Label im Basic 7.0

Nun wird es auch mit Basic 7.0 möglich, Label anstatt feststehender Zeilennummern anzugeben. Dies erleichtert die Programmierung.

Obwohl das Basic 7.0 des C128 sehr umfangreich ist, fehlt ihm eine Möglichkeit, die einige andere Computer schon aufweisen: die Angabe von Sprungzielen durch Label.

Eine große Schwierigkeit beim Erstellen umfangreicher Programme scheint es zu sein, daß man von Anfang an wissen muß, wohin ein Sprung führen soll. Auch die Zusammenstellung von Programmen durch Module ist nicht einfach, wenn Sprungadressen auftreten.

Mit diesem Programm (Listing 1) kann neben der bisherigen Möglichkeit, Sprungziele mit Zeilennummern anzugeben, folgendes programmiert werden:

1) GOTO und GOSUB auf eine errechnete Zeile.

```
A=1000:GOTO A
A=1000:GOSUB A/2+50
```

Damit lassen sich Sprungverteiler vermeiden.

2) GOTO und GOSUB auf Inhalte von Feldelementen (numerisch). In einem numerischen Feld sind Sprungziele gespeichert. Hiermit ist möglich: GOTO (oder GOSUB) A(I) mit I aus zulässigem Bereich.

```
name : sys                1300 13b7
-------------------------------------
1300 : ad 00 ff 48 a9 00 8d 00  87
1308 : ff ad 06 d5 09 06 8d 06  1d
1310 : d5 68 8d 00 ff 78 a9 22  2b
1318 : 8d 08 03 a9 13 8d 09 03  67
1320 : 58 60 20 80 03 c9 89 f0  47
1328 : 0a c9 8d f0 24 4c f3 4a  a1
1330 : ea ea ea 20 f7 77 a5 0f  3e
1338 : 30 23 20 15 88 a5 17 c9  4a
1340 : fa b0 06 20 e2 59 4c f6  30
1348 : 4a a2 0b 2c a2 11 4c 3c  88
1350 : 4d 20 1d 5a 20 86 03 20  c2
1358 : 33 13 4c f6 4a 20 7e 87  b5
1360 : f0 ea 18 69 05 90 03 4c  72
1368 : ed a5 85 93 8a e9 04 85  0f
1370 : 5c 98 e9 00 85 5d a5 2d  c7
1378 : a6 2e 85 61 86 62 a0 01  c3
1380 : 20 ec 42 f0 c7 aa a0 04  21
1388 : 20 ec 42 c9 8f f0 14 a0  fa
1390 : 00 20 ec 42 4c 7a 13 20  49
1398 : e2 42 85 a6 20 ec 42 c5  cf
13a0 : a6 d0 ec c8 c4 93 d0 ef  0e
13a8 : 18 a0 01 20 94 50 86 62  ff
13b0 : 38 20 06 5a 4c f6 4a 18  9b
```

Listing 1. Bitte mit dem MSE im C64-Modus eingeben.

3) GOTO und GOSUB auf Marken. Die Marke muß in einer REM-Zeile stehen. Es ist folgendes möglich:

```
...
...
550 GOTO"DRUCK"
...
...
... REMDRUCK
... PRINT"...
```

Gleiches gilt auch für GOSUB.

4) GOTO und GOSUB auf den Inhalt von Feldelementen, die Strings enthalten. Verfahren wie unter Punkt 2).

5) GOTO und GOSUB auf definierte Stringvariable. Dabei muß stets eine REM-Zeile mit dem entsprechenden String vorhanden sein.

```
10 A$="1000"
20 GOSUB A$
```

Eingabehinweise

Das Programm (Listing 1) geben sie bitte mit dem MSE im C64-Modus ein und speichern es. Für Floppy-Besitzer scheint weiter folgender Weg der einfachste zu sein:

1) Erstellen einer Autoboot-Diskette

Name des Basic-Programms: INIT

2) Unter INIT wird folgender Einzeiler gespeichert:

```
10 POKE 46,32:POKE 48,32:POKEDEC("2000"),0:CLR:
BLOAD"SYS":SYSDEC("1300"):NEW
```

Dieses Programm bewirkt folgendes:

a) Basic-Anfang auf $2000 einstellen
b) Variablen-Anfang auf $2000 einstellen
c) Der Befehl CLR bewirkt ein Anpassen der übrigen Zeiger auf den umgestellten Bereich
d) Nachladen des unter SYS gespeicherten Maschinenprogrammes
e) Einbinden in den Interpreter
f) Löschen

Statt f) könnte auch ein weiteres Programm nachgeladen und gestartet werden.

3) Das Maschinenprogramm ist dabei unter dem Namen SYS gespeichert.

Nicht-Floppy-Besitzer müssen folgenden Einzeiler laden und starten:

```
10 POKE 46,32:POKE 48,32: POKEDEC("2000"),0:CLR:NEW
```

Danach muß das Maschinenprogramm geladen und mit SYSDEC("1300") gestartet werden.

Die Listings 2 und 3 sollen Ihnen nur die Fähigkeiten der Erweiterung demonstrieren, müssen also nicht unbedingt abgetippt werden.

Um das Maschinenprogramm kurz zu halten, mußten folgende Restriktionen vereinbart werden:

1) Die erweiterte GOTO/GOSUB-Routine ist nur im Programm-Modus benutzbar

```
..:GOTO..
```

2) Soll nach THEN auf eine Marke gesprungen werden, muß die Syntax geändert werden

```
IF..THEN:GOTO..
```

3) Soll ELSE mit Marken verwendet werden, ist ebenfalls die Syntax zu ändern

```
IF..:ELSE:GOTO..
```

(Gerhard Seltzsam/dm)

```
10 REM   *** DEMO 1 ***                <A99>
15 PRINT CHR$(147)                     <PA3>
20 FOR I = 1 TO 10                     <I0V>
30 GOSUB "ZAHL"                        <510>
40 GOSUB "QUADRAT"                     <TUJ>
50 GOSUB "WURZEL"                      <QOP>
60 NEXT                                <8P0>
70 END                                 <7SU>
100 REM ZAHL                           <IIA>
110 PRINT I,                           <3H7>
120 RETURN                             <4S7>
300 REM WURZEL                         <JIE>
310 PRINT SQR(I)                       <JJV>
320 RETURN                             <4K7>
510 REM QUADRAT                        <GQ3>
520 PRINT I*I,                         <VAF>
530 RETURN                             <KE7>
```

Listing 2. So können Sie mit Label arbeiten

```
10 REM  *** DEMO 2 ***                                  <A99>
20 PRINT CHR$(147)                                      <1A3>
30 Z$="ZAHL": Q$="QUADRAT": W$="WURZEL"                 <LR3>
40 FOR I = 1 TO 10                                      <K0V>
50 GOSUB Z$: GOSUB Q$                                   <402>
70 GOSUB W$                                             <1CT>
80 NEXT                                                 <GU0>
90 END                                                  <RVE>
100 REM ZAHL                                            <IIA>
110 PRINT I,                                            <3H7>
120 RETURN                                              <4S7>
300 REM WURZEL                                          <JIE>
310 PRINT SQR(I)                                        <JJV>
320 RETURN                                              <4K7>
510 REM QUADRAT                                         <GQ3>
520 PRINT I*I,                                          <VAF>
530 RETURN                                              <KE7>
```

Listing 3. Auch hier wurden Zeilensprünge durch Label ersetzt