

2 Aufbau eines Part-Programms

2.1 Part-Programmwort

2.1.1. Wortstruktur

Der elementare Baustein eines Part-Programms ist das sog. Wort (Programmbefehl). Jedes Wort besteht aus einer Adressenbuchstabe und ein- oder mehrfacher Zahl des Adresseninhaltes, ggf. mit Vorzeichen und Dezimalpunkt.

Beispiel:

X 002500
Adresse
Einführungs-Nullen
Zahlenwert

2.1.2 Schreibweise des Zahleninhaltes

- Beim Schreiben des Wertes von Hilfsfunktionen können die führenden Nullen vor der ersten bedeutungstragenden Zahl entfallen. Z.B. F 1.4 S 35 uä.
- Beim Schreiben der Koordinaten ohne Dezimalpunkt können die führenden Nullen vor der ersten bedeutungstragenden Zahl entfallen, der Wert bedeutet dann die Bahn in mm. Z.B. X 3620 = 3620 mm.
- Beim Schreiben der Koordinaten mit Dezimalpunkt können die führenden Nullen vor der ersten gültigen Zahl vor dem Dezimalpunkt sowie nach der letztgültigen Zahl nach Dezimalpunkt entfallen.

Z.B. X 36.12 = 36.12 mm
X 0.2 = 0.2 mm
X 384. = 384 mm
X .12 = 0.12 mm

- Das Zeichen "-" muß angeführt werden, "+" darf jedoch entfallen.
- Wenn ein Wort keine Nullinformation trägt, muß mindestens eine Null gesetzt werden.
- Zwischen der Adresse und dem Zahlenwert eines Wortes kann (muß aber nicht) ein oder mehrere Leerzeichen gesetzt werden. Ein Zahlenwert, einschl. Vorzeichen, darf jedoch nicht mit einem Leerzeichen unterbrochen werden. Beim Programmierung von Parametern wird der Zahlenwert durch jeweiligen Parameter ersetzt (s. Kapitel mit Beschreibung von Parameterprogrammierung).

Z.B. X36.12 richtig
X 36.12 richtig
X 36.12 richtig
X 36. 12 falsch (Leerzeichen innerhalb eines Zahlenwertes)
X- 36.12 falsch (Leerzeichen nach dem Vorzeichen)

2.1.3 Datenworttypen

Je nach Qualität der Information, die in einzelnen Datenwörtern eines Part-Programms enthalten ist, werden die Wörter und deren Adressen grundsätzlich in zwei Gruppen aufgeteilt, und zwar Datenwörter mit und ohne Maß.

- Datenwörter mit Maß - neben einem Wert geben sie auch ein Maß an (mit "+" und "-" Vorzeichen und mit/ohne Dezimalpunkt). Hierzu gehören folgende Adressen: A,B,C,X,Z, U, V W, I, J, K, F, R, Q (für Verweilzeit)
- Datenwörter ohne Maß - enthalten nur eine Wertangabe. Hierzu gehören die Adressen D, G, M, S, T, P, H, Q (für eine Anzahl von Wiederholungen).

Manche Part-Programm-Datenwörter sind im System dauernd gespeichert, dh. sie sind auch für alle nachfolgenden Blöcke wirksam, bis sie nicht mit einem anderen Wert derselben Adresse überschrieben werden. Die andere Gruppe sind Datenwörter, die nur innerhalb eines Blocks wirksam sind, in dem sie programmiert wurden.

2.2 Part-Programmblock

Einzelne Datenwörter werden in Programmblöcke zusammengefügt. Ein Programmblock stellt elementare Informationseinheit ist, die Angabe der Geometrie und Technologie enthält. Programmblöcke haben verschiedene Längen.

Maximaler Blockformat:

Anm.:

In Klammern steht die max. Zifferanzahl der jeweiligen Adresse. Die Ziffern vor der G- und M-Adresse geben die Anzahl der Gruppen an.

Bedeutung der Adressen und entsprechenden Werten sind in der nachfolgenden Tabelle (Systemwörterbuch) zusammengefaßt. Diese Liste gibt eine komplette Übersicht der Adressen an, auf die CNC836 reagiert, einschl. jeweilige Bereiche.

Adresse	Beschreibung
%	Partprogramm-Nr., Bereich 1 - 999999
N	Partprogrammblock-Nr., Bereich 1 - 999999
G	Vorbereitungsfunktion 00 bis 99, unterteilt in 11 Gruppen
M	Technologische Hilfsfunktionen, Bereich 00 -99, unterteilt in 14 Gruppen
X, Y, Z, U, V, A, B, C	Bezeichnung der Koordinaten im Bereich -69999,99mm bis +69999,999mm, höchstens sechs Koordinaten aus der angeführten Anzahl werden unter Maschinenkonstanten 00 bis 05 eingegeben.
I, J, K	Adresse für Eingabe des Kreis-Mittelpunktes, I auch für Eingabe der Winkelverschiebung für mehrgängige Gewinde, K für Drehautomaten mit X-X-Koordinatensystem vorgesehen
D	Korrekturnummer-Adresse, Bereich 1 - 99
R	Wert des Koordinaten- oder Funktionsparameters
Q	Anzahl der Makrozykluswiederholungen Anzahl der Sprünge im Part-Programm oder die Verweilzeit
F	Verfahren in mm/min oder mm/U im Bereich 0.001 . 24,999, beim Gewindeschneiden mittels G33 Gewindesteigung von max. 99.999
L	Makrozyklus-, Unterprogramm- oder Blocknummer, in den im Part-Programm gesprungen wird, Bereich 1 - 9999
P	Hilfsfunktion, Bereich 1 - 99
&	Programmierte Werkzeuglänge-Korrektur, eine Vier-Dekaden-Funktion, zulässige Ziffern 0, 1, 2
S	Spindeldrehzahlen, Bereich 1 - 9999U/min
T	Werkzeugnummer, Bereich 1 - 699999

Übersicht der verwendeten G-Funktionen je nach Gruppen

Gruppe	Funktion	Bedeutung
0	G00	Eilgang
	G01	Linearinterpolation
	G02	Kreisinterpolation CW
	G03	Kreisinterpolation CCW
	G30	Fahrt in den Bezugspunkt aus dem Programm
	G33	Gewindeschneiden
1	G17	Wahl der XY-Ebene (nur für die Werkzeugradius-Korrektur)
	G18	Wahl der ZX-Ebene
	G19	Wahl der YZ-Ebene
	G14	Wahl der Z4-Ebene
	G15	Wahl der Y4-Ebene
	G16	Wahl der 4X-Ebene
2	G23	Kontinuierliche Blockverknüpfung
	G24	Beschleunigte Blockverknüpfung
	G05	Kopieren nach Vorlage
	G06	Kopieren aus dem Speicher (reserviert)
	G07	Kopieren in den Speicher (reserviert)
	G08	Schleppen einer/zwei Achsen nach den Steuerachsen
	G09	Wendel (Spirale, Schleppen nach Winkel)
	G10	Winkeleingabe für die Schleppachse in der Wendel
	G98	Grundzustand der 2. Gruppe, Ausschaltung aller G-Funktionen
	3	G40
G41		Radiuskorrektur links
G42		Radiuskorrektur rechts
4	G26	Parameterarithmetik gem. R5
	G27	Parameterarithmetik gem R5 u. R6
	G28	Parameterarithmetik gem. R5, R6 u. R7
	G29	Parameterarithmetik gem. R5, R6, R7 u. R8
5	G53	Nullpunktverschiebung 0
	G54	Nullpunktverschiebung 1
	G55	Nullpunktverschiebung 2
	G56	Nullpunktverschiebung 3
	G57	Nullpunktverschiebung 4
	G58	Nullpunktverschiebung 5
6	G59	Nullpunktverschiebung 6
	G94	Vorschub in mm/min ohne konstante Schneidgeschwindigkeit
	G95	Vorschub in mm/U ohne konstante Schneidgeschwindigkeit
	G96	Konstante Schneidgeschwindigkeit (KRR) mit Vorschub mm/min
7	G97	Konstante Schneidgeschwindigkeit (KRR) mit Vorschub mm/U
	G70	Makrozyklus- oder Programmende
	G71	Unterprogramm-Aufrufung
	G72	Unterprogramm-Aufrufung
	G73	Programmsprung
8	G79	Eintritt ins Makrozyklus/Unterprogramm
	G80	Löschung des Festzyklus
9	G81-G89	Festzyklen gem. ISO-Norm oder vom Benutzer programmierte Festzyklen
	G90	Absolutwertprogrammierung
10	G91	Inkrementalwertprogrammierung (Schrittmaßprogrammierung)
	G04	Verweilzeit nach Maßgabe der Adresse Q
	G92	Eingabe der Koordinatensystem-Anfangspunkte in die Tabelle
	G93	Eingabe der Koordinatensystem-Anfangspunkte in die Tab.

Übersicht der verwendeten M-Funktionen je nach Gruppen:

Gruppe	Funktion	Bedeutung
0	Reserve	
1	M00	Programmstopp
	M01	Wahlweiser Stopp
	M02	Partprogramm-Ende
	M30	Partprogramm-Ende
2	M03	Spindelstart CW
	M04	Spindelstart CCW
	M05	Spindelstopp
	M19	Spindelstopp im orientierten Punkt
		Anm.: In die 2. Gruppe kann man weitere 16 M-Funktionen eingeben, die in Maschinenkonstanten R160-R163 definiert sind (z.B. M13 = M3 + M07 ...SpindelSTART und Kühlung)
3	M41	Spindeldrehzahlen - Bereich 1
	M42	Spindeldrehzahlen - Bereich 2
	M43	Spindeldrehzahlen - Bereich 3
	M44	Spindeldrehzahlen - Bereich 4
4	M36	Direkte Vorschubprogrammierung
	M37	Vorschubprogrammierung 1:100
5	M07	Kühlung 2 ein
	M08	Kühlung 1 ein
	M09	Kühlung 1 u. 2 aus
	M17	Kühlung 1 und 2 ein
6	M50	Kühlung 3 ein
	M51	Kühlung 4 ein
	M52	Kühlung 4 ein
	M53	Kühlung 3 und 4 aus
7	M10	Werkstückeinspannung
	M11	Werkstücklockerung
8	M49	manuelle Überbrückung der FEED OVERRIDE-Funktion
	M48	Löschung der manuellen Überbrückung der FEED OVERRIDE-Funktion
9	M06	Werkzeugaustausch
	M60	Werkzeugaustausch
10	...	Funktionen der Maschinenkonstanten (R56)
11	...	Funktionen der Maschinenkonstanten (R57)
12	...	Funktionen der Maschinenkonstanten (R58)
13	...	Funktionen der Maschinenkonstanten (R59)
14	...	Übrige Hilfsfunktionen, die in den vorstehenden Gruppen nicht angeführt wurden, Relaisausgaben in BCD-Kode

Erklärung der Symbole in der Tabelle:

x - Hilfsfunktionen, wirksam nur im Block, wo sie programmiert wurden

+ - Nr. der Funktion, die die in den jeweiligen Gruppen angeführte Nummern aufheben und die einstweilig bei MOO- und MO1-Funktionen in M2-, M5-, und M6-Gruppen und dauernd auch bei MO2-, M30 und nach Systemnullung auftreten.

2.2.1 Weitere Hinweise für Erstellung eines Programmblocks

- Nach Adressen 4 oder 5 kann Buchstabe A,B,C,U,V,W (nach Absprache mit dem Maschinenhersteller) programmiert werden.
- Nach Adresse 6 kann Buchstabe A,B,C (nach Absprache mit dem Maschinenhersteller) programmiert werden.

- KOMMENTAR ist eine Kette von beliebigen Zeichen (außer Anführungszeichen), die zwischen den Anführungszeichen geschlossen ist. Falls nach dem Kommentar ein Zeilenende-Zeichen (CR,LF) steht, können Anführungszeichen entfallen.

Beispiel:

```
N10 X100 Y100 "DIES IST DER KOMMENTAR"
N20 X200 Y200 "DIES IST DER KOMMENTAR
" "
"
N30 X100 "DIES IST DER KOMMENTAR" Y200 Z300
N40 X200 "DIES IST DER KOMMENTAR Y200 Z300
```

Anm. Im N40-Block schließt der Kommentar auch Y200 Z300 in sich, da das zweite Anführungszeichen nicht gesetzt ist und das Zeilenende als Kommentarende verstanden wird!

- Jeder Programmblock muß mit einer N-Adresse (Block-Nr.) begonnen und mit N-Zeichen (Nr. des nachfolgenden Programmblocks) oder mit "*" - Zeichen beendet werden. Übrige Blockwörter können in beliebiger Reihenfolge programmiert werden. Es empfiehlt sich jedoch, Standard-Wortfolge zu benutzen (s. maximaler Blockformat). Falls ein Part-Programm vom System gelocht oder geschrieben wird, werden die Wörter in einem Block in diese Standardfolge angeordnet.
- Das System arbeitet mit einer variablen Blocklänge. Das bedeutet, daß jeder Block beliebige Anzahl der Wörter enthalten kann. (Auch ein leerer Block, in dem nur die Block-Nr. programmiert ist, kann gespeichert werden, was jedoch in der Praxis nicht vorkommt). Jede Adresse (Wort) kann in einem Block nur einmal auftreten, die Ausnahme davon bilden die Gruppenfunktionen (M,G), in denen Werte aus jeder Gruppe und R-Adresse stehen kann, die dann mehrmals in dem Block verwendet werden können (höchstens 32x).
- Ein Programmblock kann beliebige Anzahl von Zeichen enthalten (einschl. Trennzeichen u. Kommentar). Auf einer Zeile (zwischen CR u. LF) können jedoch max. 80 Zeichen stehen.
- Anstatt des Adressenwertes können in allen Adressen (außer N-Adressen) R-Parameter-Nr. geschrieben werden - s. Abschnitt über Parameterprogrammierung.
- Für Trennen einzelner Blockwörter kann beliebige Anzahl der Zwischenräume und Tabulatoren gesetzt werden. Die Zwischenräume und Tabulatoren können auch innerhalb eines Wortes gesetzt werden, jedoch nur zwischen der Adresse und der Nummer (nie innerhalb eines Zahlenwertes!).
- Auf beliebiger Stelle den Programmblockzeichen (zwischen einzelnen Wörtern auch innerhalb eines Wortes) kann ein beliebiges weggelassenes BL-, RS-, CR- oder DEL-Hilfszeichen gesetzt werden, das vom System nicht berücksichtigt wird. Nacheinander können höchstens 10 BL-Zeichen stehen.
- DEL-Zeichen bewirkt Nichtberücksichtigung eines Blocks von Anfang (Adresse N) bis zu diesem Zeichen. Nach ESC-Zeichen muß der geänderte oder nachstehende neue Block in die N-Adresse geschrieben werden.
- "/"-Zeichen (Schrägstrich) ist die einzige Adresse, wo kein Wert steht (die nachstehende Ziffer wird nicht ausgewertet). Bei Anwählen des "/"-Modus (Blockunterdrückung) wird der mit "/"-Zeichen angeführte Block bei Abarbeitung eines Part-Programmes nicht berücksichtigt.
- Wenn in einem Block eine Verweilzeit (GO4-Funktion) zusammen mit einer Funktion der G7-Gruppe vorgegeben ist, hat die Q-Funktion Priorität für die Verweilzeit. In diesem Fall werden die Unterprogramme, Makrozyklen, bzw. Sprünge nur einmal ausgeführt (gleich wie bei Q=1).

2.3 Aufbau der Part-Programme, Makrozyklen und Festzyklen

Das Part-Programm für ein Teil besteht in der Regel aus dem eigentlichen Part-Programm und aus einem Teil der sog. Unterprogramme. Die Unterprogramme beziehen sich lediglich auf das Part-Programm, und sie müssen unmittelbar nach seinem Ende geschrieben (gelocht) werden. Ein Part-Programm muß nicht unbedingt Unterprogramme enthalten.

2.3.1 Part-Programmanfang

Am Anfang eines Part-Programmes muss immer Part-Programm-Nr. stehen. Es wird mit %-Adresse und einer positiven Zahl (höchstens 6-Dekaden, anders als Null) ohne Dezimalpunkt eingeführt werden, Diese Zahl identifiziert das jeweilige Part-Programm innerhalb eines Systems.

Beispiel:

(1 ist identisch mit 000001)

Falls während des Part-Programmeinlesens aus einem Eingabegerät im READ-Modus kein %-Zeichen auftritt, werden die Zeichen nicht berücksichtigt, d.h. vor dem %-Zeichen kann ein beliebiges Zeichen außer Null stehen.

Wenn sich z.B. auf einem Lochstreifen mehrere Part-Programme befinden und man will nur eines davon einlesen, es genügt - bei Anwendung eines Lochstreifen-Lesegerätes - den Lochstreifen beliebig vor dem Anfang des gewünschten Part-Programms einzulegen.

Man kann die Nummer des einzulesenden Part-Programms im 1. Parameter der READ-Modus anwählen. In diesem Fall wird das gewünschte Programm auf dem Lochstreifen nach dessen Nummer gesucht. Wenn im Laufe des Einleseprozesses mehr als 10 Leerzeichen nach einem Zeichen (außer Null) eingelesen werden, wird Fehlermeldung "Streifenende" angezeigt. Dies bedeutet, daß das Streifen zu Ende gelaufen ist, ohne das Part-Programm gefunden zu haben. Mit Restart kann man das Einleseprozeß fortsetzen (z.B. wenn es sich mehr als 10 Leerzeichen zwischen einzelnen auf dem Lochstreifen befindlichen Programmen stehen). Wenn mehrere Part-Programme auf einmal eingelesen werden, muss jeder von ihnen mit "*" -Zeichen abgeschlossen werden oder es müssen mindestens 10 Leerzeichen nach dem nicht abgeschlossenen Programm stehen. In diesem Fall wird jedoch Fehlermeldung "Streifenende" angezeigt.

2.3.2 Part-Programmende

Das logische Ende eines Part-Programms wird mit Programmierung von M02- oder M30-Funktionen bestimmt, und zwar in dem letzten Part-Programmblock. Nach dem Block mit M02- oder M30-Funktion können in dem Part-Programm nur Blöcke einzelner Unterprogramme stehen, die aus der Struktur des Part-Programms "abgerufen werden".

Das Ende des gesamten Part-Programms, einschl. möglicher Unterprogramme, muss mit "*" -Zeichen (Stern) abgeschlossen werden.

2.3.3. Unterprogramm (PPRG)

Das Unterprogramm bedeutet eine Gruppe von Programmblöcken, die mit einem Standardblock beginnen und abgeschlossen sind. In logischer Hinsicht wird ein Unterprogramm nur zum jeweiligen Part-Programm zugeordnet und kann nur aus diesem Part-Programm abgerufen werden.

Jedes Unterprogramm muß unbedingt mit dem Block beginnen, in dem neben Block-Nr. nur G79-Funktion und L-Adresse mit dem Wert angeführt ist, der die Unterprogramm-Nr. festlegt.

Die Blocknummer hat für die Identifizierung eines Unterprogramms keine Bedeutung. Der Einführungsblock ist wie folgt strukturiert:

N.... G79 Lxxxx

Unterprogramm-Nr.

Unterprogrammanfang

Blocknummer

Das Unterprogramm muss mit dem Block abgeschlossen werden, in dem G-70-Funktion angeführt ist. In dem letzten Unterprogrammblock können auch andere Funktionen, bzw. Bewegungen programmiert werden. Der letzte Block ist wie folgt strukturiert:

N.... G70

Unterprogrammende

Beim Einlesen des Part-Programms müssen jeweilige Unterprogramme zusammen mit dem Part-Programm eingelesen werden, dessen Teil sie sind (unmittelbar nach dem Block, wo M02- und M30-Funktionen programmiert werden).

Insgesamt können 99 Unterprogramme programmiert werden. In dem jeweiligen Part-Programm, das mit "%" beginnt und mit "*" beendet wird (eigentliche Part-Programm mit eventuellen Unterprogrammen), darf nicht mehr als ein Programmblock mit derselben Blocknummer (Wert der N-Adresse) bezeichnet werden. Das Programm kann in jedem Block eines Part-Programms abgerufen werden. Struktur der Aufrufanweisung:

```
Nxxxx .... G71 Lxxxx Qxxxx
                Anzahl der Unterprogramm-Wiederholungen
                Nr. des abgerufenen Unterprogramms
                Aufrufbefehl
```

Die Anzahl der in der Q-Funktion programmierten Unterprogramm-Wiederholungen muß nicht angeführt werden. In diesem Fall wird das Unterprogramm nur einmal abgearbeitet (gleich wie bei Q1-Programmierung). Nach Abarbeitung des Unterprogramms bleiben die Werte der Technologie- und Hilfsfunktionen in solchem Zustand, wie sie von dem Unterprogramm eingestellt wurden. Das Unterprogramm kann in demselben Umfang wie das eigentliche Part-Programm editiert werden. In einem Programmblock des Part-Programms, aus dem das Unterprogramm abgerufen wird, kann man auch andere Vorschub- und Technologiefunktionen programmieren. Der Sprung zum gewählten Unterprogramm erfolgt erst am Ende des Programmblocks, d.h. nach Durchführung anderer programmierter Operationen. Nach Abarbeitung des Unterprogramms kehrt die Steuerung zurück zum Part-Programmblock, der nach dem Block folgt, aus dem gesprungen wurde.

2.3.4 Beispiele des Part-Programms mit Unterprogrammen

```
% 35          " PART-PROGRAMM NR. 35-ANFANG "
N10 X0 Y0 Z0 G54 " ERSTER PART-PROGRAMMBLOCK "
N20 X200 G1 F300 " ZWEITER PART-PROGRAMMBLOCK "
....
....
N80 G71 L100 Q2 " AUFRUFEN DES PART-PROGRAMMS NR. 100"
....          " UNTERPROGRAMM WIRD ZWEIMAL ABGERUFEN "
....
....
N120 G71 L200  " AUFRUFEN DES PART-PROGRAMMS NR. 200"
....
....
N310 X0 Y0 Z0 G00 " GELÄUFIGER PART-PROGRAMMBLOCK "
N320 M30         " LOG. PART-PROGRAMMSENDE "
"
" ES FOLGEN ZWEI UNTERPROGRAMME
"
" UNTERPROGRAMM NR.100 "
"
N500 G79 L100   " ERSTER BLOCK DES UNTERPROGRAMMS NR. 100 "
N510 ...
N520 ...
....
....
N590 G70       " LETZTER BLOCK DES UNTERPROGRAMMS NR.100"
"
```

```

" UNTERPROGRAMM NR.200"
"
N600 G79 L200      " ERSTER BLOCK DES UNTERPROGRAMMES NR. 200"
N610 ...
N620 ...
....
....
N700 G70          " LETZTER BLOCK DES UNTERPROGRAMMES NR.200"
*                    " STERN - PART-PROGRAMM-ENDE "

```

2.3.5 Makrozyklus (MRC)

Makrozyklus bezeichnet eine Gruppe von Programmblöcken, die ein Part-Programm für ein typisches Drehteil oder für ein auf einer konkreten Maschine zu produzierendes Teil bilden.

Am Anfang und am Ende eines jeden Makrozyklus muss ein Block stehen, der die gleiche Form als der Anfangs- und Abschlußblock des Unterprogramms hat. Allgemein bezieht sich ein Makrozyklus zu allen gespeicherten Part-Programmen, aus denen er abgerufen werden kann (er kann aus Part-Programmen als auch aus deren Unterprogrammen abgerufen werden).

Die Makrozyklen gehören zur Gruppe der sog. Bibliothek-Part-programmen und werden in das System individuell eingelesen. In das System dürfen nicht zwei oder mehrere Makrozyklen mit derselben Nummer eingelesen werden. Die Makrozyklen werden oft für Aktionen verwendet, die allen Part-Programmen gemeinsam sind, wie z.B. Anfahren der Werkzeugaustausch-Position und der eigentliche Werkzeugaustausch. Es empfiehlt sich auch Programmierung von Parametern.

Ein Makrozyklus-Aufruf kann im beliebigen Unterprogrammblock erfolgen.
Der Aufruf hat folgendes Format:

```

Nxxxx .... G72 Lxxxx Qxxxx
      Anzahl der Makrozyklus-Wiederholungen
      Nr. des abgerufenen Makrozyklus
      Aufrufbefehl

```

Der Makrozyklus kann auch aus beliebigem Block eines anderen Makrozyklus (sog. Makrozyklus-Einbettung) abgerufen werden. Nach Beendigung eines Makrozyklus bleiben die Werte der Technologie- und Hilfsfunktionen in dem Zustand erhalten, wie sie in dem Makrozyklus eingestellt wurden.

In einem Part-Programm- oder auch Makrozyklusblock (d.h. auch in dem Unterprogramm), aus dem ein anderes Makrozyklus abgerufen wird, können auch andere Vorschub- und Technologiefunktionen programmiert werden. Der Sprung zum gewählten Makrozyklus erfolgt - ähnlich wie beim Aufruf eines Unterprogramms - erst am Ende des Programmblocks. Jeder Makrozyklus muss auf die gleiche Weise wie ein Unterprogramm abgeschlossen werden:

```

Nxxxx G70
      Makrozyklus-Ende

```

2.3.6 Sprünge innerhalb eines Part-Programms und Makrozyklus

Mit Hilfe der G73-Funktion kann man einen Sprung in anderen Programmblock bestimmen. Format des Blocks in dem ein Sprung programmiert ist:

Nxxxx G73 Lxxxx Qxxxx
Anzahl der Sprungwiederholungen
Nummer des Blocks, in den gesprungen wird
Sprungbefehl

Die Sprünge dürfen nur in die Blöcke innerhalb des sog. Programmkomplexes, d.h. im Rahmen eines Part-Programms, Unterprogramms oder Makrozyklus nach vorne und nach hinten (d.h. in Richtung zum Anfang oder Ende des Part-Programms, Unterprogramms oder Makrozyklus) programmiert werden. Es können also keine Sprünge aus dem Part-Programm ins Unterprogramm ausgeführt werden.

Der Wert der Q-Funktion gibt die Anzahl der Sprungwiederholungen an. Bei Q (n) wird der Programmabschnitt zwischen jeweiligen Programmblöcken (n+1)mal abgearbeitet.

Anmerkung:

L-Hilfsfunktionen (Nr. des aufgerufenen Unterprogramms, Makrozyklus sowie Block-Nr. in den der Sprung erfolgen soll) Q (Anzahl der Unterprogramm- oder Makrozyklus-Wiederholungen oder Anzahl der Sprünge bei SPRB) bezieht sich nur zum Block, in dem sie programmiert wurden.

Beispiel:

N25

N26

N27

.....

.....

N54 G73 L26 Q3

N55 „Fortsetzung auf N55

In dem Programmblock, wo ein Sprung programmiert wurde, können auch andere Vorschub- und Technologiefunktionen programmiert werden. Der Sprung in den definierten Programmblock wird erst am Ende des Blocks ausgeführt.