

# Delete

Diese in Maschinensprache geschriebene Routine ermöglicht es, Basic-Programmzeilen in einem vorzuziehenden Zeilennummernbereich zu löschen.

Ein ärgerlicher Nachteil des C 64-Basic und auch von Simons-Basic ist das Fehlen einer Delete-Routine zum schnellen Löschen mehrerer Programmzeilen. Das folgende Programm hilft dem ab.

Der Basic-Lader speichert das Maschinenprogramm im Kassettenpuffer ab Adresse 828 (dez.). Das Programm kann dann auf drei verschiedene Arten aufgerufen werden:

- SYS 828, ZN - ZN Bereich löschen
- SYS 828, - ZN Bis Zeile alles löschen
- SYS 828, ZN - Ab Zeile alles löschen

Es werden jeweils die Zeilen inklusive der angegebenen gelöscht. Da die Routine im Kassettenpuffer untergebracht ist, belegt sie keinen Basic-Speicherplatz.

(Hans-Herbert Hagedorn / ev)

```

10 REM *****
15 REM *
20 REM * DELETE *
25 REM *
30 REM * H. H. HAGEDORN *
35 REM *
40 REM * RUPPRECHTSTR. 30 *
45 REM *
50 REM * 83 LANDSHUT *
55 REM *
60 REM * TEL. 0871/67337 *
65 REM *
70 REM *****
75 :
80 FOR I=828 TO 990 : READ A : POKE I,A
85 S=S+A : NEXT
90 IF S <> 17132 THEN PRINT "DATENFEHLER" : END
    <250>

95 PRINT "OK"
100 DATA 032,253,174,032,121,000,144,006
105 DATA 240,004,201,171,208,023,032,107
110 DATA 169,032,019,166,165,095,133,025
115 DATA 165,096,133,026,032,121,000,240
120 DATA 004,201,171,240,005,162,011,076
125 DATA 058,164,032,115,000,032,107,169
130 DATA 208,243,165,020,005,021,208,006
135 DATA 169,255,133,020,133,021,208,006
140 DATA 230,020,208,002,230,021,032,019
145 DATA 166,165,095,133,036,165,096,133
150 DATA 037,056,165,036,229,025,165,037
155 DATA 229,026,144,201,165,045,229,036
160 DATA 133,095,165,046,229,037,133,096
165 DATA 024,165,025,101,095,133,045,165
170 DATA 026,101,096,133,046,160,000,177
175 DATA 036,145,025,230,025,208,002,230
180 DATA 026,230,036,208,002,230,037,056
185 DATA 165,095,233,001,133,095,165,096
190 DATA 233,000,133,096,016,225,032,089
195 DATA 166,032,051,165,076,145,227,000
200 DATA 000,000,000
    <083>

```

# Commodore-Basic erweitert

Mit dem hier vorgestellten Maschinenprogramm wird der Basic-Befehlssatz des VC 20 oder des C 64 um sechs Befehle erweitert.

Die 6 Befehle lauten, in die Basic-Schreibweise übersetzt, GOTO N, GOSUB N, RESTORE N, READ D,A, READ N,D,A, und POP. Einen kleinen Nachteil muß man dabei allerdings in Kauf nehmen, denn diese Routinen kann man nur dem SYS-Befehl ansprechen. Es ist also nicht möglich, eine der Routinen direkt mit einem Basic-Befehlswort aufzurufen.

Der Zugriff auf diese Befehle kann insbesondere dann von großem Nutzen sein, wenn man Programme von anderen Computern umschreiben will, die diese Befehle benutzen.

Will man einen der neuen Befehle in einem Basic-Programm benutzen, muß man nur das Basic-Wort in der oben aufgeführten Liste durch ein »SYS (Adresse)« ersetzen. Die Parameter hinter dem Befehl werden genauso hinter den SYS-Befehl geschrieben, als ob sie hinter dem Basic-Befehl stehen würden.

Bei den nun folgenden Erläuterungen wird davon ausgegangen, daß sich das Maschinenprogramm im Kassettenpuffer ab Adresse 828 befindet. Würde eine andere Anfangsadresse gewählt, ändern sich auch die Adressen der einzelnen Routinen.

Der Befehl GOTO N sieht in der Form, wie er im Programm verwendet werden muß, so aus: SYS(828)N; also doch noch recht einfach. Dieser Befehl bewirkt, daß man direkt zu einer beliebigen Zeile springen kann, deren Zeilennummer »N« vorher berechnet wurde. Nun zur Syntax. Bei diesem, wie auch bei allen folgenden Befehlen ist darauf zu achten, daß die Startadresse der Routine nach dem SYS-Befehl, (hier 828) in Klammern steht, um Sie von der darauf folgenden Parameterliste zu trennen und so als Adresse kenntlich zu machen. »N« repräsentiert hier, wie auch bei den weiter folgenden Befehlen, eine beliebige gültige numerische Variable, eine Zahl oder einen numerischen Ausdruck. Für »N« ist also beispielsweise auch der Ausdruck »INT(RND(1)\*20)\*10+100« erlaubt. Der Ausdruck muß nur einen Ganzzahlenwert zum Ergebnis haben. Noch zu bemerken ist, daß zwischen der geschlossenen Klammer der Adresse und der ersten Variablen oder dem Ausdruck kein Komma stehen darf. Das Komma wirkt wie bei PRINT oder READ wie ein Trennzeichen. Da dieser Befehl aber nur eine Variable oder einen Ausdruck enthalten darf, würde das zu einem »SYNTAX ERROR« führen. Dies gilt auch bei allen folgenden Befehlen. Ist die berechnete Zeilennummer nicht im Programm enthalten, erfolgt die Fehlermeldung »UNDEF'D STATEMENT ERROR«.

Für den Befehl GOSUB N gilt das gleiche, was auch zu GOTO N gesagt wurde, unter Berücksichtigung der Tatsache, daß es sich hier um einen Unterprogramm-Aufruf handelt. Mit diesem Befehl kann man also zu einer vorher berechneten Unterprogramm-Adresse springen (SYS(834) N).

RESTORE N ermöglicht es, den DATA-Zeiger auf eine bestimmte Zeile zu setzen. SYS(866)100 beispielsweise setzt den DATA-Zeiger auf das erste Datum der Zeile 100. Mit einem anschließenden READ-Befehl kann man dann gezielt auf die