

AFS - CNC-Multiprogrammingsystem(R) V1.00

Inhaltsverzeichnis:

Allgemeines

Syntax

Befehlsreferenz

Allgemeines

Das AFS - CNC-Multiprogrammingsystem dient zur Anpassung der neusten AFS-Softwareprodukte [AFS (CAD to CNC) Profi-System, AFS-CNC Fräsen Profisystem & AFS-CNC Drehen Profisystem] an fast alle gängigen CNC-Steuerungen.

Dies ist wohl die einzigste Methode, um kostengünstig einen CNC-Simulator (in dem ja fast alle Befehle vorhanden sind) an die verschiedensten CNC-Steuerung anzupassen.

Das AFS - CNC-Multiprogrammingsystem ist eine speziell für diesen Zweck entwickelte Programmiersprache, sehr einfach zu erlernen ist.

Arno Freisinger

Dezember 1992

Syntax

Das AFS - CNC-Multiprogrammingsystem ist eine Programmiersprache, die Ihnen die Möglichkeit gibt, Ihre Steuerung an AFS (CAD to CNC) Profi-System anzupassen.

Die Steuerung wird in mehrere Definitionsteile gegliedert.

Haupt_Kopf

Unter_Kopf

GlobalerProgrammkopf

Global

Befehle

GlobalesProgrammende

Programm_Abschluß

Die einzige Funktion, die außerhalb dieser Definitionen liegen darf sind Kommentare.

Die einzelnen Definitionsteile werden durch geschweifte Klammern '{ }' begrenzt.

Die erste Klammer '{' muß direkt und ohne Freizeichen hinter dem Definitionnamen stehen, z.B. Haupt_Kopf{ ...

Die Begrenzungsklammer sollte immer alleine in einer Zeile stehen. Siehe mitgelieferte Steuerungen.

Befehle, die außerhalb dieser Klammern stehen werden nicht verarbeitet!

Befehlsreferenz

nur in 'Global{' gültige Befehle

<u>Nummerierung_0</u>	Definiert das Zeichen, das vor der CNC-Nummerierung erscheint.
<u>Nummerierung_1</u>	Definiert die CNC-Nummerierung.
<u>Kommentar</u>	Definiert, wie der Kommentar eingestellt wird.
<u>Vorschub</u>	Definiert, wie der Vorschub angegeben wird.
<u>Drehzahl</u>	Definiert, wie die Drehzahl angegeben wird.
<u>Werkzeuge</u>	Definiert, wie die Werkzeuge angegeben werden.

nur in 'Befehle{' gültige Befehle

<u>Programm_Stop</u>	Definiert den 'einfachen' Programm halt.
<u>Spindel_Rechtslauf</u>	Definiert den Spindelrechtslauf.
<u>Spindel_LinksLauf</u>	Definiert den Spindellinkslauf.
<u>Spindel_Stop</u>	Definiert den Spindelstop.
<u>Werkzeugwechsel_1</u>	Definiert den Werkzeugwechsel.
<u>Kühlmittel1_Ein</u>	Definiert, wie die Kühlmittelzufuhr eingestellt wird.
<u>Kühlmittel1_Aus</u>	Definiert, wie die Kühlmittelzufuhr abgestellt wird.
<u>Programm_End</u>	Definiert, wie das Programm beendet wird.
<u>Eilgang</u>	Definiert den Eilgang.
<u>Geraden</u>	Definiert die geradlinige Bewegung mit Vorschub.
<u>Kreis20</u>	Definiert die kreisförmige Bewegung, im Uhrzeiger-

	sinn mit Radiusangabe.
<u>Kreis30</u>	Definiert die kreisförmige Bewegung, gegen den Uhrzeigersinn mit Radiusangabe.
<u>Kreis21</u>	Definiert die kreisförmige Bewegung, im Uhrzeigersinn mit Mittelpunktsangabe.
<u>Kreis31</u>	Definiert die kreisförmige Bewegung, gegen den Uhrzeigersinn mit Mittelpunktsangabe.
<u>EbeneXY</u>	Definiert die XY-Ebene.
<u>EbeneXZ</u>	Definiert die XZ-Ebene.
<u>EbeneZY</u>	Definiert die ZY-Ebene.
<u>Korrektur_Aus</u>	Schaltet die Radiuskorrektur aus.
<u>Korrektur_Links</u>	Schaltet die Radiuskorrektur links ein.
<u>Korrektur_Rechts</u>	Schaltet die Radiuskorrektur rechts ein.
<u>Korrektur_Bis</u>	Schaltet die Radiuskorrektur bis (-) ein.
<u>Korrektur_Über</u>	Schaltet die Radiuskorrektur über (+) ein.
<u>Maß_Zoll</u>	Schaltet die Maßangaben in Zoll ein.
<u>Maß_mm</u>	Schaltet die Maßangaben in mm ein.
<u>Bezugmaß</u>	Schaltet die absolute Bemaßung ein.
<u>Kettenmaß</u>	Schaltet die inkrementale Bemaßung ein.
<u>Darstellungsfenster_1</u>	Darstellungsfensterdefinition.
<u>Rohteil_1</u>	Rohteildefinition.
<u>Darstellungsfenster_2</u>	Darstellungsfensterdefinition.
<u>Rohteil_2</u>	Rohteildefinition.

Funktion: // Kommentar

Art: Funktion (globale Definition)

Erläuterung: // leitet einen beliebigen Kommentar ein.
Alle Zeichen nach // werden nicht mehr berücksichtigt.

Beispiel: siehe GlobalerProgrammKopf

Abschnitt:

```
GlobalerProgrammKopf{  
  ...  
  ...  
  ...  
}
```

Art:

Abschnitt (globale Definition)

Erläuterung:

GlobalerProgrammKopf leitet einen Abschnitt ein, indem die ersten Programmzeilen definiert werden können, d.h. wenn Sie in der (den) ersten Programmzeile(n) immer die gleiche Befehlsfolge benutzen, so kann diese hier eingetragen werden. Diese werden immer automatisch als erste CNC-Programmzeilen in einem CNC-Programm eingefügt.

Hier kann der Text direkt, **ohne** die eckigen Klammern und **ohne** Zeilennummern angegeben werden

Beispiel:

```
GlobalerProgrammKopf{ // leitet den Abschnitt ein  
G18 M00 // beliebige definition  
G54 M06 T00 // " "  
} // muß immer am Ende eines  
// Abschnitts stehen
```

Abschnitt:

```
GlobalesProgrammende{  
...  
...  
...  
}
```

Art:

Abschnitt (globale Definition)

Erläuterung:

GlobalesProgrammende leitet einen Abschnitt ein, indem die letzten Programmzeilen definiert werden können, d.h. wenn Sie in den letzten Programmzeilen immer die gleiche Befehlsfolge benutzen, so kann diese hier eingetragen werden. Diese werden immer automatisch als letzte CNC-Programmzeilen in einem CNC-Programm eingefügt, z.B.: Werkzeug auf Wechselpunkt fahren.

Hier kann der Text direkt, **ohne** die eckigen Klammern und **ohne** Zeilennummern angegeben werden

Beispiel:

```
GlobalesProgrammende{    // leitet den Abschnitt ein  
  
G0 X0 Y0 Z150            // z.B. aus dem Werkstück fahren  
  
}                        // muß immer am Ende eines  
                        // Abschnitts stehen
```


Abschnitt:

```
Programm_Abschluß{  
...  
...  
...  
}
```

Art: Abschnitt (globale Definition)

Erläuterung: **Programm_Abschluß** leitet einen Abschnitt ein, indem das Ende eines CNC-Programmes definiert ist, d.h. ob noch Steuerungsspezifische Codes (Zeichen) an das CNC-Programm gehängt werden sollen.

Beispiel:

```
Programm_Abschluß{           // leitet den Abschnitt ein  
  
[Programm Ende]             // gibt den Text in den Klammern aus  
EOT                          // sendet das ASCII-Zeichen 4 (EOT)  
  
}                             // muß immer am Ende eines  
                             // Abschnitts stehen
```

'Normaler' Text wird in die eckigen Klammern geschrieben.
Mit dem Befehl EOT können Sie ASC-Zeichen (End of Transmission) angeben. Befindet sich der EOT-Befehl hinter einer Klammer, so müssen Sie ein Pluszeichen '+' einsetzen.

2. Beispiel:

```
Programm_Abschluß{           // leitet den Abschnitt ein  
  
[Programm Ende]+EOT  
  
}                             // muß immer am Ende eines  
                             // Abschnitts stehen
```

3. Beispiel:

```
Programm_Abschluß{           // leitet den Abschnitt ein  
  
EOT                          // z.B. Philipssteuerung  
  
}                             // muß immer am Ende eines  
                             // Abschnitts stehen
```

Abschnitt:

```
Global{
...
...
...
}
```

Art: Abschnitt (globale Definition)

Erläuterung: **Global** leitet einen Abschnitt ein, indem globale Definitionen gemacht werden können. Globaledefinitionen sind z.B. die Nummerierung, Kommentar, Vorschub, ...

Beispiel:	Global{	// leitet den Abschnitt ein
	Nummerierung_0 = Zeile	// schreibt 'Zeile 1 ...'
	Nummerierung_1 = 1	// 1'er Nummerierung
	Vorschub = F	// Vorschub wird mit F Angegeben
	...	
	}	// muß immer am Ende eines
		// Abschnitts stehen

Abschnitt:

```
Haupt_Kopf{  
...  
...  
...  
}
```

Art: Abschnitt (globale Definition)

Erläuterung: **Haupt_Kopf** leitet einen Abschnitt ein, indem die Zeilen definiert werden, die ein Hauptprogramm enthält vor der ersten Programmzeile. Meistens werden hier die Programmnummer und ein Zeichen angegeben.

Beispiel:

```
Haupt_Kopf{           // leitet den Abschnitt ein  
[%PM]                // schreibt %PM in die erste Zeile  
[N]+Programmnummer   // schreibt z.B. N9002  
}                     // muß immer am Ende eines  
                     // Abschnitts stehen
```

'Normaler' Text wird in die eckigen Klammern geschrieben und dort, wo die Programmnummer stehen soll geben Sie Programmnummer ein. Steht die Programmnummer nach einer Klammer, so geben Sie einfach noch das Pluszeichen '+' ein.

2. Beispiel:

```
Haupt_Kopf{           // leitet den Abschnitt ein  
  
[%]+Programmnummer   // schreibt % und Programmnr.  
  
}                     // muß immer am Ende eines  
                     // Abschnitts stehen
```

Abschnitt:

```
Unter_Kopf{
...
...
...
}
```

Art: Abschnitt (globale Definition)

Erläuterung: **Unter_Kopf** leitet einen Abschnitt ein, indem die Zeilen definiert werden, die ein Unterprogramm enthält vor der ersten Programmzeile. Meistens werden hier die Programmnummer und ein Zeichen angegeben.

Beispiel:

```
Unter_Kopf{           // leitet den Abschnitt ein
[%MM]                // schreibt %MM in die erste Zeile
[N]+Programmnummer   // schreibt z.B. N9002
}                     // muß immer am Ende eines
                     // Abschnitts stehen
```

'Normaler' Text wird in die eckigen Klammern geschrieben und dort, wo die Programmnummer stehen soll geben Sie Programmnummer ein. Steht die Programmnummer nach einer Klammer, so geben Sie einfach noch das Pluszeichen '+' ein.

2. Beispiel:

```
Unter_Kopf{           // leitet den Abschnitt ein

[%]+Programmnummer    // schreibt % und Programmnr.

}                     // muß immer am Ende eines
                     // Abschnitts stehen
```

Abschnitt:

```
Befehle{
...
...
...
}
```

Art: Abschnitt (globale Definition)

Erläuterung: **Befehle** leitet einen Abschnitt ein, indem die Befehle definiert werden, die CNC-Steuerung unterstützt, d.h. z.B. die G & M Funktionen.

```
Beispiel:      Befehle{                                // leitet den Abschnitt ein

PROGRAMM_STOP=M00 // definiert M00

GERADEN =G01;X;Y;Z    // definiert G1
...

}                                // muß immer am Ende eines
                                // Abschnitts stehen
```

Abschnitt:

```
Zyklen{  
...  
...  
...  
}
```

Art:

Abschnitt (globale Definition)

Erläuterung:

Zyklen leitet einen Abschnitt ein, indem die Zyklen definiert werden, die Ihre CNC-Steuerung unterstützt. Zyklen sind komplexe Fertigungsrouninen, die die CNC-Maschine selbständig ausführt, z.B. Tieflochbohrzyklus.

Beispiel:

```
Zyklen{                                // leitet den Abschnitt ein  
  
ZYKLUS_AUS=G80 // definiert wie ein Zyklus abgeschaltet  
...            // wird.  
  
}              // muß immer am Ende eines  
              // Abschnitts stehen
```

Befehl: Nummerierung_0 = ...

Art: Befehl, nur im Abschnitt 'Global' gültig.

Erläuterung: **Nummerierung_0** dient zur Definition des ersten Teils der Nummerierung in Ihrer CNC-Steuerung.

Die Nummerierung besteht im 'Normalfall' auf einem Buchstaben und einer Zahl, z.B.: N1 oder Zeile10 usw.

Mit Nummerierung_0 können Sie angeben, was der Zahl angegeben werden soll.

Wird der Befehl nicht angegeben, so werden nur Zahlen bei der Nummerierung angegeben (wenn Nummerierung_1 definiert ist)! Wird Nummerierung_1 auch nicht definiert, so wird überhaupt keine Nummerierung ausgegeben.

```
Beispiel:      Global{                                // leitet den Abschnitt ein
...
Nummerierung_0    =  N  // ergibt N...
...
}                // muß immer am Ende eines
                // Abschnitts stehen
```

```
2. Beispiel:      Global{                                // leitet den Abschnitt ein

...
Nummerierung_0   =   Zeile// ergibt Zeile...
...

}                // muß immer am Ende eines
                // Abschnitts stehen
```

Befehl: Nummerierung_1 = ...

Art: Befehl, nur im Abschnitt 'Global' gültig.

Erläuterung: **Nummerierung_1** dient zur Definition des zweiten Teils der Nummerierung in Ihrer CNC-Steuerung.

Die Nummerierung besteht im 'Normalfall' auf einem Buchstaben und einer Zahl, z.B.: N1 oder Zeile10 usw.
Mit Nummerierung_1 können Sie angeben, um wieviel von Zeile zu Zeile die Zahl addiert wird.

```
Beispiel:      Global{                                // leitet den Abschnitt ein
...
Nummerierung_1  =  1  // ergibt ...1  ... , ...2 ...  , usw.
...
}                // muß immer am Ende eines
                // Abschnitts stehen
```

```
2. Beispiel:      Global{                                // leitet den Abschnitt ein

...
Nummerierung_1   =  10 // ergibt ...10   ... , ...20 ...   , usw.
...

}                // muß immer am Ende eines
                // Abschnitts stehen
```


Befehl: **Kommentar** = ...

Art: Befehl, nur im Abschnitt 'Global' gültig.

Erläuterung: **Kommentar** gibt an, wie in Ihrer Steuerung der Kommentar gekennzeichnet wird.
In 'Normalfall' wird der Kommentar in runden Klammern angegeben.

Beispiel:	Global{	// leitet den Abschnitt ein
	...	
	Kommentar = (;	// Kommentar in runden Klammern
	...	
	}	// muß immer am Ende eines // Abschnitts stehen

[illegible]

Befehl: **Vorschub** = ...

Art: Befehl, nur im Abschnitt 'Global' gültig.

Erläuterung: **Vorschub** gibt an, wie in Ihrer Steuerung der Vorschub angegeben wird.

In 'Normalfall' wird der Vorschub mit **F** angegeben.

Beispiel:	Global{	// leitet den Abschnitt ein
	...	
	Vorschub = F	// Vorschub -> F
	...	
	}	// muß immer am Ende eines // Abschnitts stehen

Befehl: Drehzahl = ...

Art: Befehl, nur im Abschnitt 'Global' gültig.

Erläuterung: **Drehzahl** gibt an, wie in Ihrer Steuerung die Drehzahl angegeben wird.

In 'Normalfall' wird die Drehzahl mit **S** angegeben.

```

Beispiel:      Global{                               // leitet den Abschnitt ein
...
Drehzahl      =  S      // Drehzahl -> S
...
}                // muß immer am Ende eines
                  // Abschnitts stehen

```

Befehl: **Werkzeuge** = ...

Art: Befehl, nur im Abschnitt 'Global' gültig.

Erläuterung: **Werkzeuge** gibt an, wie in Ihrer Steuerung die Werkzeuge angegeben werden.

In 'Normalfall' werden die Werkzeuge mit **T** angegeben.

```
Beispiel:      Global{                               // leitet den Abschnitt ein
...
Werkzeuge    =   T           // Werkzeuge -> T
...
}                  // muß immer am Ende eines
                   // Abschnitts stehen
```

Befehl: **Programm_Stop** = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Programm_Stop** gibt an, wie in Ihrer Steuerung der Programm-
Stop angegeben wird.

In 'Normalfall' mit **M0** .

Beispiel: Befehle{ // leitet den Abschnitt ein

...
Programm_Stop = M00
...

} // muß immer am Ende eines
 // Abschnitts stehen

Befehl: **Spindel_Rechtslauf= ...**

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Spindel_Rechtslauf** gibt an, wie in Ihrer Steuerung die Spindel mit Drehrichtung rechts anläuft.

In 'Normalfall' mit **M03** .

Beispiel:

```
Befehle{                                // leitet den Abschnitt ein
...
Spindel_RechtsLauf = M03
...
}                                // muß immer am Ende eines
                                // Abschnitts stehen
```

Spindel_Linkslauf= ...

Befehl, nur im Abschnitt 'Befehle' gültig.

Spindel_Linkslauf gibt an, wie in Ihrer Steuerung die Spindel mit Drehrichtung links anluft.

In 'Normalfall' mit **M04**.

```
Befehle{                                // leitet den Abschnitt ein

...
Spindel_LinksLauf    =  M04
...

}                                // muß immer am Ende eines
                                // Abschnitts stehen
```

Befehl: **Spindel_Stop** = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Spindel_Stop** gibt an, wie in Ihrer Steuerung die Spindel gestopt wird.

In 'Normalfall' mit **M05** .

Beispiel:

```
Befehle{                                // leitet den Abschnitt ein
...
Spindel_Stop      =  M05
...
}                                // muß immer am Ende eines
                                // Abschnitts stehen
```


Programm_Ende = ...

Befehl, nur im Abschnitt 'Befehle' gültig.

Programm_Ende gibt an, wie in Ihrer Steuerung das Programmende angegeben wird.

In 'Normalfall' mit **M30** .

```
Befehle{                                // leitet den Abschnitt ein
...
Programm_Ende      =  M30
...

}                                // muß immer am Ende eines
                                // Abschnitts stehen
```

Befehl: **Eilgang** = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Eilgang** gibt an, wie in Ihrer Steuerung der Eilgang programmiert wird.

Hierzu wird die entsprechende Funktion (im 'Normalfall' **G00**) mit den Achsen angegeben. Die Angaben werden mit einem Simikolon von einander getrent:

z.B. G0;X;Y;Z

Nach diesen Angaben können Sie noch definieren, wie bei Ihrer
Steuerung inkremental bemaßt wird (nur wenn die Achsenbe-
zeichnungen von den ersten drei Angaben abweichen):

z.B. G0;X;Y;Z;IX;IY;IZ

Als letztes kann noch bei Bedarf ein zusätzlicher Text angegeben werden, der immer bei der Übersetzung ausgegeben wird. Diese Angabe wird mit einem graden Strich vom restlichen Definitionsteil getrennt:

z.B. G0;X;Y;Z|FMAX

```
Beispiel:      Befehle{                                // leitet den Abschnitt ein
...
Eilgang        =  G00;X;Y;Z|(Vorsicht Craschgefahr)
...
}               // muß immer am Ende eines
                // Abschnitts stehen
```

Befehl: Geraden = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Geraden** gibt an, wie in Ihrer Steuerung mit Vorschub im gradliniger Bewegungsrichtung verfahren wird.

Hierzu wird die entsprechende Funktion (im 'Normalfall' **G01**) mit den Achsen angegeben. Die Angaben werden mit einem Simikolon von einander getrent:

z.B. G1;X;Y;Z

Nach diesen Angaben können Sie noch definieren, wie bei Ihrer
Steuerung inkremental bemaßt wird (nur wenn die Achsenbe-
zeichnungen von den ersten drei Angaben abweichen):

z.B. G1;X;Y;Z;IX;IY;IZ

Als letztes kann noch bei Bedarf ein zusätzlicher Text angegeben werden, der immer bei der Übersetzung ausgegeben wird. Diese Angabe wird mit einem graden Strich vom restlichen Definitionsteil getrennt:

z.B. G1;X;Y;Z|(Fräser im Eingriff)

```
Beispiel:      Befehle{                                // leitet den Abschnitt ein
...
Geraden        =  G01;X;Y;Z
...
}                // muß immer am Ende eines
                  // Abschnitts stehen
```

```
Beispiel:      Befehle{                                // leitet den Abschnitt ein
...
Kreis20        =  G02;X;Y;Z;R
...
}                // muß immer am Ende eines
                  // Abschnitts stehen
```

```

Beispiel:      Befehle{                                // leitet den Abschnitt ein
...
Kreis30          =  G03;X;Y;Z;R
...
}                // muß immer am Ende eines
                  // Abschnitts stehen

```


Befehl: Kreis21 = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Kreis21** gibt an, wie in Ihrer Steuerung ein Kreis(-bogen) im Uhrzeigersinn programmiert wird.

Hierzu wird die entsprechende Funktion (im 'Normalfall' **G02**) mit den Achsen angegeben. Nach den Achsen müssen Sie angeben, wie der Mittelpunkt bemaßt wird.
Die Angaben werden mit einem Simikolon voneinander getrennt:

z.B. G2;X;Y;Z;I;J;K

Nach diesen Angaben können Sie noch definieren, wie bei Ihrer
Steuerung inkremental bemaßt wird (nur wenn die Achsenbe-
zeichnungen von den ersten drei Angaben abweichen):

z.B. G2;X;Y;Z;I;J;K;IX;IY;IZ;II;IJ;IK

Als letztes kann noch bei Bedarf ein zusätzlicher Text angegeben werden, der immer bei der Übersetzung ausgegeben wird. Diese Angabe wird mit einem graden Strich vom restlichen Definitionsteil getrennt:

z.B. G2;X;Y;Z;I;J;K;IX;IY;IZ;II;IJ;IK|(Fräsen)

```
Beispiel:      Befehle{                                // leitet den Abschnitt ein
...
Kreis21        =  G02;X;Y;Z;I;K;J
...
}                // muß immer am Ende eines
                  // Abschnitts stehen
```

Befehl: Kreis31 = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Kreis31** gibt an, wie in Ihrer Steuerung ein Kreis(-bogen) gegen den Uhrzeigersinn programmiert wird.

Hierzu wird die entsprechende Funktion (im 'Normalfall' **G03**) mit den Achsen angegeben. Nach den Achsen müssen Sie angeben, wie der Mittelpunkt bemaßt wird.
Die Angaben werden mit einem Simikolon voneinander getrennt:

z.B. G3;X;Y;Z;I;J;K

Nach diesen Angaben können Sie noch definieren, wie bei Ihrer
Steuerung inkremental bemaßt wird (nur wenn die Achsenbe-
zeichnungen von den ersten drei Angaben abweichen):

z.B. G3;X;Y;Z;I;J;K;IX;IY;IZ;II;IJ;IK

Als letztes kann noch bei Bedarf ein zusätzlicher Text angegeben werden, der immer bei der Übersetzung ausgegeben wird. Diese Angabe wird mit einem graden Strich vom restlichen Definitionsteil getrennt:

z.B. G3;X;Y;Z;I;J;K;IX;IY;IZ;II;IJ;IK|(Fräsen)

```
Beispiel:      Befehle{                                // leitet den Abschnitt ein
...
Kreis31        =  G03;X;Y;Z;I;K;J
...
}                // muß immer am Ende eines
                  // Abschnitts stehen
```


Befehl: EbeneXZ = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: EbeneXZ gibt an, wie in Ihrer Steuerung die XZ-Ebene als Bearbeitungsebene angegeben wird.

Im 'Normalfall' mit **G18**.

```
Beispiel:      Befehle{                                // leitet den Abschnitt ein
...
EbeneXZ      =  G18
...

}              // muß immer am Ende eines
               // Abschnitts stehen
```

Befehl: EbeneZY = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: EbeneZY gibt an, wie in Ihrer Steuerung die ZY-Ebene als Bearbeitungsebene angegeben wird.

Im 'Normalfall' mit **G19** .

Beispiel:

```
Befehle{                               // leitet den Abschnitt ein
...
EbeneZY    = G19
...
}                               // muß immer am Ende eines
                               // Abschnitts stehen
```

Befehl: **Korrektur_Aus** = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Korrektur_Aus** gibt an, wie in Ihrer Steuerung die Radiuskorrektur abgeschaltet wird.

Im 'Normalfall' mit **G40** .

```
Beispiel:      Befehle{                               // leitet den Abschnitt ein
...
Korrektur_Aus   =   G40
...

}               // muß immer am Ende eines
                // Abschnitts stehen
```

Korrektur_Links = ...

Befehl, nur im Abschnitt 'Befehle' gültig.

Korrektur_Links gibt an, wie in Ihrer Steuerung die Radiuskorrektur 'links' eingeschaltet wird.

Im 'Normalfall' mit **G41**.

```
Befehle{                                // leitet den Abschnitt ein

...
Korrektur_Links      =  G41
...

}                                // muß immer am Ende eines
                                // Abschnitts stehen
```

Korrektur_Rechts = ...

Befehl, nur im Abschnitt 'Befehle' gültig.

Korrektur_Rechts gibt an, wie in Ihrer Steuerung die Radiuskorrektur 'rechts' eingeschaltet wird.

Im 'Normalfall' mit **G42**.

```
Befehle{                                // leitet den Abschnitt ein

...
Korrektur_Rechts    =   G42
...

}                                // muß immer am Ende eines
                                // Abschnitts stehen
```


Korrektur_Über = ...

Befehl, nur im Abschnitt 'Befehle' gültig.

Korrektur_Über gibt an, wie in Ihrer Steuerung die Radiuskorrektur 'über' eingeschaltet wird.

Im 'Normalfall' mit **G44**.

```
Befehle{                                // leitet den Abschnitt ein

...
Korrektur_Über      =  G44
...

}                                // muß immer am Ende eines
                                // Abschnitts stehen
```

Befehl: **Maß_Zoll** = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Maß_Zoll** gibt an, wie in Ihrer Steuerung das
Maßsystem auf Zollmaß eingestellt wird.

Im 'Normalfall' mit **G70**.

Beispiel: Befehle{ // leitet den Abschnitt ein

 ...

 Maß_Zoll = G70

 ...

 } // muß immer am Ende eines
 // Abschnitts stehen

Befehl: **Maß_mm** = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Maß_mm** gibt an, wie in Ihrer Steuerung das
Maßsystem auf metrisch eingestellt wird.

Im 'Normalfall' mit **G71**.

Beispiel: Befehle{ // leitet den Abschnitt ein

 ...

 Maß_mm = G71

 ...

 } // muß immer am Ende eines
 // Abschnitts stehen

Befehl: **Bezugmaß** = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Bezugmaß** gibt an, wie in Ihrer Steuerung die Absolutprogrammierung eingestellt wird.

Im 'Normalfall' mit **G90**.

```
Beispiel:      Befehle{                               // leitet den Abschnitt ein
...
Bezugmass      =   G90
...
}               // muß immer am Ende eines
                // Abschnitts stehen
```

Befehl: Kettenmaß = ...

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **Kettenmaß** gibt an, wie in Ihrer Steuerung die Inkrementalprogrammierung eingestellt wird.

Im 'Normalfall' mit **G91**.

```
Beispiel:      Befehle{                                // leitet den Abschnitt ein
...
Kettenmaß      =  G91
...
}                // muß immer am Ende eines
                  // Abschnitts stehen
```

Befehl: **DarstellungsFenster_1= ...**

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **DarstellungsFenster_1** definiert das Darstellungsfenster nach Philips-Steuerungen.

Das Darstellungsfenster wird immer so angegeben, wie die Zeichnung auf dem Bildschirm zur Zeit der Übersetzung dargestellt wird.

Beispiel:

```
Befehle{                                // leitet den Abschnitt ein

...
DarstellungsFenster_1    = G98;X;Y;Z;I;J;K
...

}                                // muß immer am Ende eines
                                // Abschnitts stehen
```


Befehl: **DarstellungsFenster_2= ...**

Art: Befehl, nur im Abschnitt 'Befehle' gültig.

Erläuterung: **DarstellungsFenster_2** definiert das Darstellungsfenster nach Heidenhain-Steuerungen.

Das Darstellungsfenster wird immer so angegeben, wie die Zeichnung auf dem Bildschirm zur Zeit der Übersetzung dargestellt wird.

Beispiel:

```
Befehle{                                // leitet den Abschnitt ein

...
DarstellungsFenster_2    = G30;X;Y;Z|G17
...

}                                // muß immer am Ende eines
                                // Abschnitts stehen
```

Rohteil_2= ...

Befehl, nur im Abschnitt 'Befehle' gültig.

Rohteil_2 definiert das Werkstück nach Heidenhain-Steuerungen.

Das Werkstück wird immer so angegeben, wie die Zeichnung auf dem Bildschirm zur Zeit der Übersetzung dargestellt wird.

[illegible]