

Inhaltsverzeichnis zur Anleitung zum TOOL1

APPEND	Seite 1-2
CHANGE	Seite 1-4
DELETE	Seite 1-9
ERASE	Seite 1-11
FIND	Seite 1-13
KEEP	Seite 1-16
LINK	Seite 1-17
PLAST	Seite 1-18
PLIST	Seite 1-19
PROGRAM	Seite 1-20
RENUMBER	Seite 1-21
SPLIT	Seite 1-26
Liste der Instruktionen	Seite 1-27

APPEND

```
*****  
*  
* Eröffnet ein neues Programm-Modul.  
*  
*****
```

Syntax



- Hinweise:
- Es kann jeweils nur ein neues Programm-Modul eröffnet werden, d.h. ein weiteres Modul kann erst dann eröffnet werden, wenn das vorangehende Modul mindestens eine Zeile enthält.
 - Es können nur BASIC - Programm-Module verwaltet werden.
 - Das neu eröffnete Programm-Modul ist sofort aktives Modul.

Besondere Fehlermeldung: - ERROR 22: Zur Eröffnung eines neuen Moduls steht kein Speicherplatz mehr zur Verfügung.

* Bemerkung: Beachten Sie bitte im Anhang die Erläuterungen zur modularen *
* Programmierung (Seite A-6). *
*

APPEND (Beispiel)

Ein bestehendes Modul im Speicher kann um ein weiteres Modul erweitert werden:

Ausgangssituation:

MODUL 1 befindet sich im Speicher:

```
10:"A"REM MODUL 1
20:INPUT "D = ";D
30:J=D:H$=""
40:H=INT (J/16)
50:J=J-16*H
60:H$=MID$ ("0123
456789ABCDEF",
J+1, 1)+H$
70:J=H
80:IF JTHEN 40
90:PRINT "D = &";
H$
100:END
```

Nach der Befehlsfolge

APPEND (enter)

10"B"REM MODUL 2 (enter)

20PRINT "TIME = ";TIME (enter)

30END (enter)

befindet sich folgendes im Speicher:

```
10:"A"REM MODUL 1
20:INPUT "D = ";D
30:J=D:H$=""
40:H=INT (J/16)
50:J=J-16*H
60:H$=MID$ ("0123
456789ABCDEF",
J+1, 1)+H$
70:J=H
80:IF JTHEN 40
90:PRINT "D = &";
H$
100:END
10:"B"REM MODUL 2
20:PRINT "TIME =
";TIME
30:END
```


CHANGE (Fortsetzung)

- [Anzahl]: Mit diesem Parameter kann angegeben werden, wie oft [AUSDRUCK 1] durch [AUSDRUCK 2] ersetzt werden soll.
[Anzahl] muss kleiner 256 sein.
Wird * angegeben, wird so oft als möglich ausgetauscht.
Wird nichts angegeben, wird einmal ausgetauscht.
- [1. Zeile] oder [1. Markenname]: Mit diesem Parameter kann angegeben werden, ab welcher Zeile oder ab welchem Markennamen ausgetauscht werden soll.
Wird nichts angegeben, beginnt der Austausch bei der ersten Zeile des aktiven Programm-Moduls.
- [letzte Zeile] oder [letzter Markenname]: Mit diesem Parameter kann angegeben werden, bis zu welcher Zeile oder bis zu welchem Markennamen ausgetauscht werden soll.
Wird nichts angegeben, endet der Austausch bei der letzten Zeile des aktiven Programm-Moduls, oder wenn die mit [Anzahl] begrenzte Menge Austausche erreicht ist.

Hinweis: Das BASIC-Schlüsselwort REM kann nur ausgetauscht werden, indem man einer Textvariablen den Wert CHR\$ &F1 + CHR\$ &AB zuordnet und diese Textvariable anstelle von [AUSDRUCK 1] oder [AUSDRUCK 2] angibt.

ACHTUNG: ZUM LÖSCHEN EINER GANZEN ZEILE MUSS DELETE VERWENDET WERDEN (siehe Seite 1-9) !

- Besondere Fehlermeldungen:
- ERROR 17: Die Begrenzungen von [AUSDRUCK 1] und [AUSDRUCK 2] sind nicht von der gleichen Art (siehe Syntax-Diagramm).
 - ERROR 67 IN xx: Zeile xx würde nach dem Austausch die maximal zulässige Länge übersteigen.
 - ERROR 177: Das BASIC-Programm ist nicht in Ordnung.
 - ERROR 178 IN xx: Beim Austausch in Zeile xx würden Daten überschrieben, d.h. es steht für den Austausch nicht mehr genügend Speicherplatz zur Verfügung.

- Einschränkungen:
- Die numerischen Ausdrücke [Anzahl], [1. Zeile] und [letzte Zeile] dürfen nur Grundrechenoperationen enthalten. Klammersausdrücke sind verboten !
 - Anführungszeichen (") können nicht ausgetauscht werden.

CHANGE (Beispiele)

Ausgangssituation:

Es befinden sich die drei
nebenstehenden Module im Speicher:

```
100:"A"REM MODUL 1
110:PAUSE TIME
120:"Z"PRINT TIME
130:END
100:"B"REM MODUL 2
110:PRINT "LOGONX"
120:PRINT "X = ";X
130:REM Y=LOG X
140:PRINT "LOG = "
      ;Y
150:END
100:"C"REM MODUL 3
110:PRINT "LOG = "
      ;LOG Y
120:END
```

Beispiele von CHANGE-Anweisungen:

Instruktion

```
PROGRAM 2 (enter)
CHANGE !PRINT! TO !INPUT!;,120 (enter)
A$=CHR$ &F1 + CHR$ &AB (enter)
CHANGE A$ TO !!;,110, (enter)
CHANGE !PRINT! TO !PAUSE!;* (enter)
CHANGE "0" TO "**XYZ**";2 (enter)
PROGRAM 1 (enter)
CHANGE !TIME! TO "PC-1500" (enter)
CHANGE !TIME! TO !ABS SIN (X-Y)! (enter)
```

Wirkung

Programm-Modul 2 ist nun aktives Modul.
Ersetzt in Zeile 120 PRINT durch INPUT maximal einmal.
Weist A\$ das TOKEN für das BASIC-Schlüsselwort REM zu (siehe Seite A-4). Löscht ab Zeile 110 bis zum Modulende REM maximal einmal.
Ersetzt PRINT im ganzen Modul durch PAUSE so oft als möglich.
Ersetzt im gesamten Modul "0" durch "**XYZ**" maximal zwei mal.
Programm-Modul 1 ist nun aktives Modul.
ERROR 17: [AUSDRUCK 1] ist durch Anführungszeichen (") begrenzt, [AUSDRUCK 2] dagegen durch Ausrufezeichen (!).
Ersetzt im gesamten Modul TIME durch ABS SIN (X-Y) maximal einmal.

Nach diesen Anweisungen befindet sich folgendes im Speicher:

```
100:"A"REM MODUL 1
110:PAUSE ABS SIN
      (X-Y)
120:"Z"PRINT TIME
130:END
100:"B"REM MODUL 2
110:PAUSE "L**XYZ*
      *G**XYZ**NX"
120:INPUT "X = ";X
130:Y=LOG X
140:PAUSE "LOG = "
      ;Y
150:END
100:"C"REM MODUL 3
110:PRINT "LOG = "
      ;LOG Y
120:END
```