

Michael Konz

Hochauflösende Grafik mit dem CBM 3022

Für den Benutzer eines Commodore-Rechners bietet sich mit dem Drucker CBM 3022 nicht nur ein leistungsfähiges Gerät für die Wiedergabe von Texten, sondern auch ein recht brauchbarer Plotter mit einer Auflösung von maximal etwa 230 000 Einzelpunkten. Die programmiertechnischen Kunstgriffe, die erforderlich sind, um dem 3022 das Plotten beizubringen, werden im folgenden beschrieben. Am Drucker sind keinerlei Eingriffe notwendig. Es sollte daher keine Schwierigkeiten bereiten, das Programm auch mit dem neueren 4022 zu betreiben.

Der 3022 besitzt die Möglichkeit, je Druckzeile ein frei programmierbares Sonderzeichen, bestehend aus 6×7 Punkten, zu drucken. Die Programmierung dieses Sonderzeichens geschieht mittels der Sekundäradresse 5. Über einen mit dieser Sekundäradresse geöffneten Kanal werden sechs Datenbytes, getriggert von einem CR, übertragen. Diese "Druckvorschrift" bleibt solange gespeichert, bis sie durch neue Daten widerrufen wird. Der Ausdruck eines so programmierten Zeichens erfolgt durch einen von CHR\$(254) über einen normalen Druckkanal (Sekundäradresse 0) getriggert. Bild 1 verdeutlicht noch einmal, wie diese Programmierung erfolgt. Man kann leicht einzusehen, daß durch die geeignete Kombination von Sonderzeichen jeder einzelne Punkt auf der Papieroberfläche angesteuert werden kann. Wenn dies notwendig ist, muß dabei eine Druckzeile mehrfach durchlaufen werden, da pro Zeile nur einmal ein Sonderzeichen definiert werden darf. Die Steuerung dieser Vorgänge geschieht durch ein 2 KByte umfassendes Programm, das im Computer sitzt – und am Drucker. Dabei braucht sich der

Anwender um interne Vorgänge nicht zu kümmern. Nach der Initialisierung des Hilfsprogramms mit dem Befehl SYS 29696 stehen dem Benutzer komfortable Befehle zur Verfügung, die dem Befehlssatz des Tischrechners Olivetti P 6060 entlehnt wurden.

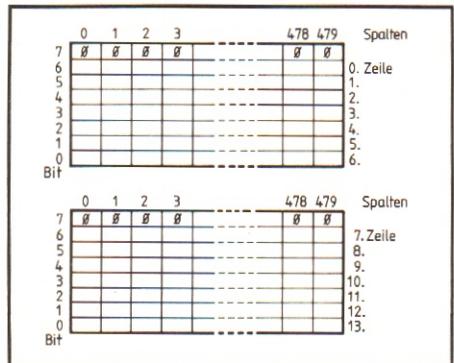


Bild 2. Interne Darstellung der Grafiken im Arbeitsspeicher des Computers CBM 3032

FRAME

Der Befehl FRAME, der unbedingt vor allen anderen Befehlen der Erweiterung stehen muß, legt die Größe der zur Ver-

fügung stehenden Fläche fest. Diese beträgt horizontal 8 Zoll entsprechend der Papierbreite des Druckers. Vertikal ist die Fläche nur durch den zur Verfügung stehenden Speicherplatz begrenzt, denn alle Punkte werden vor dem Ausdruck erst im RAM zwischengespeichert. Der Befehl FRAME reserviert gleichzeitig die benötigten Speicherplätze und sperrt sie für den Basic-Interpreter. Die Syntax des Befehls ist FRAME x, y. Dabei steht x für die horizontale Ausdehnung und ist eine beliebige reelle Zahl zwischen 0 und 8. Bei der vertikalen Ausdehnung y ist die Obergrenze speicherabhängig: Jede Einheit entspricht 60 Punkten. FRAME 7,7 definiert somit eine Fläche von 420×420 Punkten. Den benötigten Speicherraum kann man mit folgender Formel errechnen:

$$M = 60 * X * INT(60 * Y/7 + 0.5)$$
 Das obige Beispiel würde also 25 200 Byte benötigen. Sobald ein neuer FRAME-Befehl gegeben wird, wird die alte Vereinbarung hinfällig, und bereits gespeicherte Daten gehen verloren.

SCALE

SCALE legt nach FRAME die Skalierung der Zeichenfläche fest. Die Syntax für den Befehl ist SCALE xmin, xmax, ymin, ymax. Die Parameter sind beliebige reelle Zahlen, wobei zu beachten ist, daß die Minimalwerte nicht die Maximalwerte übersteigen. Wird die Zeichenfläche beispielsweise mit SCALE -10,10,-10,10 skaliert, so müssen sich die Parameter aller nachfolgenden Zeichenbefehle innerhalb dieser Grenzen bewegen. Der linke Rand der Zeichenfläche erhält dabei den Wert -10 zugeordnet, der rechte entsprechend +10. Analog dazu bezeichnet -10 den unteren Rand und +10 den oberen. Ein erneuter SCALE-Befehl hebt die bisherige Vereinbarung auf, ohne jedoch die bereits erzeugten Daten zu verändern.

CLEAR

Dieser parameterlose Befehl löscht alle bis zu diesem Zeitpunkt erzeugten Daten, ohne jedoch die durch FRAME und SCALE getroffenen Vereinbarungen zu ändern. Gleichzeitig wird der Pointer für die aktuelle Plot-Position in Nullstellung gebracht. Bei dieser Nullstellung handelt es sich um die linke untere Ecke; bei obigem Beispiel hätte dieser Punkt also die Koordinaten -10,-10. Diese Nullstellung wird auch bei FRAME automatisch ausgeführt, so daß

```

10 OPEN 5,4,5
20 PRINT#5,CHR$(2)+CHR$(85)+CHR$(21)+CHR$(85)+CHR$(14)+CHR$(1)
30 CLOSE 5:OPEN 1,4
40 PRINT#1,CHR$(254)
50 CLOSE 1
    
```

Bild 1. Ansteuerung der einzelnen Matrixpunkte beim CBM 3022. Pro Druckzeile ist allerhöchstens ein frei definierbares Zeichen möglich

CLEAR eigentlich nur benötigt wird, wenn eine neue Zeichnung mit den alten Skalierungsvorschriften angefertigt werden soll.

DOT

Mit DOT kann ein einzelner Druckerpunkt direkt angesteuert werden. Die Syntax ist DOT x, y. Die x- und y-Parameter müssen sich dabei innerhalb der Skalierungsgrenzen bewegen.

PLOT

In der Syntax PLOT x, y bewirkt dieser Befehl, daß zwischen der aktuellen Posi-

tion des Plot-Pointers und den angegebenen Koordinaten x, y eine gerade Verbindungslinie gezogen wird.

MOVE

Der Befehl MOVE x, y setzt den Plot-Pointer auf die angegebenen Koordinaten. Es werden keine Linien oder Punkte gezeichnet. Der softwaremäßig erzeugte „Zeichenstift“ wird gleichsam angehoben und über das Papier geführt.

XAXIS

Der Befehl zeichnet eine horizontale Koordinatenachse mit oder ohne ge-

kennzeichneten Einheiten. Die Syntax ist XAXIS y, u, xmin, xmax. Y bezeichnet die Ordinate, auf der die Achse gezeichnet werden soll. Die übrigen Angaben sind optional, d. h. sie können entfallen. Der Rechner setzt dann automatisch vorgegebene Wert für sie ein. Mit dem Parameter u kann festgelegt werden, in welchen Abständen auf der Koordinatenachse Markierungen angebracht werden. Fehlt u, dann wird die Achse ohne Markierung gedruckt, ebenso, wenn u größer als die mit SCALE festgelegte Breite des Zeichenfeldes ist.

xmin und xmax bestimmen, ob die Achse in voller Länge gezeichnet werden

Bild 3. Hex-Dump des CBM-3032-Maschinenprogramms. Es erweitert den Basic-Befehlssatz um Grafik-Befehle für den Drucker 3022

7400	A9	4C	85	70	A9	21	A0	74	7608	02	E6	10	A5	5F	19	E5	63	7810	06	7C	18	90	B6	A5	74	A4	
7408	85	71	84	72	A9	00	85	10	7610	75	85	44	84	45	84	45	A5	40	7818	75	85	44	84	45	A5	40	84
7410	07	89	84	80	78	20	1C	0A	7618	65	00	48	A5	41	65	C1	0D	7820	22	7C	00	07	68	C0	21	7C	
7420	60	86	86	8A	8D	01	91	C9	7620	F0	05	24	68	40	94	77	4C	7828	F0	05	24	68	40	94	77	4C	
7430	00	09	00	01	B1	77	20	F7	7830	16	77	20	62	7A	A2	00	F0	7838	16	77	20	62	7A	A2	00	F0	
7440	02	E6	78	4C	76	00	A5	77	7848	8D	F0	78	95	C1	8D	FC	78	7848	8D	F0	78	95	C1	8D	FC	78	
7450	84	FE	88	F0	2B	38	F1	77	7850	00	02	06	C1	95	C0	16	00	7850	00	02	06	C1	95	C0	16	00	
7460	F4	C9	80	F0	0C	A0	00	E6	7858	00	00	00	94	7C	9D	95	7C	7858	00	00	00	94	7C	9D	95	7C	
7468	FE	E8	80	68	78	10	FA	30	7860	00	00	00	94	7C	9D	95	7C	7860	00	00	00	94	7C	9D	95	7C	
7470	E6	20	03	C8	A5	FE	0A	AA	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7480	8D	92	78	48	4C	70	00	A0	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7488	01	20	03	C8	4C	38	74	4C	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7490	23	D1	85	10	F0	03	20	1D	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7498	7A	20	7D	79	C0	01	30	96	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
74A0	D0	E0	E0	E0	E0	E0	E0	FC	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
74B0	78	8C	FD	7B	20	F8	C0	20	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
74B8	9E	06	7C	8C	07	7C	20	C3	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
74C0	7A	8E	91	7C	C9	06	F0	93	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
74C8	EE	01	7C	DC	AD	FD	7B	AC	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
74D0	02	20	16	DA	AD	FD	7B	AC	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
74D8	7C	78	20	D2	A9	54	A0		7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
74E0	00	20	34	D9	20	93	70	A9	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
74E8	00	18	E5	62	A0	93	74	E5	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
74F0	2A	90	42	86	34	80	05	E4	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7500	10	20	79	C5	20	64	91	0F	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7508	36	79	00	00	00	98	91	0F	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7510	0F	00	02	E6	10	A6	10	E6	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7518	EC	00	F2	A6	0F	E0	00	10	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7520	7C	A9	04	85	D4	20	9F	79	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7528	FF	78	8E	FE	78	D0	01	88	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7530	0A	8E	20	7C	8C	24	55	C3	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7538	C4	C6	4C	23	D1	4C	55	C3	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7540	20	64	79	AD	01	7C	8D	03	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7548	7C	20	96	79	A9	00	8D	02	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7550	7C	A9	04	85	D4	20	9F	79	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7558	50	18	C0	78	D6	2E	02	7C	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7560	20	76	00	F0	10	20	F8	CD	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7568	20	78	D6	E0	04	90	CB	E0	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7570	10	80	C7	86	D4	20	9F	79	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7578	A9	8D	20	B8	79	20	A5	79	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7580	20	9F	79	20	D0	FD	AE	02	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7590	7C	86	62	FA	HC	FD	7B	AD	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7598	C4	D0	F0	8C	FD	7B	AD	FC	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
75A0	78	85	5F	D0	F0	8C	FD	FC	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
75A8	C4	5F	50	05	A5	60	00	06	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
75B0	C4	5F	50	02	A4	5F	84	60	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
75B8	B1	0F	11	0F	D0	08	88	10	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
75C0	F9	20	F2	79	A0	01	B1	0F	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
75D0	30	20	B8	79	C8	C4	63	90	F6	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C
75E0	F0	F4	A9	00	C0	06	30	06	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
75E8	20	B0	79	C8	D0	F6	20	10	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
75F0	79	E6	62	A9	D0	20	B8	79	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
75F8	66	92	00	90	00	FD	00	00	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	
7600	38	A5	0F	65	63	85	0F	50	7868	00	00	00	94	7C	9D	95	7C	7868	00	00	00	94	7C	9D	95	7C	

oder nur ausschnittsweise. Werden diese Angaben fortgelassen, wird die gesamte Achse gedruckt.

PRINT

Das selbe gilt für den Befehl PRINT x, u, ymin, ymax.

DRAW

Die bisher gezeichneten Linien und Punkte wurden lediglich in das RAM geschrieben. Mit dem Befehl DRAW wird die gespeicherte Zeichnung auf den Drucker ausgegeben. Der Parameter bestimmt dabei, um wieviel Druckstellen die Zeichnung eventuell nach rechts verschoben werden soll. Das ist dann sinnvoll, wenn die Zeichnung in der Mitte des Papiers erscheinen soll und die maximale Breite von 8 Zoll gewählt wurde. Wird n weggelassen, beginnt der Ausdruck unmittelbar am linken Papierrand.

Durch mehrmaliges Aufrufen dieses Befehls ist es möglich, eine Zeichnung mehrfach ausgeben zu lassen, ohne die Daten erneut im RAM erzeugen zu müssen.

Graphische Darstellung der Grafiken

Im Speicher wird jeder Punkt von einem Bit repräsentiert. Ist dieses Bit gesetzt, so wird an der entsprechenden Stelle beim Ausdruck ein Punkt gedruckt. Pro Byte werden jedoch nur sieben Bit genutzt, das höchstwertige Bit bleibt unberücksichtigt.

Der Grund dafür liegt in der 6 * 7-Matrix des Druckers. Ein Byte stellt so genau eine Spalte des Druckkopfes dar. Um eine volle Papierbreite von 480 = 480 Spalten zu beschreiben, werden also 480 Byte benötigt.

Beim Ausdruck wird nichts anderes gemacht als Gruppen von jeweils sechs Bytes zusammenzufassen und direkt über die Sekundäradresse 5 an den Drucker weiterzugeben. Anschließend erfolgt der Ausdruck des so programmierten Sonenzeichens, Wagenrücklauf und Neufunktionierung des Druckkopfes auf das nächste Zeichen. Lediglich wenn alle sechs Bytes einer Gruppe den Wert 0 aufweisen, entfällt der Wagenrücklauf, was es wird ein Leerzeichen (CHR\$(32)) ausgegeben.

Dies wiederholt sich solange, bis der gesamte Speicherinhalt zu Papier gebracht wurde. Der Druckvorgang kann übrigens jederzeit mit STOP unterbrochen werden.

Der Interpreter zur Verfügung ste-

```

4 FRAME7,5,7,5
5 OPEN1,0
10 PRINT"GEBEN SIE DIE
    GEWUNSCHETE FUNKTION EIN:
20 PRINT"Y=F(X)=";
30 INPUT#1,A#:PRINT
40 PRINT"DEFFN(X)=";A#
50 PRINT"RUN600";
55 POKE623,13:POKE624,13:POKE158,2:END
60 DEFFN(X)=LIST
70 PRINT"Q"
71 INPUT"X-INTERVALL=";X1,X2
72 IFX1>X2THEN71
73 INPUT"Y-INTERVALL=";Y1,Y2
90 IFY1>Y2THEN73

100 INPUT"MEINHEIT AUF X=";E1
120 IFE1<(X2-X1)/50THEN100
130 INPUT"MEINHEIT AUF Y=";E2
150 IFE2<(Y2-Y1)/50THEN130
180 D1=(X2-X1)*14/350
190 D2=(Y2-Y1)*14/350
200 SCALEX1-D1,X2+D1,Y1-D2,Y2+D2
210 XAXIS0,E1,X1,X2
220 YAXIS0,E2,Y1,Y2
230 PRINT"BITTE WARTEN"
380 MOVEX1,FNA(X1)
390 FORI=X1TOX2STEP(X2-X1)/175
420 PLOTI,FNA(I)
430 NEXT
440 DRAW
    
```

Bild 4. Beispielprogramm zum Zeichnen einer Sinus-Kurve

henden Platz. Die Adresse ist aus den beiden Zero-Page-Zellen \$34 und \$35 zu erfahren.

Fehlermeldungen der Grafik-Software

Die Erweiterung kennt vier verschiedene Fehlermeldungen:

?NO FRAME ERROR

Es wurde versucht, einen Befehl auszuführen, ohne das zuvor FRAME erfolgte.

?NO SCALE ERROR

Einer der Befehle DOT, PLOT, MOVE, XAXIS oder YAXIS wurde gegeben, ohne daß das Feld zuvor skaliert wurde.

?OUT OF SCALE ERROR

Ein Parameter einer der koordinatenbezogenen Befehle hat die durch SCALE vorgegebenen Grenzen überschritten.

?BOUND ERROR

Die Untergrenze in einem SCALE-Befehl ist größer als die Obergrenze.

Laden und Initialisieren des Programms

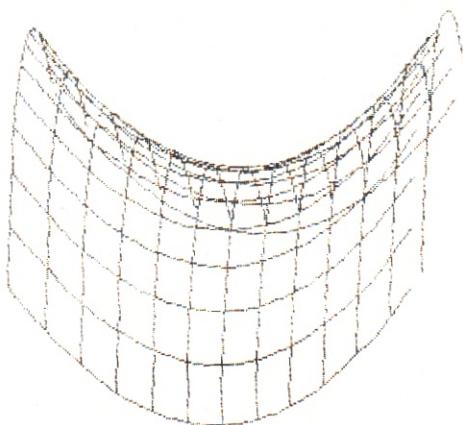
Das Programm (Bild 3) belegt den oberen Speicherbereich ab \$7400. Wird es mit dem LOAD-Befehl geladen, so muß an-

schließend ein NEW erfolgen. Ein etwa bereits im Speicher befindliches Basic-Programm wird dabei zerstört. Deshalb sollte das Programm vorzugsweise mit dem .L-Befehl des Monitors geladen werden. Aber Vorsicht: Aufgrund eines Fehlers im Betriebssystem funktioniert dieser Befehl nur, wenn das Statusbyte (Adresse \$0096) null ist. Befindet sich das Programm im Speicher, so wird es mit SYS 29696 initialisiert. Die oben beschriebenen Befehle stehen zur Verfügung. Die Decodierung der neuen Anweisungen geschieht auf die in [2] beschriebene Art. Die dadurch bedingte Verlangsamung der Arbeitsgeschwindigkeit des Interpreters ist nur geringfügig und ohne Bedeutung. Die Bilder 4 und 5 zeigen zwei Beispiele für die Anwendung der neuen Befehle.

Literatur

- [1] Druckerhandbuch CBM 3022/3023, Fa. Commodore.
- [2] AIM-65-Basic-Erweiterung. FUNKSCHAU 1981, Heft 2.

Bild 5. Mit einem verhältnismäßig überschaubaren Programm entsteht eine räumliche Grafik, ein Hyperboloid



```

20 FRAME6,6
30 SCALE-5,5,-5,5
110 A0=75
140 K=.5
180 FORY=-3TO3STEP.5
190 X0=-3:Y0=Y:GOSUB370
200 MOVES,T
210 FORX=-3TO3STEP.2
220 X0=X:Y0=Y:GOSUB370
230 PLOTS,T
240 NEXT NEXT
260 FORX=-3TO3STEP.5
270 X0=X:Y0=-3:GOSUB370
280 MOVES,T
290 FORY=-3TO3STEP.2
300 X0=X:Y0=Y:GOSUB370
310 PLOTS,T
320 NEXT NEXT
340 DRAW
369 END
370 Z=(X0*X0-Y0*Y0)/4
380 S=X0+K*Y0*COS(A0*PI/180)
390 T=Z+K*Y0*SIN(A0*PI/180)
400 RETURN
    
```

Herwig Feichtinger

Strichcode lesen mit AIM-65 und PC-100

Zwei verbreitete Computer fehlen uns in der Sammlung der Strichcode-Einlese-Software: Die intern baugleichen Geräte AIM-65 (Rockwell) und PC-100 (Siemens). Am folgenden Programm, das unabhängig von der ursprünglichen 6502-Version des Apple-II entwickelt wurde, sieht man auch gleich, wie unterschiedlich Programme für den gleichen Zweck ausfallen können.

Der Strichcode-Leser wird bei AIM-65 und PC-100 am Port PA5 angeschlossen (Bild 1), der am Application Connector herausgeführt ist. Das Programm (Bild 2) verwendet den Timer T1 desselben VIA-Bausteins; er ist nach einem Reset stets im gewünschten One-Shot-Modus, so daß keine weitere Initialisierung der Steuerregister nötig ist. Die Software getestet ausgiebig, ob es sich wirklich um Strichcode handelt, z. B. durch wiederholte Zeitabfragen. Dabei wird zur Übergabe einer Fehlermeldung aus Unterprogramm-Ebenen das Carry-Flag der CPU verwendet, außer bei TST, wo es anzeigt, daß sich der Leser auf einem schwarzen Strich befindet. Wird eine Zeile der Länge Null gelesen (letzte Zeile), so erfolgt ein Rücksprung zum Monitor über RTS. Dies setzt den Programmstart mit einer der Tasten F1...F3 voraus.

Die Bedienung des Programms

Zunächst geht man mit Taste E in den Texteditor und wählt die gewünschte Anfangsadresse, ab der die Zeichen abgespeichert werden sollen; bei Basic-Programmen gewöhnlich 0200, bei Maschinenprogrammen die Startadresse. Als Endadresse (der Editor fragt automatisch danach) sollte man stets eine Adresse unter 0F20 eingeben. (Die Einhaltung dieser Endadresse wird allerdings beim Einlesen des Strichcode nicht geprüft.) Ist die Initialisierung des Editors erfolgt, so geht man z. B. mit Escape in das Monitorprogramm und beginnt das Strichcode-Lesen nach Druck auf F1. Sobald der Leser auf weißem

Papier liegt, erscheint im Display die Nummer der jeweils einzulesenden Zeile (zuerst 01). Ein Fehler wird mit ERROR quittiert, und der Zeilenzähler wird dann nicht inkrementiert. Das Programm paßt sich in einem sehr großen Bereich der Lesegeschwindigkeit an, und es ist belanglos, ob man in einer oder in zehn Sekunden über eine Zeile fährt. Ein goldenes Mittelmaß für erste Versuche ist aber eine Zeit von etwa drei Sekunden für eine Zeile. Nach dem Einlesen der letzten Zeile erscheint links im Display wieder der Monitor-Prompt, und man kann mit T in den Editor zurück, um nachzusehen, was eigentlich geladen wurde. Sollte in der ersten Zeile nur Merkwürdiges stehen, sollte man mit dem M-Befehl das erste Textbyte (gewöhnlich bei 0200) untersuchen. Ist es hex 0D, sollte man es mit dem „/“-Befehl durch hex 20 ersetzen.

Übernahme in den Basic-Interpreter

Hat man eventuell nötige Anpassungen des Programms mit dem C-Befehl des Editors durchgeführt, so kann man den Programmtext (wenn es sich um ein Basic-Programm handelt) nun auf Kassetten abspeichern; man sollte sich vorher aber vergewissern, ob der Gap-Wert in Zeile A409 mindestens hex 18 ist. Ferner es nötig, als letzte Basic-Programmzeile ein CTRL-Z einzugeben; es ist im Display unsichtbar. Dann initialisiert man Basic und lädt den Test von der Kassetten per LOAD in den Basic-Interpreter. Dieses Verfahren mag auf den ersten Blick umständlich erscheinen, hat aber den großen Vorteil, daß man vor der Übernahme in Basic die komfortablen Editiermöglichkeiten des Texteditors in Anspruch nehmen kann.

In Kürze wird es bei mc zu interessanten Programmen auf Anforderung Strichcode-Listings geben.

Literatur

- [1] ROM und RAM in AIM-65 und PC-100. Franzis-Sonderheft Nr. 33, „Mikrocomputer-Anwendungen“.
- [2] Apple-II liest Strichcode. mc 1981, Heft 2.

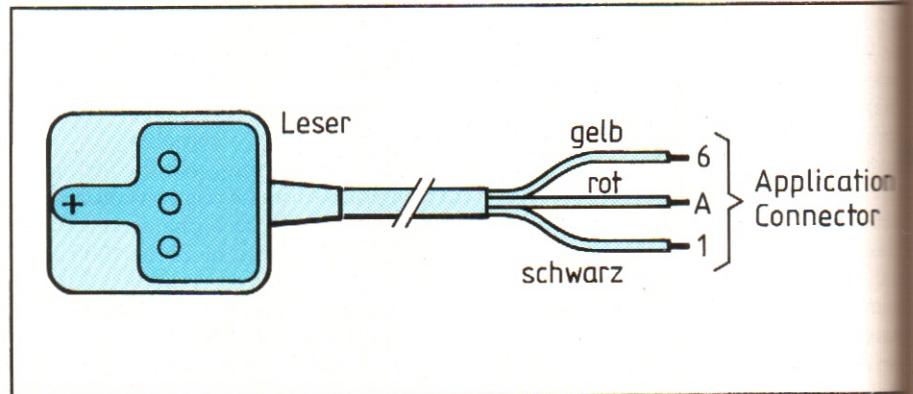


Bild 1. Anschluß des Strichcodelesers an AIM-65 oder PC-100

Assemblerlisting der Software zum Strichcode-Lesen

```

00FC PASS 1
00FC PASS 2
0000 ;BARCODE READER
0000 ;MC/FE/11-1981/AIM
0000 PA =$A00F
0000 TIMH =$A005
0000 IFR =$A00D
0000 NUMA =$EA46
0000 CKER =$E394
0000 CRLF =$E9F0
0000 DISP =$E9BC
0000 BUF =$116
0000 *=$E1
00E1 ;TEXT EDITOR
00E1 ADL *=$+1
00E2 ADH *=$+1
00E3 TPL *=$+1
00E4 TPH *=$+1
00E5 *=$E7
00E7 CHR *=$+1
00E8 REF *=$+1
00E9 LEN *=$+1
00EA CHKL *=$+1
00EB CHKH *=$+1
00EC CNT *=$+1
00ED *=$10C
00EC 4C20OF JMP INIT
00ED *=$F20
00F0 ;INIT FOR 1ST LINE
00F0 INIT A5E3 LDA TPL
00F2 85E1 STA ADL
00F4 A5E4 LDA TPH
00F6 85E2 STA ADH
00F8 A901 LDA £1
00FA 85EC STA CNT
00FC ;WAIT FOR WHITE
00FC WWH A9FF LDA £255
00FE 8D05A0 STA TIMH
0100 WWH1 20B00F JSR TST
0102 B0F6 BCS WWH
0104 2CODAO BIT IFR
0106 50F6 BVC WWH1
0108 ;SHOW LINE NR.
010A 20FOE9 JSR CRLF
010C A5EC LDA CNT
010E 2046EA JSR NUMA
0110 ;WAIT TILL BLACK
0110 SYN 20B00F JSR TST
0112 90FB BCC SYN
0114 ;CHECK 2 SYNCHS
0116 20CBOF JSR BLT3
0118 B0DF BCS WWH
011A 4A LSR A
011C 85E8 STA REF
011E 20BCOF JSR BLT
0120 B037 BCS ERR
0122 C5E8 CMP REF
0124 9034 BCC ERR
0126 ;READ LENGTH BYTE
0128 20E40F JSR RDB
012A B02E BCS ERR
012C 86EA STX CHKL
012E 86EB STX CHKH
0130 A8 TAY
0132 FO2D BEQ RDY
0134 C914 CMP £20
0136 B023 BCS ERR
0138 85E9 STA LEN
013A ;READ DATA BYTES

```

```

OF6B RDD 20E00F JSR RDB
OF6E B01C BCS ERR
OF70 991601 STA BUF,Y
OF73 65EA ADC CHKL
OF75 85EA STA CHKL
OF77 9002 BCC RDD1
OF79 E6EB INC CHKH
OF7B RDD1 88 DEY
OF7C DOED BNE RDD
OF7E ;GET CHECKSUM
OF7E 20E00F JSR RDB
OF81 C5EA CMP CHKL
OF83 D007 BNE ERR
OF85 20E00F JSR RDB
OF88 C5EB CMP CHKH
OF8A F009 BEQ STO
OF8C ;DISPL.'ERROR'
OF8C ERR 2094E3 JSR CKER
OF8F 4C2COF JMP WWH
OF92 ;BACK TO MONITOR
OF92 RDY 91E1 STA (ADL),Y
OF94 60 RTS
OF95 ;STORE IT TO MEMORY
OF95 STO A6E9 LDX LEN
OF97 ST01 BD1601 LDA BUF,X
OF9A 91E1 STA (ADL),Y
OF9C E6E1 INC ADL
OF9E D002 BNE ST02
OFA0 E6E2 INC ADH
OFA2 ST02 CA DEX
OFA3 DOF2 BNE ST01
OFA5 F8 SED
OFA6 A5EC LDA CNT
OFA8 6900 ADC £0
OFAA 85EC STA CNT
OFAC D8 CLD
OFAD 4C2COF JMP WWH
OFB0 ;
OFB0 ;SUBROUTINES
OFB0 ;
OFB0 ;TEST IF BLACK
OFB0 TST 48 PHA
OFB1 18 CLC

```

```

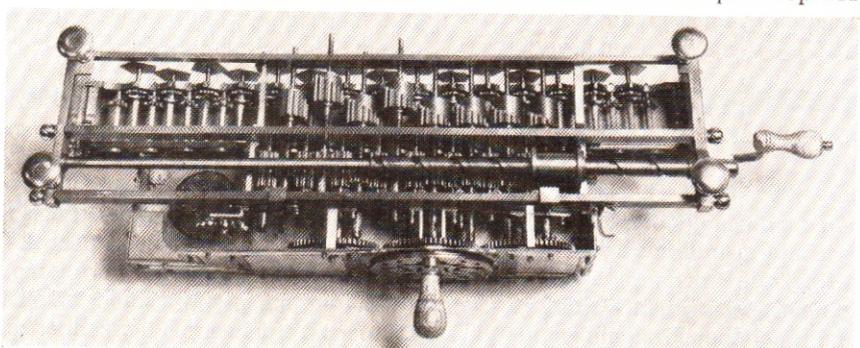
OFB2 ADOFA0 LDA PA
OFB5 2920 AND £$20
OFB7 F001 BEQ TST1
OFB9 38 SEC
OFBA TST1 68 PLA
OFBB 60 RTS
OFBC ;WAIT FOR BLACK
OFBC BLT A9FF LDA £$FF
OFBE 8D05A0 STA TIMH
OFC1 BLT0 2CODAO BIT IFR
OFC4 7018 BVS BLT2
OFC6 20B00F JSR TST
OFC9 90F6 BCC BLT0
OFCB ;TEST BLACK TIME
OFCB BLT3 A9FF LDA £$FF
OFCD 8D05A0 STA TIMH
OFDO BLT1 2CODAO BIT IFR
OFD3 7009 BVS BLT2
OFD5 20B00F JSR TST
OFD8 B0F6 BCS BLT1
OFDA 4D05A0 EOR TIMH
OFDD 60 RTS
OFDE BLT2 38 SEC
OFDF 60 RTS
OFE0 ;READ A BYTE,
OFE0 ;ADJUST SPEED
OFE0 RDB A208 LDX £8
OFE2 RDB2 20BCOF JSR BLT
OFE5 B014 BCS RDB0
OFE7 C5E8 CMP REF
OFE9 66E7 ROR CHR
OFEB C5E8 CMP REF
OFED B003 BCS RDB1
OFEF 0A ASL A
OFF0 65E8 ADC REF
OFF2 RDB1 4A LSR A
OFF3 85E8 STA REF
OFF5 CA DEX
OFF6 DOEA BNE RDB2
OFF8 A5E7 LDA CHR
OFFA 18 CLC
OFFB RDB0 60 RTS
OFFC .END
OFFC ERRORS= 0000

```

Spruch des Monats

„Es wird dann beim Auftreten von Streitfragen für zwei Philosophen nicht mehr Aufwand an wissenschaftlichem Gespräch erforderlich sein als für zwei Rechnerfachleute. Es wird genügen, Schreibzeug zur Hand zu nehmen, sich vor das Rechengert zu setzen und zueinander (wenn es gefällt, in freundschaftlichem Ton) zu sagen: Laßt uns rechnen.“

LEIBNIZ, um 1680 in: *De scientia universali seu calculo philosophico*



Michael Konz

AIM-65 macht Musik

Auch Mikrocomputer-Profis benutzen ihr „Siliziumhirn“ manchmal zu durchaus freizeit-füllenden Zwecken. Hier dient der AIM-65 (oder der PC-100) dazu, auf eine recht komfortable Methode Melodien zu komponieren und zu erzeugen.

Für fast jeden Mikrocomputer gibt es inzwischen Programme, die es erlauben, auf einem angeschlossenen Lautsprecher Melodien wiederzugeben. Im Bedienungskomfort unterscheiden sich diese Programme jedoch erheblich. Das folgende Programm für den AIM-65 zeichnet sich durch eine besonders bequeme Erstellung der notwendigen Daten aus: Der Editor des AIM-65 wird zur Eingabe der (einstimmigen) Melodie benutzt. Mit ihm können fertig komponierte Melodien auch auf Kassette gespeichert werden.



Bild 1. Beispiele für die Toncodierung

Nach dem Aufruf des Editors, wobei der Speicherbereich 0200...052C, in dem das Programm steht, zu meiden ist, werden

Bild 2. Anschluß eines Lautsprechers

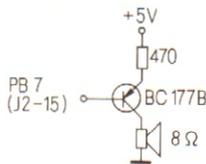


Bild 3. Das komplette Musikprogramm

```

0200 A9 60 8D 0E A0 A9 0D 8D 04 A4 A9 04 8D 05 A4 58 03A0 BA 65 AA 85 AA 20 BD 04 A5 AE 85 AF A5 AA 85 AC
0210 A9 00 85 B1 85 B2 A2 00 20 35 04 20 5F E9 20 46 03B0 A5 AB 85 AD A9 FB 8D 80 A4 24 A5 10 0D A5 A8 A4
0220 04 B0 F3 85 A7 AD 12 A4 85 A0 A9 4D 8D 12 A4 20 03C0 A9 BC 06 A0 8D 07 A0 4C EF 03 A9 40 8D 0B A0 A9
0230 73 04 A2 0C 20 35 04 20 5F E9 20 46 04 B0 F3 85 03D0 50 8D 08 A0 A9 C3 8D 09 A0 A9 20 2C 0D A0 F0 FB
0240 A1 A2 13 20 35 04 20 85 E7 A2 00 86 AF E8 8E 09 03E0 A2 C0 A5 A8 A4 A9 8C 04 A0 8D 05 A0 8E 0B A0 A2
0250 A0 A9 00 85 B0 20 53 04 C9 40 D0 16 20 BD 04 20 03F0 A0 A5 AA A4 AB 8C 08 A0 8D 09 A0 2C 82 A4 10 23
0260 90 E7 D0 03 4C 23 04 A2 00 8E 0B A0 20 73 04 4C 0400 8E 0E A0 A9 FF 8D 80 A4 A5 A6 4C 58 02 48 A5 AD
0270 49 02 C9 00 F0 E6 C9 2D D0 09 20 53 04 A2 80 86 0410 8D 08 A0 A5 AC 8D 09 A0 C6 AF D0 05 A9 20 8D 0E
0280 A5 D0 69 A2 00 86 A5 C9 28 D0 03 20 53 04 C9 29 0420 A0 68 40 A9 00 8D 0B A0 A9 60 8D 0E A0 A5 A0 8D
0290 D0 36 A5 B0 D0 BB E6 B0 A0 00 A5 DF D0 02 C6 E0 0430 12 A4 4C 82 E1 20 F0 E9 BD C9 04 C9 3B F0 06 20
02A0 C6 DF B1 DF C9 28 F0 AD C9 40 D0 06 20 53 04 4C 0440 BC E9 E8 D0 F3 60 C9 30 90 07 C9 3A B0 04 29 0F
02B0 55 02 C9 0D D0 E4 F8 A5 B1 38 E9 01 85 B1 A5 B2 0450 24 38 60 20 93 E9 F0 FA C9 20 F0 F7 C9 0D D0 12
02C0 E9 00 85 B2 D8 4C 9A 02 C9 50 D0 20 53 04 20 0460 F8 18 A5 B1 69 01 85 B1 A5 B2 69 00 85 B2 D8 4C
02D0 46 04 90 05 A0 01 4C 89 04 A2 00 86 A8 86 A9 4A 0470 53 04 60 20 BC F8 A6 A7 CA 30 0B 20 93 E9 F0 07
02E0 85 A3 46 A4 20 53 04 85 A6 4C 72 03 C9 23 D0 06 0480 C9 40 D0 F7 F0 F2 60 A0 04 A2 21 20 35 04 98 20
02F0 20 53 04 A2 07 2C A2 00 86 A2 C9 48 D0 02 A9 42 0490 46 EA A2 2A 20 38 04 A5 B2 20 46 EA A5 B1 20 46
0300 38 E9 41 B0 05 A0 02 4C 89 04 C9 07 B0 F7 65 A2 04A0 EA C0 04 F0 15 20 5F E9 20 93 E9 C9 20 D0 F9 A9
0310 48 20 53 04 C9 2C F0 14 20 46 04 B0 B7 48 20 53 04B0 02 48 A9 57 48 A9 0D 4C 56 04 4C 23 04 A6 AF D0
0320 04 C9 2C F0 05 A0 03 4C 89 04 68 2C A9 00 85 A2 04C0 FC A9 20 2C 0D A0 F0 FB 60 4D 45 4C 4F 44 59 20
0330 20 53 04 20 46 04 B0 9C 4A AA BD 28 05 85 A3 20 04D0 4E 4F 2E 3D 3B 53 50 45 45 44 3D 3B 23 20 4F 46
0340 53 04 C9 2E D0 06 A2 80 20 53 04 2C A2 00 86 A4 00 86 A4 04E0 20 52 45 50 45 41 54 53 3D 3B 2A 2A 45 52 52 4F
0350 85 A6 68 0A AA BD F8 04 85 AB BD F9 04 85 A9 A6 04F0 52 20 3B 20 49 4E 20 3B 08 DF 07 E7 0E ED 0D 4C
0360 A2 F0 0F A5 A9 D0 02 C6 A8 C6 A9 46 A8 66 A9 CA 0500 0B D8 0B 2F 09 F5 08 60 07 76 0E 18 0C 8E 0B 2E
0370 D0 F1 A5 A1 0A AA A9 63 85 AE BD 14 05 85 AA BD 0510 0A 8D 09 66 9C 40 75 30 5D C0 4E 20 3E 80 34 25
0380 15 05 85 AB A6 A3 F0 07 46 AA 66 AB CA D0 F9 24 0520 2E E0 27 10 22 17 1F 40 00 01 02 04 03 DA DF 7F
0390 A4 10 12 A5 AB A6 AA 46 AA 66 AB 18 65 AB 85 AB
    
```

ROR 4 IN 0000' und das Programm springt zum Monitor zurück. Bei erfolgreicher Suche wird der Benutzer anschließend aufgefordert, die Spielgeschwindigkeit zu bestimmen. Er hat dabei die Wahl zwischen 60, 80, 100, 120, 150, 180, 200, 240, 275 und 300 Viertelnoten pro Minute, was den Ziffern von 0...9 entspricht. Als letztes wird noch eingegeben, wie oft der Computer die Melodien spielen soll. Die Eingabe erfolgt in der vom K-Befehl des Monitors bekannten Weise, d. h. als zweistellige Dezimalzahl.

Das Programm verfügt über verschiedene Fehlermeldungen, um aufzuzeigen, wann während der Eingabe der Daten Fehler entstanden sind. Dabei wird neben der Art des Fehlers auch noch die Zeile, in der er auftritt, angegeben. Zur leichteren Korrektur empfiehlt es sich daher, in einer Editor-Zeile nur jeweils einen Takt unterzubringen. Die Fehlermeldungen bedeuten im einzelnen folgendes:

- **ERROR 1 – Das Programm hat eine Ziffer erwartet.
- **ERROR 2 – Das Programm hat einen Buchstaben (A, C, D, E, F, G, H) erwartet.
- **ERROR 3 – Hier sollte ein Komma stehen.

Nach erfolgter Fehlermeldung kann das Spiel durch Drücken einer beliebigen Taste fortgesetzt werden, um möglicherweise weitere Fehler zu entdecken. Ein Abbruch des Spiels ist jederzeit mit ESC möglich.

Wenn das Programm in Verbindung

...laufen soll, dann ist zu beach-
...die Zero-Page-Adressen A0...B2
...benutzt werden dürfen, ebensowe-
...vom Editor belegte Arbeitsbe-
...Es werden sowohl Timer 1 als
...Timer 2 im User-VIA verwendet,
...letzterer zur Interrupterzeugung
...gezogen wird. Das Tonsignal kann
...PB 7 abgenommen werden. Bild
...wie ein 150-Ω-Lautsprecher an-
...schaltet wird. Bild 3 das AIM-Pro-
...
...s bereits gespeichertes Stück kann
...in eine andere Tonart transponiert
...
...La Paloma“ als Codierung im AIM-Texteditor

werden. Die Tabelle zeigt, wie die Fre-
quenztafel von 04F8...0513 zu ändern ist.
Aus den aus der ersten Zeile zu entneh-
menden Basiswerten für die Töne lassen
sich leicht die Werte für die höheren
Oktaven dadurch ableiten, daß man von
dem Wert 1 subtrahiert und anschlie-

Tabelle: Transponierung in andere Tonarten

a	h	c	d	e	f	g	ais	c ¹	cis	dis	f	fis	gis	
8DF	7E7	EED	D4C	BD8	B2F	9F5	860	776	E18	C8E	B2E	A8D	966	C-Dur
5EB	546	9F5	8DF	7E7	776	6A5	597	4FA	966	860	776	70B	646	G-Dur

```

-[L]
/
OUT=Y
G1,8
(G1,2-
G1,8E1,6F1,6G1,8A1,8
H1,6C2,8A1,6H1,8G1,8
F1,2-
F1,4, D2,8
D2,2-
D2,6E2,8C2,6D2,8H1,8
C2,6H1,8A1,6G1,8F1,8
E1,2-
E1,4, G1,8
G1,2-
G1,8E1,6F1,6G1,8A1,8
H1,6C2,8A1,6H1,8G1,8
F1,2-
F1,4, D2,8
D2,2-
    
```

```

D2,6E2,8C2,6D2,8H1,8
C2,6H1,8A1,6G1,8A1,8
C2,2-
C2,4 P8 G1,8 )
C2,8C2,6C2,6C2,8H1,8
D2,8D2,6C2,6H1,8A1,8
A1,4 G1,4-
G1,4, G1,8
H1,8H1,6H1,6H1,8A1,8
A1,8G1,6G1,6G1,8A1,8
G1,6 F1,6 E1,8-E1,4-
E1,4, G1,8
C2,8C2,6C2,6C2,8H1,8
D2,8D2,6C2,6H1,8A1,8
A1,4 G1,4-
G1,4, G1,8
H1,8H1,6H1,6H1,8A1,8
A1,8G1,6G1,6G1,8A1,8
G1,6 F1,6 E1,8-E1,4-
E1,8 G1,6 G1,6 G1,6 A1,8 #F1,6
    
```

```

G1,2-
G1,8G1,6G1,6G1,8A1,8
H1,8 D2,4.-
D2,6 F2,8 E2,6 D2,8 E2,8 C2,8
D2,8 H1,8 C2,6 A1,6 H1,6 C2,6
E2,8 D2,4.-
D2,6 D1,6 E1,6 F1,6 A1,8 G1,8
E1,2-
E1,8 G1,6 G1,6 G1,6 A1,8 #F1,6
G1,2-
G1,8 G1,6 G1,6 G1,8 A1,8
H1,8 D2,4.-
D2,6 F2,8 E2,6 D2,8 E2,8 C2,8
D2,8 H1,8 C2,6 A1,6 H1,6 C2,6
E2,8 D2,4.-
D2,6 D1,6 E1,6 F1,6 A1,8 G1,8
C2,2-
C2,4 P4
END
    
```

Disk-Append beim IBM 3000

Bild aufgelistete Maschinenpro-
gramme findet in dem normalerweise
zwei Speicherbereich des Kassetten-
speichers Platz und erlaubt es bei Com-
puter der Serie CBM 3000, ein Basic-
programm von der Floppy-Disk an ein
im Speicher stehendes anzuhän-
gen. Voraussetzung ist, daß das nachzu-
hängende Programm höhere Zeilennum-
mern aufweist. Das Append-Maschinen-
programm erklärt sich beim Betrieb
selbst. Klaus Gano

```

826 #033A A9 9C LDA #9C : Textausgabe 'FILENAME = '
828 #033C A0 03 LDY #03
830 #033E 20 1C CA JSR CA1C
833 #0341 20 6F C4 JSR C46F : Eingabe abwarten
836 #0344 E8 INX : Filelänge berechnen und abspeichern
837 #0345 BD 00 02 LDA 0200,X
840 #0348 D0 FA BNE 0344
842 #034A 06 D1 STX D1
844 #034C A9 00 LDA #00 : Floppy-Parameter setzen
846 #034E A0 02 LDY #02
848 #0350 85 96 STA 96
850 #0352 85 DA STA DA
852 #0354 84 DB STY DB
854 #0356 88 DEY
855 #0357 84 D2 STY D2
857 #0359 A9 00 LDA #00
859 #035B 85 D4 STA D4
861 #035D A9 60 LDA #60
863 #035F 85 D3 STA D3
865 #0361 20 0A F4 JSR F40A : Ausgabe 'SEARCHING FOR XXXXXXXX'
868 #0364 20 66 F4 JSR F466 : Open
871 #0367 20 B6 F0 JSR F0B6 : Talk
874 #036A A5 D3 LDA D3 : Sekundäradresse auf BUS ausgeben
876 #036C 20 28 F1 JSR F128
879 #036F 20 8C F1 JSR F18C : Programmstartzeiger einlesen
882 #0372 20 8C F1 JSR F18C
885 #0375 A5 2E LDA 2E : Startadresse für anzuhängendes
887 #0377 85 FC STA FC Programm berechnen
889 #0379 38 SEC
890 #037A A5 2A LDA 2A
892 #037C E9 02 SEC #02
894 #037E 85 FB STA FB
896 #0380 B0 02 BCS 0384
898 #0382 C6 FC DEC FC
900 #0384 A9 A9 LDA #A9 : Ausgabe 'APPENDING'
902 #0386 A0 03 LDY #03
904 #0388 20 1C CA JSR CA1C
907 #038B 20 55 F3 JSR F355 : Programm laden
910 #038E 34 C9 LDY C9 : BASIC-Variablenzeiger auf Ende
912 #0390 A6 CA LDX CA des angehängten Programms setzen
914 #0392 84 2A STY 2A
916 #0394 86 2B STX 2B
918 #0396 20 42 C4 JSR C442 : Link-Adressen berechnen u. abspeichern
921 #0399 4C 89 C3 JMP C389 : Ausgabe 'READY.'
    
```

```

.\ 039C 0D 46 49 4C 45 4E 41 4D .FILENAME
.\ 03A4 45 20 3D 20 00 0D 41 50 E = ..AP
.\ 03AC 50 45 4E 44 49 4E 47 00 PENDING.
    
```

Maschinenprogramm zum Anhängen
des Basic-Programms von der Floppy an ein
im Speicher stehendes Programm

Herwig Feichtinger

Wie der Computer wieder auf die Füße fällt

Besonders bei Automobil-Anwendungen ist mit unvorhersehbaren Störimpulsen zu rechnen, die sich auf Adressen- oder Datenleitungen eines Mikrocomputers übertragen können und somit das Programm aus seinem normalen Ablauf werfen. Zum Glück gibt es aber Methoden, die dafür sorgen, daß das Programm nach einer solchen Situation wieder auf die Füße fällt.

Normalerweise wird ein als Steuer- oder Regeleinheit irgendwo in einer Applikation, etwa im Auto, eingebauter Mikrocomputer irgendwann – z. B. beim Umdrehen des Zündschlüssels – eingeschaltet und soll dann bis zum Wiederausschalten sein Programm abarbeiten. Im Gegensatz zu Tischcomputern kann man dabei vom Benutzer nicht erwarten, daß bei offensichtlicher Fehlfunktion dieser einen Reset-Knopf drückt, um das Programm wieder in einen definierten Zustand zu bringen.

Ursachen für „aufgehängte“ CPUs

Im wesentlichen gibt es nur zwei Möglichkeiten, daß ein Mikrocomputer ein Programm nicht so abarbeitet, wie sich der Programmierer das ursprünglich vorstellte: Entweder liegt ein Programmierfehler vor, der bei einer nicht vorhergesehenen und damit nicht rechtzeitig geprüften Betriebssystemsituation die Software in eine Endlosschleife geraten läßt, oder aber die Hardware wird von außen beeinflusst bzw. gestört, so daß für einen kurzen Augenblick die Signale auf dem Systembus nicht definiert sind. Die erste Möglichkeit läßt sich durch umfangreiche Tests weitgehend ausschließen, während man gerade in der

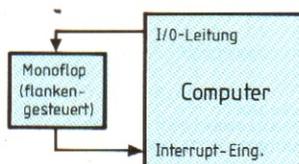


Bild 1. Wenn der Interrupt-Timer vom normal laufenden Betriebsprogramm immer wieder rechtzeitig nachgetriggert wird, tritt normalerweise kein Interrupt auf – erst dann, wenn das Programm sich aus irgendwelchen Gründen „aufhängt“ hat

störimpulsverseuchten Umgebung eines Automobil-Computers damit rechnen muß, daß durch kapazitive oder elektromagnetische Einstreuung der Prozessor plötzlich ein Byte aus seinem Programmspeicher nicht so liest, wie es dort programmiert wurde. Dies kann zu falschen Sprüngen, zu unzulässigen Operationscodes oder zur falschen Interrupt-Bearbeitung führen, letztendlich zu einer falschen Weiterführung oder gar zum „Aufhängen“ des Programms.

Problemlösung mit Interrupt-Timer

Da sich ein Mikrocomputer-Programm nun einmal nicht so schreiben läßt, daß es solche Probleme grundsätzlich ausschließt, muß man ein wenig Hardware einsetzen, um dafür zu sorgen, daß die CPU wieder auf die Füße fällt. Allerdings ist diese Hardware in vielen Mikrocomputern bereits vorhanden – nämlich ein programmierbarer Interrupt-Timer.

Die Funktion eines solchen Timers ist leicht erklärt: Er wird vom Programm aus mit einem bestimmten Zeitwert (z. B. 100 ms) geladen und löst nach Ablauf dieser Zeit einen Interrupt aus, der auch bei „aufgehängtem“ Programm wirksam ist, da er über einen im Programmspeicher-ROM stehenden Vektor zu einer Interrupt-Routine führt (Bild 1). Wenn man nun im Hauptprogramm, das im normalen Betrieb des Computers abgearbeitet wird, dafür sorgt, daß dieser Timer spätestens alle paar Millisekunden immer neu geladen wird, tritt der Interrupt nur dann auf, wenn dieses Hauptprogramm aus irgendwelchen Gründen nicht mehr läuft. Die Interrupt-Routine kann dann dafür sorgen, daß, wenn möglich, die Ursache des „Auf-

hängens“ erforscht und das Hauptprogramm an einer geeigneten Adresse neugestartet wird.

Welcher Interrupt ist geeignet?

Die meisten Mikroprozessoren verfügen über unterschiedliche Möglichkeiten, Interrupts auszulösen. Die dafür vorhandenen CPU-Anschlüsse nennen sich meist RES, NMI und IRQ.

Am naheliegendsten ist es, den Ausgang des Interrupt-Timers mit dem Reset-Anschluß RES zu verbinden, dann ist nämlich der Neustart des Betriebsprogramms an einer definierten Adresse sichergestellt. Allerdings wird das nicht immer sinnvoll sein, da die Reset-Sequenz im Programm oft wichtige Daten im RAM initialisiert, die bei einer kurzen Betriebsstörung eigentlich nicht verloren gehen dürften. Eine Alternative ist, das Programm von vornherein so auszulegen, daß die Initialisierung nicht automatisch bei jedem Reset erfolgt, sondern nur dann, wenn die CPU erkennt, daß die im RAM stehenden Daten unsinnig sind (was z. B. beim ersten Einschalten des Computers der Fall ist), was man dann „kalten Reset“ nennt.

Der meist ebenfalls vorhandene Anschluß IRQ (Interrupt Request) ist gewöhnlich weniger zur Selbstüberwachung des Computers geeignet, weil die CPU die Wirksamkeit dieses Interrupts durch Befehle wie SEI (Set Interrupt Disable) unterbinden kann. Wenn der Interrupt-Disable-Flag durch einen Störimpuls versehentlich gesetzt wird, funktioniert die freiwillige Selbstkontrolle des Programms nicht mehr. Geeigneter ist schon wieder der NMI-Anschluß (Non-Maskable Interrupt), ein Impuls an diesem CPU-Pin auf jeden Fall auch zu einem Sprung zur NMI-Interruptroutine führt. Leider besitzen nicht alle CPU-Typen einen NMI-Anschluß: während z. B. der 6502 dieses Komfort durchaus hat, verfügt sein kleiner Bruder 6504, der z. B. im Einplattener Mikrocomputer EMUF aus mc 1981, Heft 2, eingebaut ist, nicht über. Dann muß man den Reset-Anschluß für die freiwillige Selbstkontrolle des Systems verwenden (Bild 2).

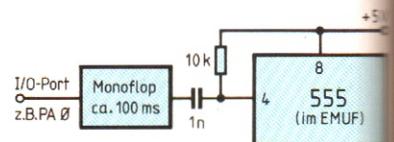


Bild 2. Beim Einplattener Mikrocomputer „EMUF“ aus mc 1981, Heft 2, läßt sich die Selbstüberwachung mit dem schon auf der Platine befindlichen Reset-Timer 555 mit einem Monoflop realisieren

Computer sind keine Taschenrechner

...fehler, die durch die binäre ...metik und die begrenzte Stellen... entstehen, lassen den EG 3003 (wie... viele andere Basic-Computer) vor... Billig-Taschenrechner erblassen. ... Die Stärke des Computers liegt ja... in erster Linie in seinen Rechen... , sondern in seinen alphanume... Ein- und Ausgabemöglichkeiten. ... wie dem auch sei, ganz so dumm... EG 3003 nun auch nicht. Winkel... tionen werden wider erstes Erwar... erstaunlich penibel behandelt. Hin... Rückrechnungen erfolgen natürlich... bis auf die n-te Stelle genau hinter... Komma, aber doch mit meist hin... ender Genauigkeit. ... grafischen Möglichkeiten des EG... halten sich wie auch beim TRS 80... Grenzen. Die Auflösung von 48 x 127... punkten ist für viele Fälle ausrei... höhere Ansprüche können aber... befriedigt werden. Dies stört den... professionellen Anwender eigent... , wenn er bei Spielen einen et... schärften Blick aus dem Cockpit... simulierten Flugkörpers hat.

Handbuchelemente

...Lieferumfang gehören drei Hand... , zwei davon in englischer Spra... . Unter der Bezeichnung Basic Manu... man das deutsche Betriebs... buch, welches einen Umfang von... hat. Was erwartet man von... solchen Handbuch? – Hier scheit... die Geister! ... end der absolute Anfänger mit die... Basic Manual doch recht gut klar... rümpft jeder „Fortgeschrittene“... die Nase ob der Unvollstän... Zugegeben, vollständig ist die... wichtigste Dokument nicht. Das wä... auch nicht sinnvoll, denn es gibt... Spezialliteratur über den... , die weitgehend auch auf den... EG 3003 angewendet werden kann. ... man die Mühen kennt und um die... einer Übersetzung weiß, dann... man den Produzenten nur Hoch... zollen. Hier waren keine „Nur... am Werk.

Wert und Unwert

...schwer, die Frage zu beantwor... auch heute die Entscheidung... für den EG 3003 gefallen wäre, ... Betriebserfahrungen vorliegen. ... ist der EG 3003 sicher wert... wird nicht betrogen. Zusammen

mit einem passenden Drucker hat man ein recht schönes System beisammen. Leider ist das Kunststoffgehäuse nicht abriebfest; sehr bald sieht die Holzimitation ziemlich schäbig aus. Die Tastatur hat bisher allen Ruinierungsversuchen widerstanden. Unbedingt anzuraten ist die regelmäßige Reinigung des Ton- und

Löschkopfs im Kassettenlaufwerk. Das System, so wie man es aus dem Karton auspackt, nutzt auch in Verbindung mit dem Basic Manual wenig, wenn man nicht Kontakte zu TRS-80-Brüdern knüpft und sich hier die kleinen Kniffe oder auch einmal ein fertiges Programm „abkuppert“.

Sortieren mit berechnetem Restore

Das Sortieren eines umfangreichen Datensatzes mit einem Basic-Programm ist eine zeit- und speicheraufwendige Sache. Bei einem Adreßverzeichnis etwa, das einmal nach Namen, ein andermal nach Postleitzahlen sortiert werden soll, müssen ja jeweils alle zusammengehörigen Daten mitsortiert werden. Das hier vorgeschlagene Verfahren ist schneller und weniger speicheraufwendig. Es setzt die Verwendung des berechneten Restore voraus, für das M. Penzkofer im FUNKSCHAU-Sonderheft „Mikrocomputer-Anwendungen“, S. 73, ein Programm vorgestellt hat (Bild 1). Das Prinzip ist einfach: Der Datensatz wird zunächst auf die Bestandteile abgefragt, nach denen sortiert werden soll (Zeile 140, in Bild 2). Sortiert und als Array gespeichert werden nur diese Datenwörter – hier die Ortsnamen – und die Zeilennummern (Zeilen 150...180). Unumgängliche Voraussetzung ist dazu nur, daß alle DATA-Zeilen nach dem gleichen Prinzip gebildet und in gleichen Abständen numeriert sind. Nach dem Sortiervorgang kann das Feld mit den Datenwörtern gelöscht werden; das empfiehlt sich vor allem bei umfang-

reichem Datensatz, sonst macht der Computer im weiteren Programmverlauf „Denkpausen“. Anschließend wird der Datensatz mit Hilfe des Restore-Maschinenprogramms und nach Maßgabe der sortierten Zeilennummern abgearbeitet (siehe SYS 860 Z in Zeile 200). Ein Vergleich der beiden Sortiermöglichkeiten (1. alle Daten in einem Durchgang mitsortieren – 2. nur Zeilennummern sortieren, dann berechnetes RESTORE) ergab eine Verringerung sowohl des zusätzlich benötigten Platzes im Arbeitsspeicher als auch der Sortierzeit auf weniger als die Hälfte. Da die zum Sortieren benötigte Zeit bei wachsendem Datensatz in einer Exponentialfunktion zunimmt, zahlt sich der etwas höhere Programmaufwand bald aus.

Meno Sellschopp

```
033A 28 8B CC 20 D2 D6 A0 28
034A A6 29 20 30 C5 90 03 4C
034A B0 C7 4C EB C7 A5 5D E9
0352 00 85 3F 60 4C EB C7 00
035A 00 00 20 8B CC 20 D2 D6
0362 A5 28 A6 29 20 30 C5 90
036A 0D A5 5C E9 01 85 3E A5
0372 5D E9 00 85 3F 60 4C EB
037A C7 00 00 00 00 04 FF 04
```

Bild 1. Hier noch einmal der Hex-Dump des Maschinenprogramms aus dem FUNKSCHAU-Sonderheft „Mikrocomputer-Anwendungen“, das den RESTORE-Befehl für bestimmte, berechnete Zeilennummern wirksam werden läßt (CBM-Betriebssystem)

```
100 REM ***** DATEN SORTIEREN *****
102 REM ** MIT BERECHNETEM RESTORE **
110 READN: DIMX$(N): DIMX(N)
120 FORA=1TON: Z=1000+A
130 READN$,V$,S$,P$,O$
140 X$=O$
150 IFA=1THENB=1:GOTO180
160 FORB=ATO2STEP-1
170 IFX$(B)=X$(B-1)THENB=300:NEXTB
180 X$(B)=X$:X(B)=Z:NEXTA
190 FORA=1TON: X$(A)="" :NEXT
200 FORA=1TON: Z=X(A):SYS860Z
210 READN$,V$,S$,P$,O$
220 PRINTV$;" ";N$:PRINTS$:PRINTP$;" ";O$
230 PRINT:NEXT:END
300 X$(B)=X$(B-1):X(B)=X(B-1):RETURN
1000 DATA 5
1001 DATA ABEL, BERTA, CUNOWRING 13, 4444, DEBERG
1002 DATA BABEL, ERICH, ALMGASSE 3 A, 2222, EWEILER
1003 DATA CABLE, DANIEL, BAUMSTRASSE 80, 5555, ADORF
1004 DATA DEBEL, CARL, ENNOPLATZ 3, 1111, CEHAUSEN
1005 DATA EBEL, ANTON, DONBERG 28, 3333, BESTADT
```

Bild 2. Programm zum Sortieren mit berechnetem Restore. In Zeile 1000 steht die Anzahl der zu sortierenden DATA-Zeilen