

Auto-Save

Wenn Ihnen beim Abtippen eines Programms schon mal der Strom ausgefallen ist, dann ist das folgende Programm genau das richtige für Sie.

Die abgedruckten Listings in den Fachzeitschriften sind ja eine willkommene Programmquelle. Allerdings ist das Abtippen der oft sehr langen Programme eine sehr mühselige Angelegenheit. Besonders groß wird der Ärger, wenn ein Stromausfall beziehungsweise ein Hardware-Defekt stundenlange Tipparbeit zunichte machen. Abhilfe schafft hier nur das umständliche Zwischenspeichern in gewissen Abständen, doch das wird oft vergessen oder aus Bequemlichkeit unterlassen.

Das hier vorgestellte Programm nimmt nun dem geplagten Computer-Benutzer diese Arbeit ab. Es speichert automatisch das bisher Getippte in Intervallen oder auch auf Knopfdruck ab. Bitte tippen Sie vor der Benutzung Listing 1 mit dem MSE und Listing 2 mit dem Checksummer ab. Vergessen Sie das Speichern nicht!

Hinweise zur Benutzung:

Geladen wird das Programm mit LOAD "AUTO-SAVE",8 und anschließend mit RUN. Der Maschinensprachteil wird nachgeladen und dann der Name des einzutippenden Programms erfragt. Keine beziehungsweise Eingabe von Leerzeichen wird abgefangen.

Bei der Eingabe des Speicher-Intervalls ist folgendes zu beachten: Die Intervalle werden nicht durch die Zeit, sondern durch die Speicherbelegung gesteuert. Dies geschieht durch eine Abfrage der Speicherstelle 46. Diese enthält das High-Byte des Zeigers auf den Variablen-Anfang (=Programmende). Gibt man also als Intervall zum Beispiel 4 ein, so wird immer dann, wenn 1 KByte Programmtext neu im Speicher stehen, zwischengespeichert. Um außerdem besonders schwierige Programmteile, zum Beispiel Bildschirmmasken oder mathematische Formeln, sofort zu speichern, gibt es die Möglichkeit, per Knopfdruck ein Speichern auszulösen.

Dies geschieht durch Drücken der @-Taste (Klammeraffe) und anschließendem RETURN. Diese Funktion kann mit POKE 49177,ASC("@ taste") leicht auf jede andere Taste verlegt werden.

Beendet werden kann das Programm durch RUN STOP/RESTORE oder einen Reset, neu wird es gestartet mit SYS 49152.

Erläuterungen zum Programm: »SAVE.OBJ«:

Die Gliederung des Maschinenprogramms läßt sich am besten anhand einer Tabelle durchführen:

Speicherbereich	Belegung
hexadezimal	
\$C000-C014	INIT
\$C015-C02A	BEFEHL
\$C02B-C037	AUTO
\$C038-C0A6	SAVE
\$C0A7-C0D2	Textmeldung
\$C0D3	Sollwert
\$C0D4	Intervall
\$C0D5	Namenlänge
\$C0D6	Namenlänge+ 2
\$C0D7-C0D8	"S:"
\$C0D9	Programmname

INIT: Zwei Vektoren werden umgeändert und damit das Maschinenprogramm aktiviert.

BEFEHL: Dieser Teil wird über den Vektor "Neuen Basic-

Befehl ausführen" (dez. 776-777) angesprungen. Er prüft, ob der Klammeraffe beziehungsweise die festgelegte Taste gedrückt wurde. Falls ja, wird zu SAVE verzweigt, ansonsten wieder in die Interpreterschleife.

AUTO: Diese Routine wurde über den "Zeiger auf Tastatur-Dekodiertabelle" (dez. 655-656) eingebunden. Bei jedem Tastendruck wird der Inhalt der Speicherstelle 46 mit dem Sollwert (siehe oben) verglichen und bei Gleichheit wiederum SAVE angesprungen.

SAVE: Der eigentliche Hauptteil des Programms. Er wird von den oben genannten Routinen angesprungen und hat mehrere Aufgaben:

- »Sollwert« um »Intervall« erhöhen
- bisherigen Programmteil löschen
- neuen Programmteil speichern
- Textmeldung ausgeben.

Um den »Replace«-Befehl mittels Klammeraffen, der bekanntlich seine Tücken hat, zu umgehen, wurde der etwas umständlichere Weg gewählt, vor dem Speichern einen »Scratch«-Befehl an das Laufwerk zu senden und somit das alte Programm zu löschen.

Die Meldung »Saving« des Interpreters wird unterdrückt und statt dessen der Text »Programm wird abgespeichert« ausgegeben. Wem diese Meldung nicht gefällt, der kann sie leicht ändern, indem er sie im ASCII-Format in den oben genannten Speicherbereich schreibt. (Manfred Lins/bs)

```

1 REM *****
2 REM *
3 REM *          AUTO - SAVE          *
4 REM *
5 REM *          1985 BY MANFRED LINS  *
6 REM *          REITACKER 1          *
7 REM *          6492 ZUENTERSBACH    *
8 REM *
9 REM *****
10 :
11 :
12 REM MASCHINEN-PROGRAMM NACHLADEN
13 IF R=0 THEN R=1:LOAD"SAVE-OBJ",8,1
14 :
16 PRINT CHR$(147)CHR$(18)" {13SPACE}AUTO -
   SAVE {16SPACE}"
19 PRINT:PRINT:PRINT
21 :
22 REM NAME EINGEBEN
23 PRINT" {2SPACE}PROGRAMMNAMEN (MAX. 16 ZE
   ICHEN) EIN-"
24 PRINT" {2SPACE}GEBEN : "
25 :
26 S=10:Z=6:GOSUB 62
27 IF A$="" OR LEFT$(A$,1)="" THEN 26
28 :
29 REM NAMEN UND LAENGE ABSPEICHERN
30 N$="S:"+LEFT$(A$,16):L=LEN(N$)
31 POKE 49365,L-2:POKE 49366,L
32 :
33 FOR I=1 TO L:POKE 49366+I,ASC(MID$(N$,I
   ,1)):NEXT I
36 PRINT:PRINT
39 REM INTERVALL EINGEBEN
40 PRINT" {2SPACE}ABSPEICHER-INTERVALL (256
   BYTE = 1)"
41 PRINT" {2SPACE}EINGEBEN : "
43 S=13:Z=10:GOSUB 62
46 REM INTERVALL ABSPEICHERN
47 I=VAL(A$):IF I<1 THEN I=1
48 POKE 49363,8+I:POKE 49364,I
51 REM MASCHINENPROGRAMM INITIALISIEREN
52 SYS 49152
53 :
54 REM LADEPROGRAMM LOESCHEN
55 NEW:END
60 REM INTERPROGRAMM STRINGEINGABE
62 PRINT CHR$(144):OPEN 1,0
63 POKE 214,Z:POKE 211,S:SYS 58640
64 INPUT#1,A$
65 CLOSE 1:PRINT CHR$(154):RETURN

```

Listing 2. »AUTO-SAVE«: Sicherheit bei Stromausfall

```
programm : save-obj          c000 c0d3
```

```
c000 : a2 15 a0 c0 8e 08 03 8c bb
c008 : 09 03 a2 2b a0 c0 8e 8f 0a
c010 : 02 8c 90 02 60 20 73 00 92
c018 : c9 40 f0 06 20 79 00 4c 65
c020 : e7 a7 20 73 00 20 38 c0 b5
c028 : 4c 7b a4 a5 2e cd d3 c0 32
c030 : d0 03 20 38 c0 4c 48 eb f8
c038 : a9 01 a2 08 a0 6f 20 ba 87
c040 : ff a9 00 20 bd ff 20 c0 f6
c048 : ff a9 08 20 b1 ff a9 6f c3
c050 : 20 93 ff a2 00 bd d7 c0 5d
c058 : 20 a8 ff e8 8a cd d6 c0 dd
c060 : d0 f3 a9 08 20 ae ff a9 60
c068 : 01 20 c3 ff ad d3 c0 18 17
c070 : 6d d4 c0 8d d3 c0 a2 00 f7
c078 : bd a7 c0 20 d2 ff e8 8a 23
c080 : c9 2c d0 f4 a9 00 a2 08 67
c088 : 20 ba ff a2 d9 a0 c0 ad 5a
c090 : d5 c0 20 bd ff a9 00 85 dd
c098 : 9d a9 2b a6 2d a4 2e 20 9a
c0a0 : d8 ff a9 80 85 9d 60 11 db
c0a8 : 20 20 20 20 20 12 20 50 98
c0b0 : 52 4f 47 52 41 4d 4d 20 ba
c0b8 : 57 49 52 44 20 41 42 47 75
c0c0 : 45 53 50 45 49 43 48 45 c6
c0c8 : 52 54 20 92 20 20 20 20 63
c0d0 : 20 20 11 85 3b 85 2d 4c 23
```

Listing 1.
»SAVE.OBJ«
wird von
Listing 2
»AUTO-SAVE«
nachgeladen.
Bitte
beachten
Sie die
Eingabe-
hinweise
auf Seite 7.

Ziffern und Zeiger auf dem C 64

Digitaluhren sind out. Um dem Trend der Zeit zu folgen, haben wir für Sie eine »Analog-Uhr« auf dem C 64 installiert.

Es sind schon etliche Versionen von Digitaluhren auf dem C 64 veröffentlicht worden. Zur Abwechslung haben wir diesmal eine Interrupt-gesteuerte Analoguhr für Sie. Das Programm blendet ein klassisches Ziffernblatt mit Zeigern in die rechte obere Ecke des Bildschirms ein.

Hinweise zum Eintippen. Geben Sie zuerst das Maschinenprogramm »UHR PRG.« (Listing 1) mit dem MSE ein, und speichern Sie es. Dann tippen Sie »UHR SPRITEGEN« (Listing 2) mit dem Checksummer ab und speichern es eben-

falls. Starten Sie »UHR SPRITEGEN«. Dieses lädt zuerst das Maschinenprogramm und berechnet danach die Sprite-Daten für die Uhr und die Zeiger. Nach etwa vier Minuten werden die Spritedaten inklusive der Interruptroutine auf Diskette gespeichert. Das Programm wird dann mit LOAD »ANALOGUHR«, 8,1 geladen.

Zum Programm. Die Uhr wird durch »SYS 40000,hh,mm,ss« eingeschaltet. Dabei bedeuten hh, mm, ss jeweils Stunden, Minuten und Sekunden mit je zwei Ziffern (Zum Beispiel »SYS 40000,15,05,00« stellt die Uhr auf 15:05:00). Der Bildschirmspeicher liegt jetzt von 35840 bis 36839 und der Basic-Speicher wird auf etwa 33 000 Bytes begrenzt. Durch RUN/STOP-RESTORE wird die Uhr abgeschaltet und der Bildschirm liegt wieder bei 1024. Das Ziffernblatt und die drei Zeiger bestehen aus Sprites, die von der neuen Interrupt-Routine gesteuert werden. Diese Routine liest bei jedem Interrupt die »time-of-day«-Register der CIA 1, berechnet für jeden Zeiger die Bildschirmposition und setzt die Spritpointer. Dann wird zum normalen Interrupt des Betriebssystems gesprungen.

Programmbeschreibung: Das Basic-Programm »UHR SPRITEGEN« besteht hauptsächlich aus drei Schleifen. In den Zeilen 160-260 werden die Zeigersprites erzeugt. Die IF-Zeilen bestimmen die Startkoordinaten der Linien. Durch READ in Zeile 220 werden die Endkoordinaten aus dem ersten Datenblock gelesen. Die aufgerufenen Unterprogramme löschen das zu bearbeitende Sprite (460), ziehen eine Linie im Sprite (370) und übertragen das Sprite in den endgültigen Speicher unter dem Basic-ROM (480). In den Zeilen 270-290 werden die Daten für die vier Sprites, die das Zifferblatt darstellen, gelesen und gePOKEt. Zeile 300 liest die Daten für das Maschinenprogramm, das das Programm »ANALOGUHR« speichert. Dieses wird in Zeile 310 aufgerufen. Das Unterprogramm Zeile 340-360 setzt einen Punkt im bearbeiteten Sprite, in Zeile 380-440 werden die Punkte für eine Linie (X1, Y1, X2, Y2) berechnet und gesetzt. Zeile 460 löscht das Arbeitssprite. In Zeile 480 werden die Spritedaten in den Speicherbereich, entsprechend der Variablen »W«, die in Zeile 240 berechnet wird, übertragen. Die Datenzeilen enthalten die X2,Y2-Koordinaten der Zeiger, die Daten für das Zifferblatt und für die Speicherroutine.

Das Maschinenprogramm »UHR PRG.« arbeitet folgendermaßen: nach SYS 40000 wird zuerst die Uhr in CIA1 gestellt, der Bildschirm verschoben und die Spriteregister des VIC initialisiert. Dann wird der Basic-Speicher begrenzt und der NMI- und IRQ-Vektor verbogen. Anschließend wird zum Basic-Warmstart gesprungen. Die neue NMI-Routine setzt nur den Bildschirm wieder zurück, dann wird zum »Restore« des Betriebssystems gesprungen.

```
programm : uhr prg.          9c40 9e4c
```

```
9c40 : ad 0f dc 29 7f 8d 0f dc 2b
9c48 : ad 0e dc 09 80 8d 0e dc bb
9c50 : a2 03 d0 03 20 73 00 20 e6
9c58 : fd ae 38 e9 30 0a 0a 0a 87
9c60 : 0a 85 fd 20 73 00 38 e9 9c
9c68 : 30 05 fd c9 60 90 03 4c 03
9c70 : 48 b2 e0 03 d0 0d c9 24 8f
9c78 : b0 f5 c9 12 90 05 18 f8 5b
9c80 : 69 68 d8 9d 08 dc ca d0 3b
9c88 : cb 8e 08 dc a9 34 8d 18 db
9c90 : d0 a7 95 8d 00 dd a7 8c fb
9c98 : 8d 88 02 a9 ff dd 15 d0 81
9ca0 : 8d 10 d0 8e 29 d0 a9 01 fd
9ca8 : 8d 28 d0 8d 2a d0 8d 2b e5
9cb0 : d0 8d 2c d0 8d 2d d0 8d 0d
9cb8 : 27 d0 a9 8c 85 34 85 38 c4
9cc0 : 78 a9 21 8d 14 03 a9 9d 42
9cc8 : 8d 15 03 a9 19 8d 18 03 3a
9cd0 : a9 9d 8d 19 03 68 68 58 94
9cd8 : a9 22 8d 06 d0 8d 0a d0 fa
```

```
9ce0 : a9 3a 8d 0c d0 8d 08 d0 c6
9ce8 : a9 38 8d 07 d0 8d 07 d0 31
9cf0 : a9 4d 8d 0b d0 8d 0d d0 54
9cf8 : a9 d4 8d fb 8f a9 d5 8d a7
9d00 : fc 8f a9 d6 8d fd 8f a9 63
9d08 : d7 8d fe 8f a9 93 20 d2 b5
9d10 : ff a9 80 8d 00 a0 4c 74 d5
9d18 : a4 a9 04 8d 88 02 4c 66 da
9d20 : fe 78 ad 0b dc 29 1f 20 fb
9d28 : 30 9e aa 0a 18 69 bc 85 5e
9d30 : fd 8a c9 09 90 0e ad 4a cb
9d38 : 9e 8d 00 d0 ad 4b 9e 8d 82
9d40 : 01 d0 d0 30 c9 06 90 0e 0f
9d48 : ad 48 9e 8d 00 d0 ad 49 42
9d50 : 9e 8d 01 d0 d0 1e c9 03 3a
9d58 : 90 0e ad 46 9e 8d 00 d0 1b
9d60 : ad 47 9e 8d 01 d0 d0 0c fc
9d68 : ad 44 9e 8d 00 d0 ad 45 58
9d70 : 9e 8d 01 d0 ad 0a dc 20 0e
9d78 : 30 9e aa 09 80 8d f9 8f 3f
9d80 : 8a c9 2d 90 15 ad 4a 9e 71
9d88 : 8d 02 d0 ad 4b 9e 8d 03 e6
9d90 : d0 a9 0f 05 fd 8d 8f 8f 65
```

```
9d98 : d0 41 c9 1e 90 15 ad 48 38
9da0 : 9e 8d 02 d0 ad 49 9e 8d 5a
9da8 : 03 d0 a9 01 05 fd 8d f8 06
9db0 : 8f d0 28 c9 0f 90 13 a5 f8
9db8 : fd 8d f8 8f ad 46 9e 8d 4e
9dc0 : 02 d0 ad 47 9e 8d 03 d0 83
9dc8 : d0 11 ad 44 9e 8d 02 d0 15
9dd0 : ad 45 9e 8d 03 d0 a5 fd c2
9dd8 : 8d f8 8f ad 09 dc 20 30 d3
9de0 : 9e aa 09 80 8d fa 8f 8a 2a
9de8 : c9 2d 90 0e ad 4a 9e 8d f0
9df0 : 04 d0 ad 4b 9e 8d 05 d0 3d
9df8 : d0 30 c9 1e 90 0e ad 48 d7
9e00 : 9e 8d 04 d0 ad 49 9e 8d 3b
9e08 : 05 d0 d0 1e c9 0f 90 0e e1
9e10 : ad 46 9e 8d 04 d0 ad 47 46
9e18 : 9e 8d 05 d0 d0 0c ad 44 85
9e20 : 9e 8d 04 d0 ad 45 9e 8d 3b
9e28 : 05 d0 ad 08 dc 4c 31 ea cc
9e30 : aa 4a 4a 4a 4a 4a 8a 29 42
9e38 : 0f c0 00 f0 06 18 69 0a a0
9e40 : 88 d0 fa 60 36 39 36 4d 9c
9e48 : 22 4d 22 39 01 ff 34 01 a3
```

Listing 1. Das Maschinenprogramm »UHR PRG.« Bitte mit dem MSE eingeben.