

```

2460 PRINT#15,"B-P";2;I:REM
2470 IF P=1 THEN FT=(64 OR FT):REM
2480 IF P=0 THEN FT=(255 AND FT-64):REM
2490 PRINT#2,CHR$(FT);:REM
2500 RM=1:REM
2510 RETURN

```

```

PUFFER POINTER AUF FILETYP <110>
FILETYP SCHUETZEN <004>
SCHUTZ ZURUECKNEHMEN <990>
GEAENDERTER FILETYP <178>
VERAENDERUNGEN DURCHGEFUEHRT <001>
<101>

```

P-Basic-V2: Autostart mit Rück- wärtsgang

Dieser Autostarter erzeugt nicht nur ein einfaches Lade-Programm für Basic und Assembler-Programme. Es wird zusätzlich die RUN/STOP-Taste gesperrt und ein Kopierschutz installiert. Um ein so behandeltes Programm wieder verändern zu können, kann mit einem Befehl der gesamte Vorgang rückgängig gemacht werden.

Das Programm »P-Basic-V2« läuft auf dem Commodore 64 (ohne Basic-Erweiterung) und der Floppy 1541 (nicht auf der Datasette, da das Programm beim Laden den Kassettenpuffer benutzt). Es ist zirka 260 Bytes lang und liegt im Speicher ab 49152 (\$c000).

Nach dem normalen Laden und Starten des Basic-Programmes, ist das Basic-V2 um drei neue Befehle erweitert:

1. !SAVE"Name", Gerätenummer,1
2. !LOAD"Name", Gerätenummer,1
3. !NEW

Der Hauptbefehl ist !SAVE. Mit ihm kann man ein Basic-Programm oder ein Maschinenprogramm (mit Basic-Startzeile) so abspeichern, daß es nach dem Laden von allein startet. Außerdem wird die RUN/STOP/RESTORE-Taste abgestellt. Das Programm ist somit vor Änderungen gesichert. Auch das Kopieren ist nur mit besonderen Kopierprogrammen möglich. Doch durch eine Codewortabfrage in den ersten Zeilen wird es wohl keiner schaffen, das Programm zum Laufen zu bringen. Ein Programm, das mit dieser Basic-Hilfe abgespeichert wurde, muß mit dem Zusatz »,1« absolut, nicht ab Basic-Start (2049), geladen werden. Lädt man ein solches Programm ohne diesen kleinen Zusatz in den Speicher, so läßt es sich weder listen, noch starten. Weil ein normales Programm mit der Zwei-

Byte-Startadresse am Anfang der Datei gespeichert wird, ist es durch ein nochmaliges anderes Abspeichern ganz verloren.

Man kann diese Routine natürlich auch nur zur Bequemlichkeit benutzen. Um die Stopp-Routine wieder zu aktivieren, hilft: POKE808,237 als erster Befehl des Programms.

Ist in dem geschützten Programm nun aber ein Fehler, oder will man es erweitern oder verbessern, so hilft der zweite Befehl. Mit !LOAD kann man alle Programme, die mit Autostart geschützt sind, knacken und nach eigener Lust bearbeiten.

Tritt bei den Arbeiten mit diesen Befehlen ein Fehler auf, den Sie oder das Betriebssystem (Floppy/Computer) verursacht haben, so können sie mit Eingabe von !NEW alle wichtigen Vektoren korrigieren und somit das gerade bearbeitete Programm wieder listen lassen.

Erläuterungen zum Assemblerlisting

In dem Programmabschnitt »Vektorsetzung« wird die Routine zur Erkennung der Befehls-Token (Basic-Befehle) auf die erweiterte Routine (Erkennung) umgelenkt und danach der Name der Basic-Erweiterung ausgegeben.

In der Erkennungsroutine werden die Befehls-Token mit dem neuen Befehl (!) und danach mit den eigentlichen Befehlen (SAVE, LOAD, NEW) verglichen. »SAVE« hat das Basic-Token 148, »LOAD« hat 147 und »NEW« hat 162. Wird keiner dieser Befehle interpretiert, so wird Syntax Error (Jmp \$af08) angezeigt. Die DATAs werden in der Vektorsetzungsroutine ausgegeben.

Der Befehl »LOAD«, der wie der Befehl »SAVE« nur im Direktmodus verwendet werden darf, setzt zuerst das Programmmodulflag, damit es keine Ausgabe wie »searching for ...« und so weiter gibt. Danach werden die LOADparameter geholt (\$e1d4) und nach LOAD verzweigt (\$fd5). Nachdem das Programm geladen ist, werden alle Vektoren mit der NEW-Routine wieder zurückgesetzt und dann zur END-Routine verzweigt (\$a831).

Der Befehl »SAVE« gleicht am Anfang (bis Ende Zeile 166) dem LOAD-Befehl. In Zeile 168 wird die Steuerroutine in den Kassettenpuffer (Sprite 11) geladen. Dann werden die »Input«- und »Output«-Vektoren auf die Steuerroutine gesetzt und das Programm ab Byte 806 gespeichert. Die Vektoren werden dann mit der NEW-Routine normalisiert.

Die Steuerroutine arbeitet ungefähr so wie die NEW-Routine. Nur bei dieser Routine werden zusätzlich die RUN-Zeiger (\$a659) gesetzt, die Run-Stop/Restore-Funktion abgeschaltet und dann zur Interpreterschleife verzweigt.

Bemerkungen zum Programm

Das Befehlszeichen »!« kann natürlich durch fast jedes Zeichen ersetzt werden, nur wenn das Zeichen ein Token ist, muß nicht der ASCII-Wert, sondern der Tokenwert angegeben werden, da sonst der Befehl nicht erkannt wird. Wenn das Zeichen ein Buchstabe ist, kann derselben Variablen kein Wert zugewiesen werden. Denn auch dabei entsteht ein Fehler.

(Jan Kusch/gk)

```

100 REM***** <181>
101 REM* P-BASIC-V2. * <010>
102 REM* * <073>
103 REM* GESCHRIEBEN 1984 VON * <050>
104 REM* * <075>
105 REM* JAN KUSCH * <163>
106 REM***** <187>
107 FOR X=0 TO 255:READ A:S=S+A <229>
108 POKE 49152+X,A:NEXT <051>
109 IF S=26106 THEN 112 <055>
110 PRINT" [DOWNS]?FEHLER [SPACE]IIN[SPACE]DATAS
[SPACE]:" <211>
111 PRINT"PRUEFSUMME [SPACE]:"S" [SPACE2](26106)"
:END <048>
112 POKE 49152+X,ASC ("!") <182>
113 SYS 49152:NEW <240>
114 DATA 169,17,141,8,3,169,192,141,9,3,169,49,
160,192,76,30,171 <165>
115 DATA 32,115,0,205,0,193,240,3,76,231,167,32,
115,0,201,147,240 <161>
116 DATA 64,201,148,240,88,201,162,240,3,76,8,
175,76,180,192,147,144 <101>
117 DATA 18,32,208,45,194,65,83,73,67,45,214,50,
46,32,32,13,195 <112>
118 DATA 45,54,52,32,38,32,214,195,45,49,53,52,
49,13,195,79,80 <074>
119 DATA 89,82,73,71,72,84,32,40,67,41,9,14,8,0,
169,0,133 <068>
120 DATA 157,32,115,0,32,212,225,169,0,166,43,
164,44,133,10,32,213 <225>
121 DATA 255,134,45,132,46,76,180,192,169,0,133,
157,32,115,0,32,212 <037>
122 DATA 225,162,0,189,218,192,157,64,3,232,201,
255,208,245,169,38,133 <207>
123 DATA 43,169,3,133,44,169,64,141,36,3,141,38,
3,169,3,141,37 <058>
124 DATA 3,141,39,3,32,68,229,32,89,225,169,1,
133,43,169,8,133 <068>
125 DATA 44,169,87,141,36,3,169,241,141,37,3,
141,39,3,169,202,141 <210>
126 DATA 38,3,169,237,141,40,3,169,0,133,198,76,
49,168,169,1,133 <178>
127 DATA 43,169,8,133,44,169,87,141,36,3,169,
241,141,37,3,141,39 <173>
128 DATA 3,169,202,141,38,3,169,225,141,40,3,32,
89,166,76,174,167 <222>
129 DATA 255 <160>

```

```

129 ;##### ERKENNUNG #####
130 ;#####
131 ERKENNUNG JSR $0073 : CMP BEFEHL
132 BEQ ENTSCHEIDUNG
133 JMP $A7E7
134 ;
135 ENTSCHEIDUNG JSR $0073
136 CMP #147 : BEQ LOAD
137 CMP #148 : BEQ SAVE
138 CMP #162 : BEQ NEW 1
139 JMP $AF08
140 ;
141 NEW 1 JMP NEW
142 ;
143 ;
144 ;#####
145 ;##### DATA S #####
146 ;#####
147 DATA S .ASC "[CLEAR, BLACK, RVSON, SPACE]
-BASIC-V2. [SPACE2]"
148 .BYTE 13 : .ASC "[&-64 [SPACE]&[SPACE]V&-1541"
149 .BYTE 13 : .ASC "[COPYRIGHT [SPACE] (C)"
150 .BYTE 9,14,8,0
151 ;
152 ;#####
153 ;##### BEFEHL .LOAD #####
154 ;#####
155 LOAD LDA #0 : STA $9D
156 JSR $0073 : JSR $E1D4
157 LDA #00 : LDX $2B : LDY $2C
158 STA $0A : JSR $FFD5
159 STX $2D : STY $2E
160 JMP NEW
161 ;
162 ;#####
163 ;##### BEFEHL .SAVE #####
164 ;#####
165 SAVE LDA #00
166 STA $9D : JSR $0073 : JSR $E1D4
167 ;
168 LDX #00
169 VERSCHIEBUNG LDA ROUTINE,X
170 STA $0340,X : INX
171 CMP #$FF : BNE VERSCHIEBUNG
172 ;
173 LDA #<806 : STA $2B
174 LDA #>806 : STA $2C
175 LDA #<832 : STA $04 : STA 806
176 LDA #>832 : STA 805 : STA 807
177 ;
178 JSR $E544 : JSR $E159
179 ;
180 NEW LDA #1 : STA $2B
181 LDA #8 : STA $2C
182 LDA #$57 : STA 804
183 LDA #$F1 : STA 805 : STA 807
184 LDA #$CA : STA 806
185 LDA #237 : STA 808
186 LDA #0 : STA 198 : JMP $A831
187 ;
188 ;
189 ;##### STEUERROUTINE #####
190 ;
191 ROUTINE LDA #1 : STA $2B
192 LDA #8 : STA $2C
193 LDA #$57 : STA 804
194 LDA #$F1 : STA 805 : STA 807
195 LDA #$CA : STA 806
196 LDA #225 : STA 808
197 ;
198 JSR $A659 : JMP $A7AE
199 ;
200 .BYTE $FF
201 ;
202 BEFEHL .BYTE "!"
203 ;

```



```

100 SYS 9*4096 : .OPT 00
101 *=49152
102 ;
103 ;
104 ;
105 ;#####
106 ;* * * P - BASIC - V2. * * *
107 ;#####
108 ;
109 ; *****
110 ;*GESCHRIEBEN1984V ON*
111 ;**
112 ;*JANKUSCH(J-KSOFT)*
113 ;**
114 ; *****
115 ;
116 ;
117 ;#####
118 ;##### VEK TO RSETZUNG #####
119 ;#####
120 ;
121 LDA #<ERKENNUNG : STA $0308
122 LDA #>ERKENNUNG : STA $0309
123 ;
124 LDA #<DATA S : LDY #>DATA S
125 JMP $AB1E
126 ;
127 ;
128 ;#####

```