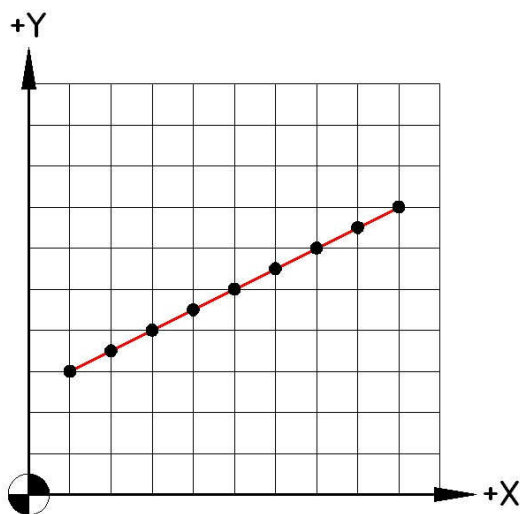


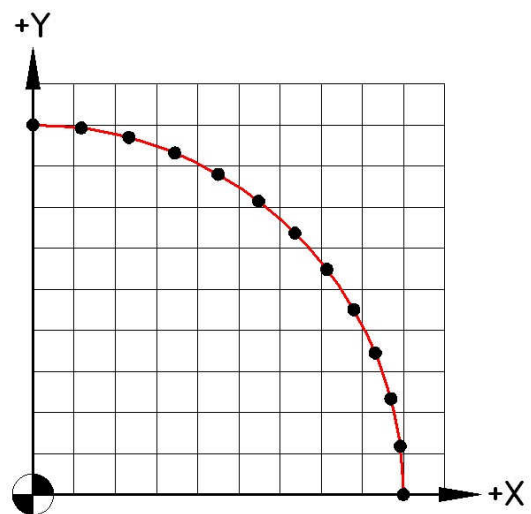
Kreisprogrammierung

Zuerst einige Worte zur **Interpolation**:

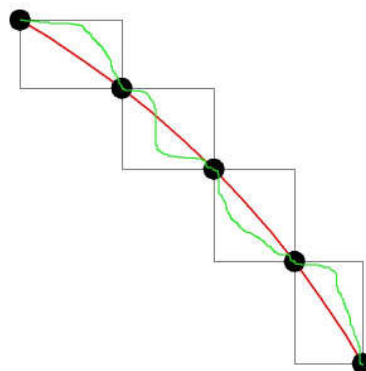
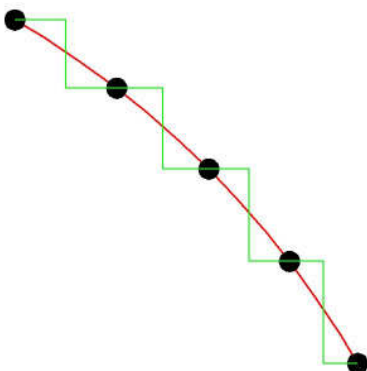
Wenn eine mathematische Formel grafisch als Punkte dargestellt wird, die zu einer (angenäherten) Geraden oder Kurve verbunden werden, dann ist dies eine Interpolation im Sinne der CNC-Technik. Jede Gerade, jeder Kreis und andere mathematische Grundgebilde wie Parabel, Hyperbel, usw. werden üblicherweise durch eine Formel definiert. Man kann zu einer Reihe vorher bestimmter X-Werte die zugehörigen Y-Werte nach der Formel ausrechnen (Wertetabelle) und in ein Schaubild eintragen und danach die Punkte in der richtigen Reihenfolge untereinander verbinden. Dadurch erhält man ein Schaubild (Graph) der betreffenden Formel.



Geraden-Interpolation
mit Ideallinie (rot)



Kreis-Interpolation
mit Ideallinie



Die Berechnung der Interpolation erfolgt in Stufen, je nach Auflösung in 1/1000 oder 1/100mm-Schritten, so dass dies oft als Treppe dargestellt wird, wie im linken Bild. In Wirklichkeit wird jedoch nie die rote Ideallinie, sondern eine beliebige Kurve innerhalb des überwachten Toleranzbereiches der Maschine gefahren. Dies ist von vielen Faktoren abhängig, wie vom Spiel der Achsantriebsspindeln, welche Achse schwergängiger ist, welcher Achsantrieb zuerst anspricht, usw.

Grundsätzliches zur CNC-Kreisprogrammierung nach DIN / ISO:

Die gewünschte Arbeitsebene G17, G18, G19 muss vorher definiert sein.
Vor dem Satz mit dem Kreisbefehl muss zuerst der Anfangspunkt des Kreises angefahren sein.

Der Kreisbogen wird je nach Drehrichtung mit G02 oder G03 bestimmt, dazu werden im selben Satz (Zeile) der Endpunkt und die Interpolationsparameter oder der Radius definiert, um den Kreisbogen auszuführen.

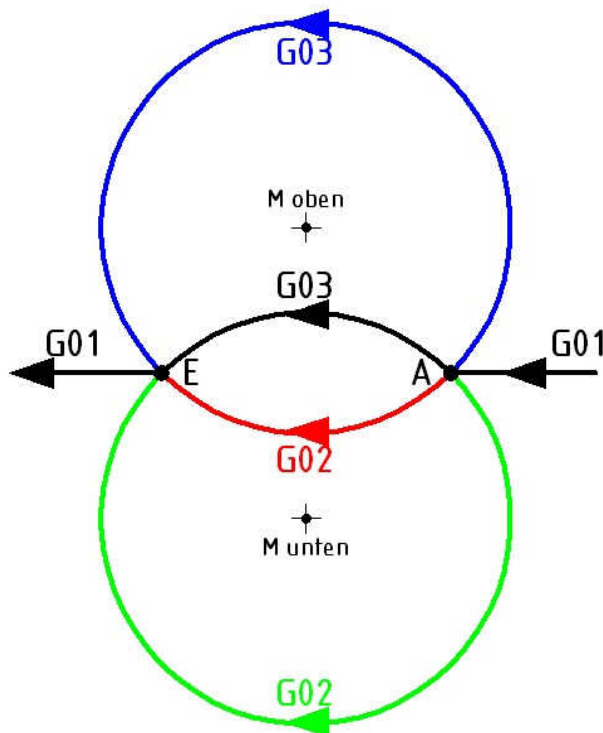
Wenn im nächsten Satz eine Gerade gefahren werden soll, nicht vergessen, G01 oder G00 zu schreiben.

Bei der Programmierung eines Vollkreises (ganzer Kreis, 360 Grad) kann es bei sehr alten Steuerungen vorkommen, dass dies nicht funktioniert. Eigentlich logisch, denn der Anfangspunkt ist gleich dem Endpunkt, das heißt: keine Bewegung.

Die heutigen Steuerungen führen natürlich den gewünschten Vollkreis aus.

Wer mit einer Billig- oder Uralt-Steuerung arbeiten muss, kann sich so behelfen:
Erste Möglichkeit: zwei Halbkreise programmieren. Allerdings entsteht dabei eine weitere Freischneidmarke am Übergang der beiden Halbkreise.

Zweite Möglichkeit: den Endpunkt nicht exakt programmieren, sondern in einer Achse z. B. 1 bis 2 tausendstel mm (abhängig von der Eingabefeinheit der Steuerung) vorher anhalten. Idealerweise in der Achse, wo die letzte Hauptfahrtrichtung zum Endpunkt erfolgt. Also: wenn Endpunkt oben oder unten, dann Abweichung in X-Achse, wenn Endpunkt links oder rechts, dann Abweichung in der Y-Achse eingeben. Der Differenzwert darf nur nicht größer sein als die Kreisendpunktüberwachung in den Maschinendaten (meistens 0,01 bis 0,02 mm). Diese minimale Abweichung ist am Werkstück normalerweise nicht sichtbar.



Um einen Kreisbogen mit bekanntem Radius vom Anfangspunkt A zum Endpunkt E zu programmieren, wären hier immerhin vier Möglichkeiten denkbar.

Es müssen dazu noch die Bewegungsrichtung des Kreisbogens angegeben und sein Mittelpunkt oder Radius bestimmt werden.

Zum Beispiel müssen wir bei dem schwarzen Kreisbogen G3 angeben und die Interpolationsparameter vom Anfangspunkt des Kreises zum unteren Kreismittelpunkt angeben. Bei vielen Steuerungen kann anstatt der Interpolationsparameter auch der Radius angegeben werden.

Vergleich der Programmierung mit Interpolationsparametern oder mit Radius

Die Kreisprogrammierung mit Hilfe der Interpolationsparameter ist die ursprüngliche und bei alten Steuerungen die einzige Möglichkeit, eine Kreisbewegung zu programmieren. Die Interpolationsparameter sind jeweils der Abstand vom Anfangspunkt des Kreises zum Kreismittelpunkt und zwar I in X-Richtung, J in Y-Richtung, K in Z-Richtung. Dafür entfällt die Angabe des Radius. Wenn in G17-Ebene (also in X- und Y-Achse) gearbeitet wird, dann gelten I und J. Beim Drehen mit der X- und Z-Achse werden I und K verwendet. Es ist oft lästig, diese Abstände auszurechnen und sehr häufig wird das Vorzeichen von I, J und K falsch gesetzt, was manchen schon zur Verzweiflung gebracht hat.

Merke: I, J und K sind die Werte **vom** Anfangspunkt **zum** Kreismittelpunkt. Damit wird auch das Vorzeichen definiert.

Im o. a. Beispiel sind I und J für den schwarzen Kreisbogen beide negativ (vom Punkt A zum Mittelpunkt unten). Wenn der blaue Kreisbogen gefahren werden soll, dann wäre I weiterhin negativ, aber J positiv, denn dann beziehen wir uns ja auf den Mittelpunkt des oberen Kreises.

Vorteile der (umständlichen) Programmierung mit I, J und K:

1. funktioniert gewöhnlich auf jeder Maschine
2. in den Maschinendaten wird eine Toleranz für die Kreisendpunktüberwachung eingegeben und so werden unzulässige Abweichungen sofort erkannt.
Wer es genau wissen will: Mit Anfangspunkt, Endpunkt und nur einem der beiden Interpolationsparameter wäre ein Kreis schon auszurechnen. Durch den zweiten Interpolationsparameter ist der Kreis mathematisch „überbestimmt“. Deshalb ist die Kontrolle möglich.

Bei der Kreisprogrammierung mit Radiusangabe entfällt die Angabe von I, J und K. Dafür wird der Radius angegeben und zwar als R für Kreisbögen bis 180 Grad (Halbkreis), für Kreisbögen über 180 Grad wird R negativ angegeben (R-...). Der schwarze und rote Bogen wird somit mit R... definiert, der blaue und grüne mit R-... Nachteile der (einfacheren) Programmierung mit R und R-:

1. Geht nicht bei jeder Steuerung
2. Keine Fehlererkennung möglich
3. Keine Vollkreisprogrammierung möglich

Programmierung der oben abgebildeten Kreisbögen:

Mit I und J

-Schwarzer Kreisbogen

G1 X... Y...

G3 X... Y... I-... J-...

-Blauer Kreisbogen

G1 X... Y...

G3 X... Y... I-... J...

-Roter Kreisbogen

G1 X... Y...

G2 X... Y... I-... J...

-Grüner Kreisbogen

G1 X... Y...

G2 X... Y... I-... J-...

Mit R und R-

G1 X... Y...

G3 X... Y... R...

G1 X... Y...

G3 X... Y... R-...

G1 X... Y...

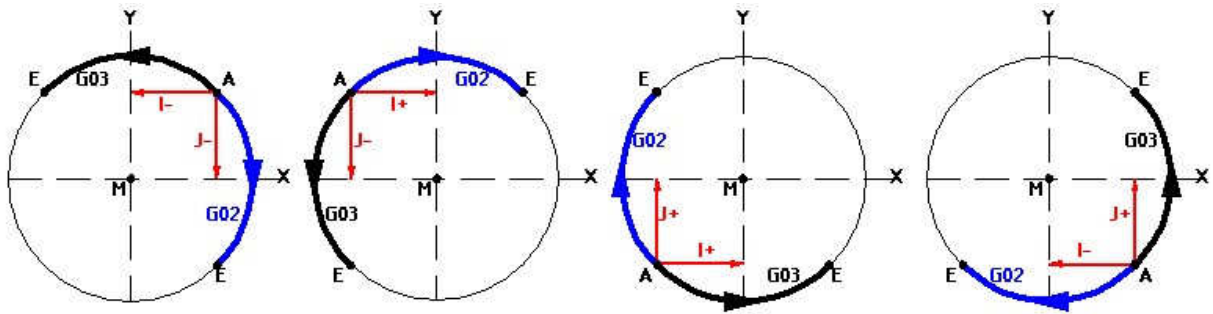
G2 X... Y... R...

G1 X... Y...

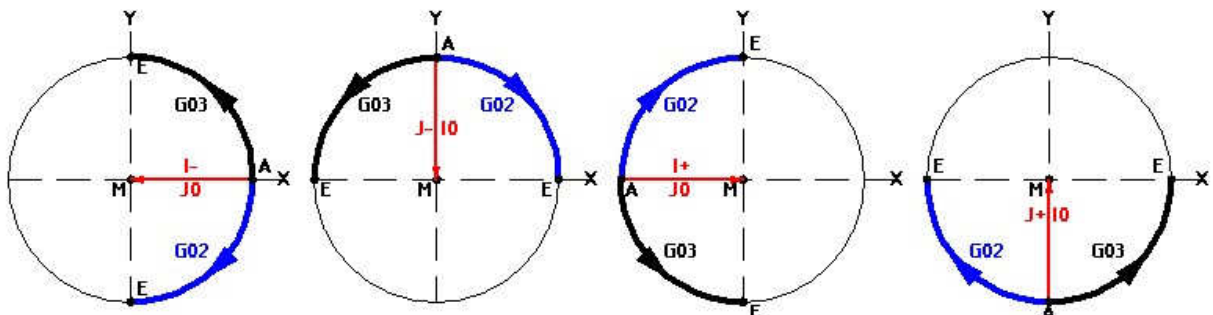
G2 X... Y... R-...

Anfangspunkt A

Endpunkt E



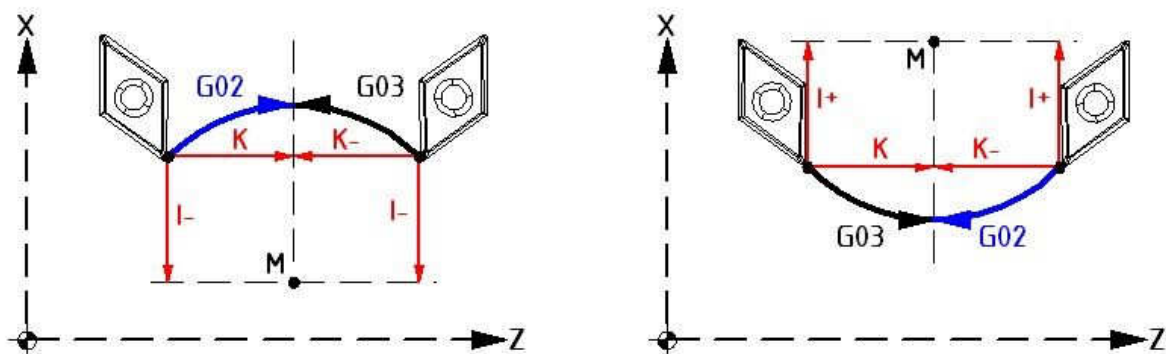
Die Vorzeichen von I und J in Abhängigkeit von der Lage des Anfangspunktes in der XY-Ebene (G17).



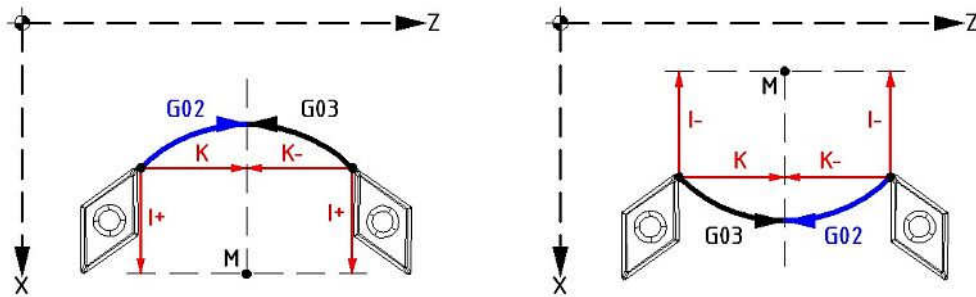
Sonderfälle: wenn in X oder Y kein Versatz ist, dann bekommen I oder J den Wert Null

Bei den anderen Ebenen G18 (X, Z) und G 19 (Y, Z) verhält es sich mit I und K oder J und K dementsprechend. Die Ebene kann wechseln, wenn Kreisbögen in die Höhe oder Tiefe ausgeführt werden oder wenn eine Universalfräsmaschine von Horizontal- auf Vertikalbetrieb umgebaut wird.

Bei Drehmaschinen wird gewöhnlich die G18-Ebene (X,Z) verwendet.



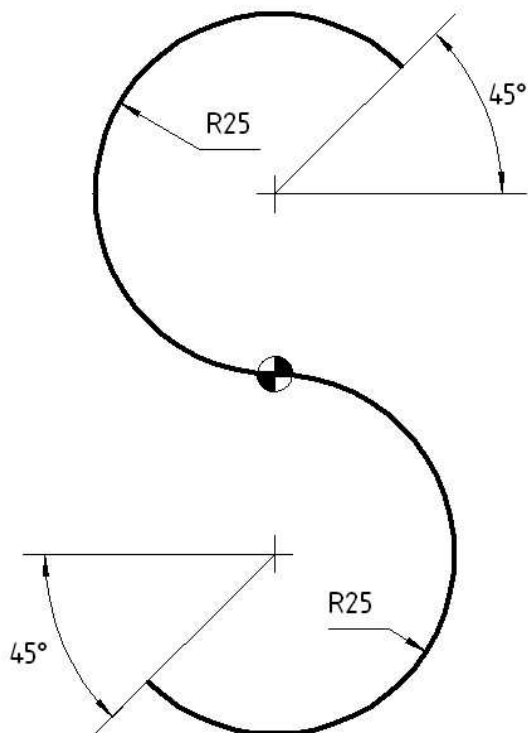
Kreisbögen bei Drehmaschinen mit Bearbeitung hinter der Mitte (Normalfall)



Kreisbögen bei Drehmaschinen mit Bearbeitung vor der Mitte

Hinweis: bei manchen Steuerungen ist die Angabe der Mittelpunktsparemeter auch absolut auf den Nullpunkt bezogen möglich oder sogar vorgeschrieben. Die Werte für I, J und K werden dann durch den Abstand vom Werkstück-Nullpunkt zum Kreismittelpunkt ersetzt. Dazu müssen eventuell andere G-Befehle benutzt, oder die Interpolationsparameter durch neue Bezeichnungen ersetzt werden.

Programmierübung



Hier eine S-Kurve als Übung zur Kreisprogrammierung in der G17-Ebene (X und Y).

Der Nullpunkt sitzt in der Mitte der Kurve.

Zuerst errechnen wir alle relevanten Punkte:

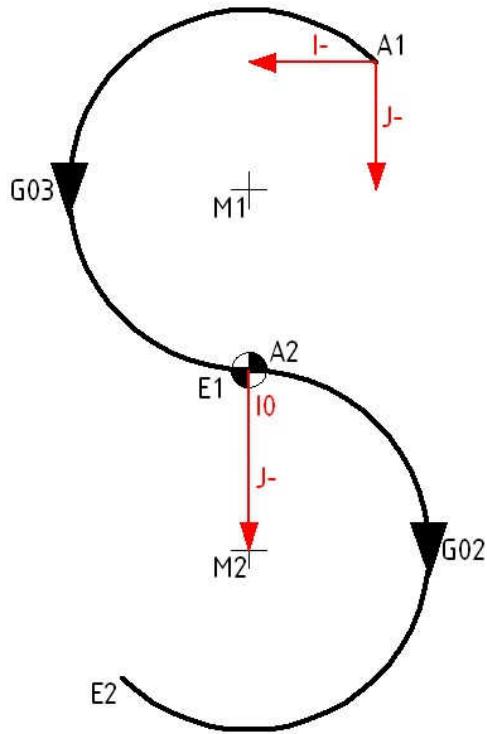
Anfangspunkt oben $X=17,678$, $Y=42,678$

Mittelpunkt oberer Kreis $X=0$, $Y=25$

Übergang oberer in unteren Kreis $X=0$, $Y=0$

Mittelpunkt unterer Kreis $X=0$, $Y=-25$

Endpunkt unten $X=-17,678$, $Y=-42,678$



Kurve von oben nach unten graviert:

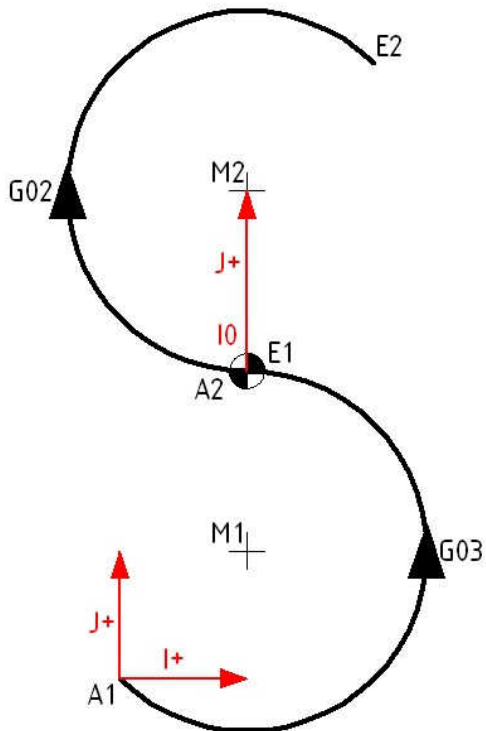
```

%
:0001
N0005 G17 G90 G94 G40 G49
N0010 T1 M6
N0015 S3500 M3
N0020 G0 G54 X17.678 Y42.678 M8
N0025 G43 Z2 H1
N0030 G1 Z-0.5 F150
N0035 G3 X0 Y0 I-17.678 J-17.678
N0040 G2 X-17.678 Y-42.678 I0 J-25
N0045 G0 Z50 M9
N0050 M5
N0055 G28 G49
N0060 M30
%
```

oder:

```

N0035 G3 X0 Y0 R-25
N0040 G2 X-17.678 Y-42.678 R-25
```



Kurve von unten nach oben graviert

```

%
:0003
N0005 G17 G90 G94 G40 G49
N0010 T1 M6
N0015 S3500 M3
N0020 G0 G54 X-17.678 Y-42.678 M8
N0025 G43 Z2 H1
N0030 G1 Z-0.5 F150
N0035 G3 X0 Y0 I17.678 J17.678
N0040 G2 X17.678 Y42.678 I0 J25
N0045 G0 Z50 M9
N0050 M5
N0055 G28 G49
N0060 M30
%
```

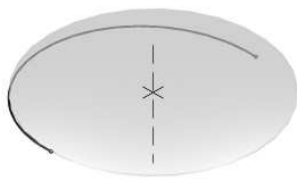
oder:

```

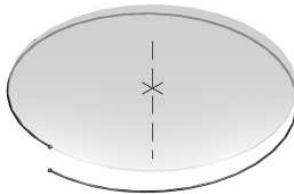
N0035 G3 X0 Y0 R-25
N0040 G2 X17.678 Y42.678 R-25
```

Helix-Programmierung

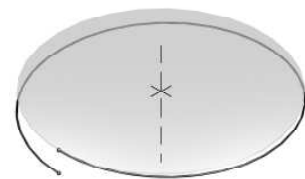
Bei 3D-Steuerungen kann auch eine Schraubenlinie (Helix) programmiert werden. Zu den X- und Y- Werten für den Kreisendpunkt kommt noch zusätzlich ein Z-Wert, welcher bewirkt, dass während der Kreisbewegung in X und Y eine Zu- oder Abnahme der Z-Tiefe erfolgt, was eine Schraubenlinie erzeugt (Helix-Interpolation). Dies kann z. B. zum Gewindefräsen verwendet werden (auch für mehrgängige Gewinde).



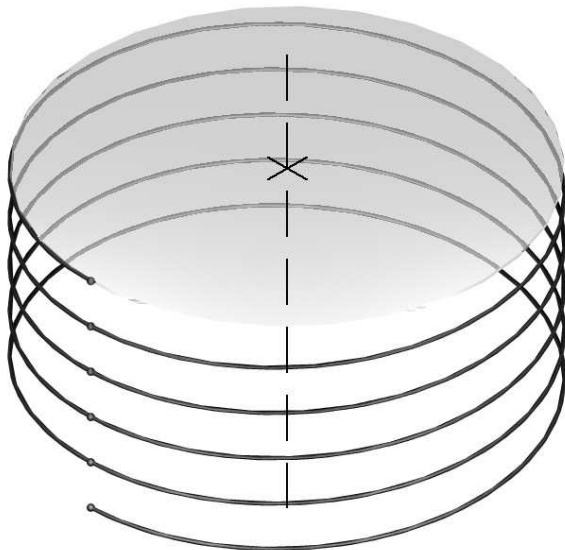
Halbkreis nach unten
rechtsdrehend
oder nach oben
linksdrehend



Vollkreis nach unten
rechtsdrehend
oder nach oben
linksdrehend



Vollkreis nach unten
linksdrehend
oder nach oben
rechtsdrehend



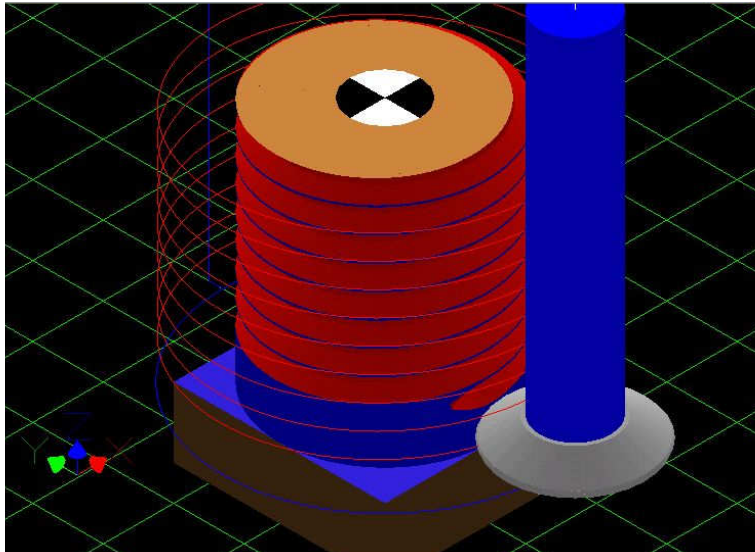
Mehrfach-Helix mit 5 Windungen

Einige Steuerungen ermöglichen die Definition der Anzahl der Windungen mittels eines zusätzlichen Parameters im Programmsatz.

Andere Steuerungen benutzen Z als Gesamttiefe und K als Steigung.

Wer dies nicht hat, programmiert die Bewegung mit G91 (Kettenmaß), legt diese in einem Unterprogramm ab und ruft dann das Unterprogramm beliebig oft auf.

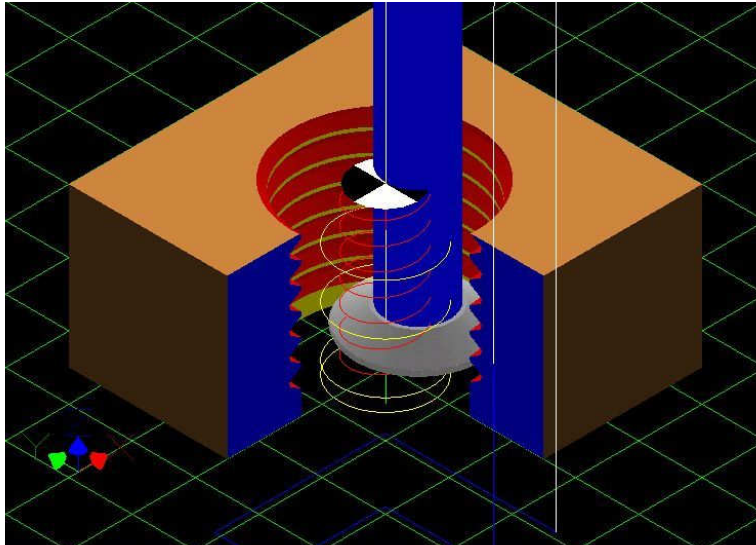
Die nachstehenden Beispiele wurden auf die selbe Weise für die CNC-Steuerung Fanuc 6M erstellt.



Fräsen eines Außengewindes an einem Zapfen.
Zur besseren Übersicht wurde hier nur die entscheidende Gewindebearbeitung aufgelistet.
In diesem Beispiel wird ohne Fräserradiuskorrektur gearbeitet.

```

%
:0010
N1005 G17 G90 G94 G40 G49 G21
N1010 T4 M6
.
.
.
N1065 T7 M6
N1070 S2500 M3
N1075 G0 X0 Y-27
N1080 G43 Z2 H7
N1085 G1 Z0 F100
(Bolzenradius + WKZ-Radius - P*0.613)
(15 + 10 - 2,147 = 22,853)
N1090 Y-22.853
N1095 M98 L9 P1001
N1100 G1 Y-27
N1105 G0 Z50
N1110 M5
N1115 G28 G49
N1120 M30
%
%
:1001
N1125 G91
N1130 G2 X0 Y0 Z-3.5 I0 J22.853
N1135 G90
N1140 M99
%
```

Fräsen eines Innengewindes
in einer Bohrung.
Die vorhergehenden
Bearbeitungsschritte wurden
ebenfalls weggelassen.
Auch hier wurde keine
Fräserradiuskorrektur benutzt.

Das Bild zeigt die Bearbeitung
in der CNC-Simulation
CNCezPRO.

```

%
:0030
N1005 G17 G90 G94 G40 G49 G21
N1010 T1 M6
.
.
.
N1210 T7 M6
N1215 S2500 M3
N1220 G0 X0 Y0
N1225 G43 Z2 H7
N1230 G1 Z0 F100
(Kernlochradius - WKZ-Radius + P*0.541)
(13,25 - 10 + 1,895 = 5,145)
N1235 X5.145
N1240 M98 L8 P3001
N1245 G1 X0 Y0
N1250 G0 Z50
N1255 M5
N1260 G28 G49
N1265 M30
%
%
:3001
N1270 G91
N1275 G2 X0 Y0 Z-3.5 I-5.145 J0
N1280 G90
N1285 M99
%
```