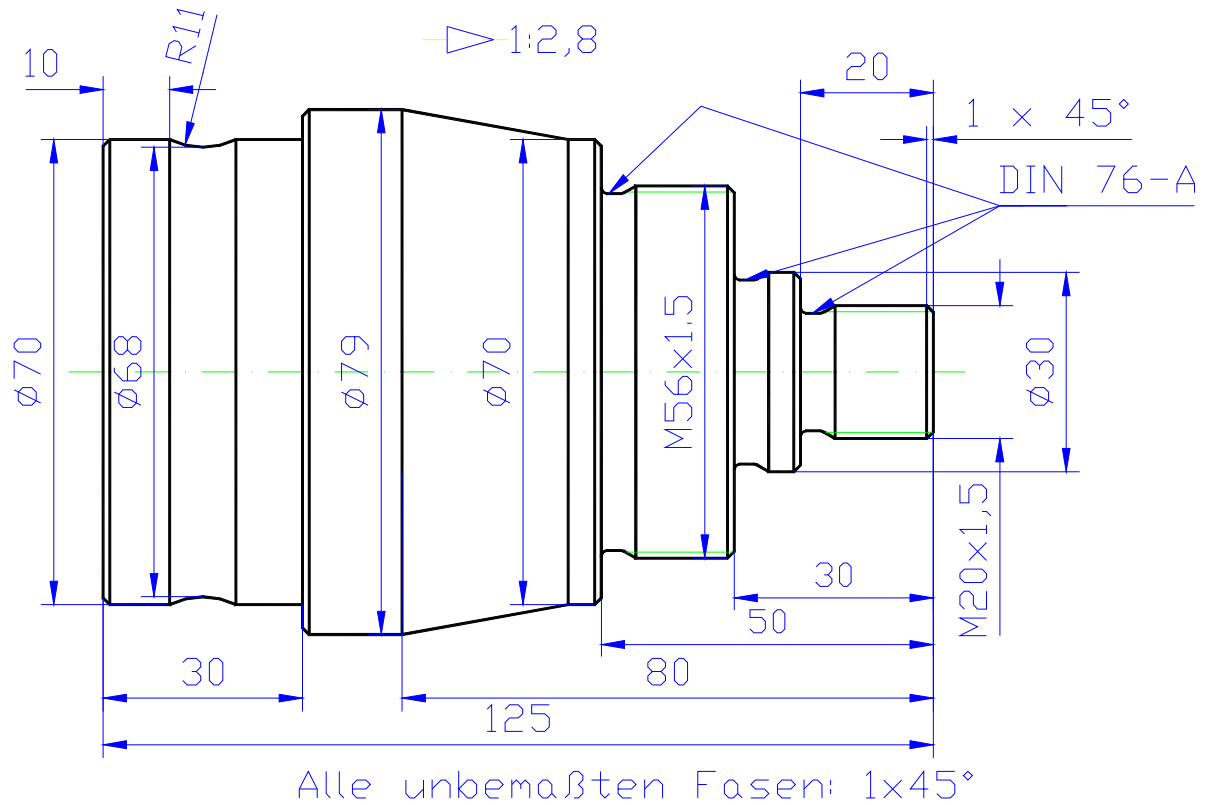
Erstelle ein CNC-Programm für folgendes Drehteil aus EN AW-Al Mg2 !
(Störe dich nicht daran, dass die Schraffur im Halbschnitt eigentlich in die untere Hälfte gehört!)




zur nächsten Aufgabe

K\$# Übungsaufgabe Nr. 2

Erstelle ein CNC-Programm für folgende Welle aus 10 S 20 !



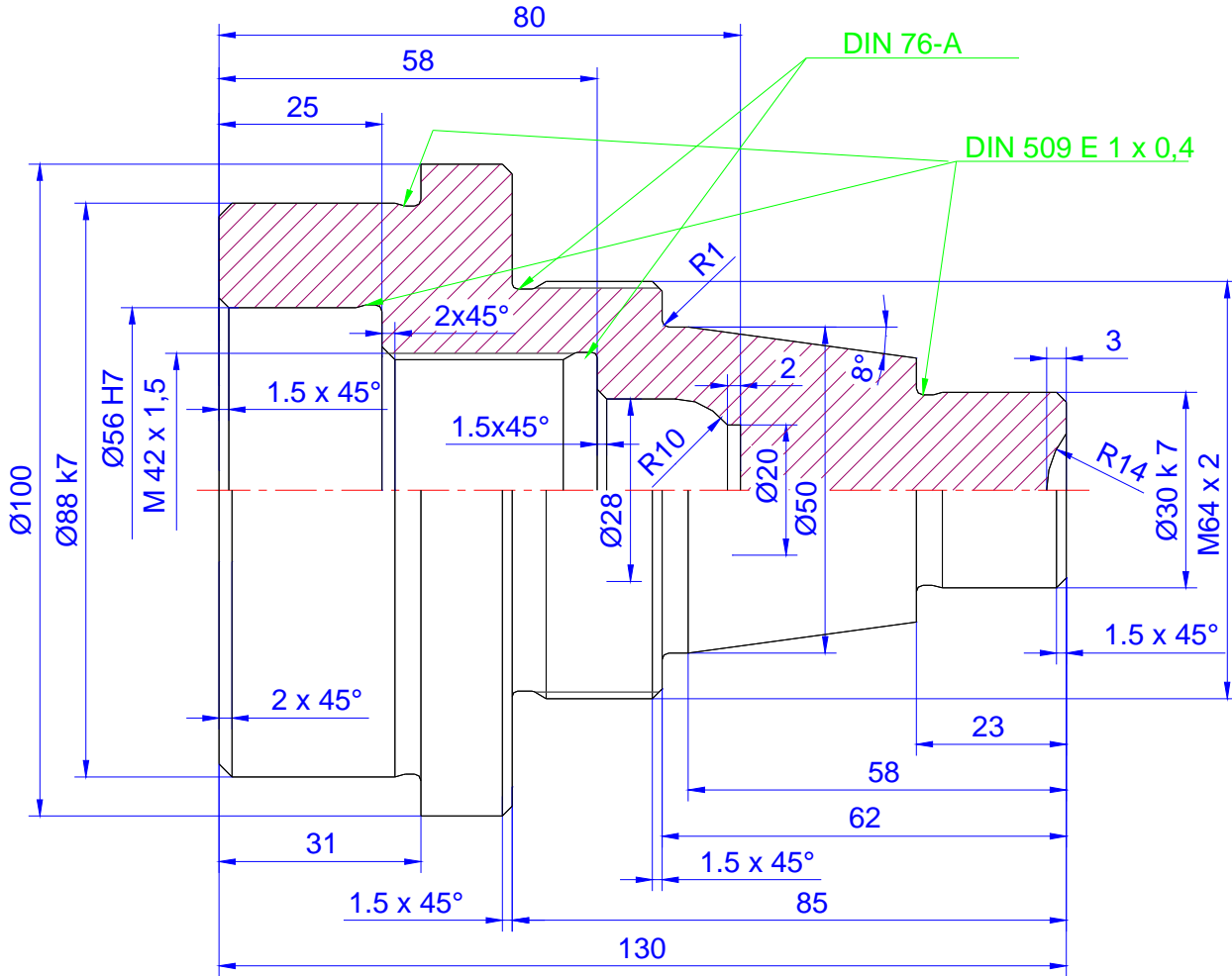
→
zur nächsten Aufgabe

K\$# Übungsaufgabe Nr. 3b

Erstelle ein CNC-Programm für folgenden Bolzen aus 11SMn30 !

Verwende dazu die Unterprogrammtechnik!

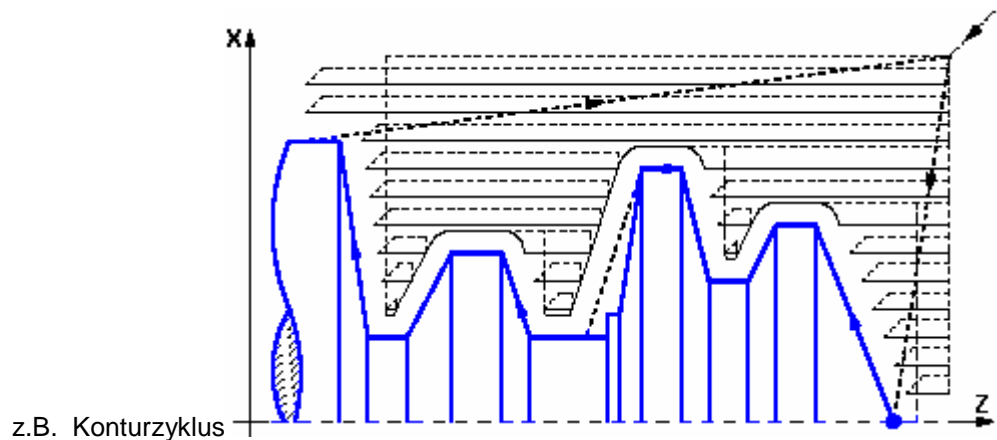
(Störe dich nicht daran, dass die Schraffur im Halbschnitt eigentlich in die untere Hälfte gehört!)



→
zur nächsten Seite

K\$# Ausblick

Die hier beschriebenen und von euch einzuübenden PAL-CNC-Funktionen sind zwar vollständig (mehr wird in der schriftlichen Gesellenprüfung nicht verlangt), aber sie sind noch lange nicht das, was „richtige“ Steuerungen zu bieten haben bzw. von einem Facharbeiter abverlangen.

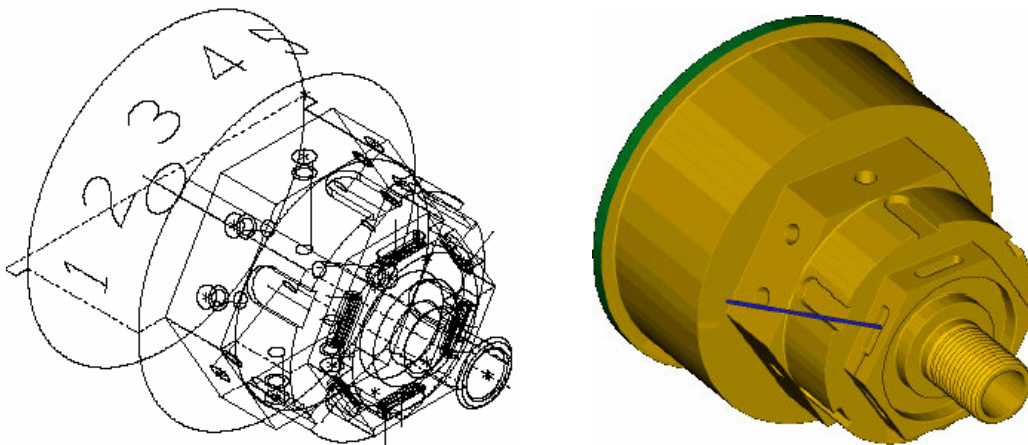


Der Befehlsumfang üblicher NC-Steuerungen sind viel umfangreicher. Sie haben eine Vielfalt verschiedenster Zyklen, benutzen angetriebene Werkzeuge, Setzstöcke, Gegenfutter usw. Sie erlauben eine grafische Konturerstellung im Dialog, interpolieren in mehr als 3 Maschinenachsen und erlauben ein Programmieren in Hochsprache. Dadurch können – bes. durch Zugriff auf Variable, Zustände und Messwerte -, eigene, angepasste Zyklen geschrieben werden.

So verrechnet z.B. ein Bad Sobernheimer Unternehmen die störende Wärmedehnung von noch nicht ganz abgekühlten Schmiedeteilen innerhalb eines CNC-Fräsprogrammes mittels Temperaturmessung. Oder eine immer bekannter werdende Firma aus Stipshausen erweitert den Funktionsumfang einer Steuerung, in dem sie spez. Zyklen für ihre selbstentwickelten Ultraschallmaschinen programmiert.

Andererseits hat das manuelle Programmieren in der betrieblichen Praxis eine starke Konkurrenz: das maschinelle Programmieren mittels **CAD-CAM**:

Die meisten Programme werden heute mittels Zusatzsoftware direkt aus den CAD-Daten (und aus einem Pool von Technologie-Daten) generiert und mit Hilfe von sog. Postprozessoren an die jew. vorhandene Maschinensteuerung (Gildemeister, Siemens, Traub, Index, Heidenhain, Fagor, Fanuc, ...) angepasst. Dabei ist bei Änderungen nicht mehr der eigentliche CNC-Programmcode von Interesse, sondern es wird ein übergeordneter Quellcode verwendet, der übersichtlich bleibt und einfacher zu ändern ist.



^K Ausblick; Entwicklungstendenzen

^S Ausblick

[#] IHD_Ausblick

Wer aber jetzt glaubt, die mühsame manuelle Programmierung mit der „popligen“ PAL-CNC sei für ihn unwichtig geworden, täuscht sich gewaltig. Die Grundlagen sind wichtig. Nur auf diesen lässt sich Kompetenz aufbauen. Back to the roots. (Wer z.B. mit HTML-Dateien arbeitet, sieht sicher analoge Entwicklungen und wird mir Recht geben.)

Wichtig ist ebenso, dass man sich darüber im Klaren ist, dass man nur durch virtuoson Umgang mit Simulationssoftware noch lange kein CNC-Spezialist ist. Unbedingt erforderlich ist es, aufbauend auf den gewonnen Erkenntnissen, min. eine Steuerung an einer realen Maschine kennen gelernt zu haben. Möglichkeiten dazu bietet unsere zwar gebrauchte, aber voll funktionsfähige 5-Achsen-Fräsmaschine im Raum 119, die unser aller Stolz ist, die gleichen „Features“ wie große Fräszentren besitzt und (unserm Schulleiter sei Dank) ausschließlich von Spendengeldern Idar-Obersteiner Unternehmen beschafft wurde:



Jeder Schüler sollte sich möglichst schnell fit machen, um an der Maschine zumindest einmal

- eine Referenzfahrt durchzuführen,
- den Werkstücknullpunkt einzustellen,
- ein einfaches CNC-Programm samt Taschenfräszyklus zu programmieren, über die DNC einzugeben und abzuarbeiten,
- die Technologiewerte an Hand der erzielten Oberflächen korrigieren bzw. optimieren,
- ein z.B. nachgeschliffenes Werkzeug an der Maschine neu zu vermessen und in den Werkzeugkorrekturspeicher einzugeben und evtl.
- mit den Besonderheiten und erweiterten Möglichkeiten der Steuerung vertraut zu machen.

H. Klinkner, Frühsommer 2002