

Einzeipreis DM 7,50

# MAGAZIN

Das Profi-Magazin für alle Anwender von MZ-700/800

## Zwei Bildschirme für MZ-800

Gleich zu anfang ein Tip für diejenigen, die ihren MZ-800 mit einer Grafikerweiterung ausgerüstet haben (womöglich aufgrund des Hardware-Tips von Jan Zamojski in Heft 2/87, Seite 11). Die Grafikerweiterung läßt sich nicht nur dafür nutzen, 16 Farben gleichzeitig auf dem Bildschirm darzustellen. Wenn man sich weiterhin auf vier Farben (bzw. im 80-Zeichen-Modus auf zwei Farben) beschränkt, stehen mit der Grafikerweiterung zwei Bildschirmspeicher zur Verfügung, die sich unabhängig voneinander bearbeiten und anzeigen lassen.

Über den Nutzen braucht man nicht viele Worte zu verlieren: Man kann auf einem Bildschirm Hilfstexte ablegen und einblenden, ohne den Inhalt des anderen Bildschirms anschließend neu aufbauen zu müssen. Es lassen sich Bewegungseffekte erzielen, indem Bildschirm 2 im Hintergrund bearbeitet und dann eingeblendet wird (während dann wiederum Bildschirm 1 im Hintergrund bearbeitet wird) usw. Nur: Wie macht man das alles vom BASIC aus? Hierzu ein kleines Beispielprogramm. Die Routinen "Bild1" und "BILD2" aktivieren den jeweiligen Bildschirm für die Bearbeitung, d.h. alle Anzeigebefehle (CLS, PRINT, SET, LINE usw.) wirken sich auf diesen Bildschirm aus. Es ist zu beachten, daß die Cursorposition beim Bildschirmwechsel erhalten bleibt, d.h. nach einem PRINT-Befehl (ohne ";" am Ende) in Zeile 3 von Bildschirm 1 kommt Inhalt des nächsten PRINT-Befehls nach Zeile 4, egal welcher Bildschirm gerade aktiv ist. Ggf. muß also beim

Bildschirmwechsel erst mit CURSOR positioniert werden. Auch sollte man dafür sorgen, daß bei Beendigung des Programms stets Bildschirm 1 aktiviert wird. Ansonsten passiert allerlei Merkwürdiges. Mit OUT@ \$CE,0 (bzw. OUT@ \$CE,4 im 80-Zeichen-Modus) wird Bildschirm 1 sichtbar, mit OUT@ \$CE,1 Bildschirm 2 (OUT@ \$CE,5 für 80-Zeichen-Darstellung). *Jan Zamojski*

```

50 CLS:PRINT"DAS IST BILDSCHIRM 1"
60 WAIT 2000:GOSUB "BILD2"
70 CLS:PRINT"UND DAS IST BILDSCHIRM 2"
80 WAIT 2000:GOSUB "BILD1"
90 PRINT"JETZT SCHREIBEN WIR AUF BILD 1"
100 WAIT 2000:GOSUB "BILD2"
110 'CURSOR 0,1
120 PRINT"UND JETZT AUF BILD 2"
130 WAIT 2000:GOSUB "BILD1"
140 LINE 0,25,100,25
150 WAIT 2000:GOSUB "BILD2"
160 LINE 0,30,150,30
170 WAIT 2000:GOSUB "BILD1":POKE
180 LABEL "BILD1"
190 POKE $7527,3:POKE$137B,3:POKE$109C,0
200 '80-Z. MODUS POKE$7527,1 POKE$137B,1
210 POKE$109F,3:OUT@ $CE,0
220 '80-Z. POKE $109F,1 OUT@ $CE,4
230 RETURN
240 LABEL "BILD2"
250 POKE$7527,$1C:POKE$109C,$10:
    POKE$137B,$1C
260 '80-Z. POKE$7527,$14 POKE$137B,$14
270 POKE$109F,$1C:OUT@ $CE,1
280 '80-Z. POKE $109F,$14 OUT@ $CE,5
290 RETURN

```

Peinlich, peinlich, was hier in den letzten Wochen immer wieder am Telefon zu hören war. Eine Reihe Abonnenten hat offensichtlich das Heft 2 nicht erhalten. Manch einer hat keine Rechnung, dafür aber eine Mahnung bekommen. Da fragt man sich natürlich, wie so etwas passieren kann. Der MZ-Verlag ist ja kein Neuling bei Abo-Werken, denn schließlich bringen wir seit geraumer Zeit das *Handbuch für Heilpraktiker* heraus. Da klappt's ja auch.

Ursache des Dilemmas ist ein Programm. Wir benützen im MZ-Verlag Macintosh-Computer von Apple. Das sind wirklich tolle Computer, mit denen wir alle Bücher und sonstigen Druckwerke herstellen. Doch kaum ein Mensch würde auf die Idee kommen, diese teuren Maschinchen für Fakturierung und Buchhaltung einzusetzen. Deshalb darf man nicht wählerisch sein, wenn man entsprechende Software kaufen möchte. Wir haben das vor mehr als einem Jahr gemacht. Nur richtig funktioniert hat sie leider nicht. Deshalb mußten wir uns immer wieder mit Notlösungen über die Runden retten.

Jetzt haben wir endlich ein Programm, das richtig funktioniert. Nur der Computer fehlt jetzt noch, denn wir sind aus lauter Verzweiflung mit der Buchhaltung zur Konkurrenz übergelaufen: Der Neue ist ein AT-Kompatibler. Bis er da ist, helfen wir uns mit unserem MacCharlie. Doch MacCharlie hat leider keine Festplatte. Deshalb konnten wir noch nicht alle Daten übertragen. Aber danach wird es dann keine Probleme mehr geben, oder...?

Ich danke Ihnen herzlich für Ihr Verständnis! Denn eins eint uns ja alle: Unsere Computer bescheren uns täglich spannende Erlebnisse. Wer wäre nicht schon einmal ausgeflippt, wenn sich der Rech-



## Verleger Harald Schicke

ner dann verabschiedet (für die Anfänger: Das Programm stürzt ab), wenn man gerade mühevoll ganz viele Daten eingegeben oder ein umfangreiches Listing abgetippt hat. Man weiß natürlich, daß man sicherheitshalber immer mal wieder zwischendurch alles abspeichern soll. Aber da's lange dauert, insbesondere wenn man nur einen Kassettenrekorder benützt, läßt man's manchmal. Und das kostet dann ganz viel Zeit!

Ganz viel Zeit kostet es auch, wenn man seine Programme schlampig oder gar nicht dokumentiert. Nach und nach hat man das Programm verbessert, hier und da etwas hinzugefügt. Dann war man zufrieden. Bis sich plötzlich herausstellte, daß da doch noch ein Fehler war. Und dann saß man vor einem Listing, daß man selbst kaum noch verstehen konnte. Ein anderer könnte das schon gar nicht. Deshalb ist eine gute Dokumentation, im Programm oder auf Papier, sehr wertvoll. Gerade die besten Programmierer vergessen das gerne. Deshalb zeigt unser Hauptartikel von Herrn Beinert, wie

Redaktion: Harald Schicke, Edgar Lefgrün

Mitarbeiter dieser Ausgabe: Walter Beinert, Ulrich Christ, Edgar Lefgrün, Eckart Schatter, Harald Schicke, Karin Schicke, Jan Zamojski

Fotos:

Anzeigen: Harald Schicke

Vertrieb: Elisabeth Sternhagen

Druck: P & N, Hamburg

Verlag:

MZ-Verlag Harald Schicke, Postfach

(für Pakete: Lindenweg 18),

D-2110 Buchholz 5

☎ 0 41 87/65 33

Telex: 051933521 dmbx g

ref: box:dm4:mz-verlag

Magazin 700-800 ist eine unabhängige Zeitschrift und nicht SHARP Electronics angegliedert.

Magazin 700-800 erscheint sechs mal im Jahr. Der Einzelpreis beträgt DM 7,50. Im Abonnement kostet es DM 36,- pro Jahr (Ausland DM 42,-).

Für unverlangt eingereichte Manuskripte und Fotos übernimmt der Verlag keine Haftung. Nachdruck, auch auszugsweise, nur mit schriftlicher Genehmigung des Verlages. Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt. Gerichtsstand ist Tostedt.

Manuskripte dürfen nicht gleichzeitig anderweitig zur Veröffentlichung in Zeitschriften, Jahrbüchern usw. angeboten werden. Manuskripte sind an den MZ-Verlag Harald Schicke, Postfach, D-2110 Buchholz 5 zu senden. Für den Inhalt namentlich gekennzeichnete Beiträge tragen die Autoren die Verantwortung. Mit dem Abdruck erwirbt der Verlag das ausschließliche Recht der Vervielfältigung, auch auf Tonträgern, und die Rechte sonstiger Wiedergabemöglichkeiten, z.B. fotomechanisch, auf Mikrofilm, auf Datenträgern usw., ebenso das Recht der Übersetzung in fremde Sprachen und das Recht der Veröffentlichung im In- und Ausland. Autoren erhalten ein kostenloses Belegexemplar (ab Beiträgen von mindestens einer Seite Länge) sowie ein Honorar von DM 50,- pro Seite.

ISSN 0931-8860

man modular programmiert und dabei die Programme "zerstückelt", so daß sie leicht les- und änderbar bleiben. Ziel ist dabei die eigene Programm-Bibliothek, aus der man einfach die nötigen Stücke zu neuen Programmen zusammenfügt. ■

Beim MZ-700 werden durch das Video-RAM 25 Reihen mit je 40 Zeichen adressiert. Ein Byte (ab D000) beschreibt die Adresse des Zeichen-ROMs, ein zweites (ab D800) die Vordergrund- und die Hintergrund-Farbe. Das höchstwertige Bit des Farb-Bytes erlaubt noch die Wahl des ersten und des zweiten Zeichensatzes. Eine Grafik ist prinzipiell nur mit den vordefinierten Zeichen aus dem ROM möglich.

Beim MZ-800 wird zwar im 700er Modus in der gleichen Weise an den Bildschirm ausgegeben; das Zeichen-ROM (ab 1000) wird jedoch vorher in das Video-RAM (ab C000) kopiert und kann daher von dem Programmierer manipuliert werden. Man kann z.B. die Zeichen gleich nach dem ASCII-Code sortieren und spart sich damit die umständlichen Umcodierungen oder eigene Zeichen definieren.

Während dem MZ-700 im Video-RAM ganze 4096 Byte (4 KB) zur Verfügung stehen, sind es beim MZ-800 immerhin 16 KB in der Grundausstattung. Damit läßt sich im 80-Zeichen-Modus für jeden der 640 Punkte auf 200 Schirm-Zeilen ein Bit bereithalten, im 40-Zeichen-Modus sogar zwei. Eine kurze Rechnung: 16 KB=131 072 Bits, 640\*200=128 000. Die überzähligen Bits bleiben ungenutzt.

Mit zwei Bits lassen sich für jeden Punkt vier Zustände wahlfrei einstellen, also vier Farbcodes. Damit Sie trotz der etwas dürftigen Wahl 1 aus 4 jede Farbe darstellen können, ist jedem möglichen Code (hier 0...3) mit dem PAL-Befehl eine der 16 Farben (0...15) zuweisbar. Diese Zuweisung gilt dann natürlich für den ganzen Bildschirm.

INIT "CRT:M1":PAL 0,15:PAL 3,0 läßt die Schrift im BASIC schwarz auf weiß und nicht wie sonst weiß auf

schwarz erscheinen. Die 16 möglichen Farben ergeben sich aus den drei kombinierbaren Grundfarben blau (1), rot (2) und grün (4) sowie Intensiv-Hell (8). Wenn die 2 bzw. vier Codes zu ungenügend erscheinen, der sollte sich zwei RAMs vom Typ xx416 kaufen. Sie kosten zusammen zwischen 10 und 15 DM. Es sind 64K-RAMs mit der Organisation 16K\*4 Bit. Die verbreitetere Type 64K\*1 (im Programm-RAM) ist hier nicht einsetzbar, weil in den 50 Mikrosekunden einer Videozeile 320 Bits aus einem RAM gelesen werden müssen. Das ist nur mit 4 Bits auf einmal zu schaffen. Ich habe zwei D41416 C-12 von NEC in Betrieb. Andere RAMs vom Typ 416 tun es sicher genau so. Mit dem erweiterten RAM (siehe Handbuch S.7-18) funktioniert dann auch der INIT-Befehl mit M2 und M4 und der Bildschirm erstrahlt im M2-Modus in vollen 16 Farben. Noch nicht bunt genug? Testen Sie dann einmal das folgende Programm, das es ermöglicht, eine beliebige Randfarbe einzustellen. Wegen der verwendeten Pokes könnte es bei der älteren BASIC-Version 1Z016 Probleme geben. Diese Version stürzt auch ab, wenn der SYMBOL-Befehl mit einem undefinierten String benutzt wird. Die neuere CMT/QD-Version MZ-5Z009 hat diesen Fehler nicht.

Eckart Schatter ■

```

10 'I/O-Portsteuerung mit
    Adreßausgabe
20 FOR I=$680D TO $6813
30 POKE I,0:NEXT'Alle Ports
    ansprechbar
40 POKE $680B,$51,$85
50 POKE $6820,$51,$85
60 'Statt nur Portadr
    jetzt:
70 'B*256+Portadr
80 'B-Register wird auf die
    Adreßlines
90 'A8...A15 gegeben
100 'Syntax:
110 'INP@ $bbpp,var
```

Zwei Bildschirme	1
Editorial	2
Bildschirm-Ansteuerung beim MZ-800	3
Inhalt	3
Modulares Programmieren in BASIC	4
Club-News	7
Briefe	7
Tips & Tricks	7
Fragen & Antworten	8
Turbo-Lader für das 800er BASIC	9
Tips & Tricks	10
Leserbriefe	11
Tips & Tricks	12
Fragen	12
Textbearbeitung	13
MZ-Video	13
Konvertierung	14
Hardcopy-Programme für MZ-800 und MZ-1P16	15
Tips, Tricks & Informationen	16
Praxistest XBC-BASIC- Compiler	18
Kleinanzeigen	19
Grafisch orientierte Benutzeroberfläche	19
INPUT-Routine MZ-700	20

**STOP** Bevor Sie sich beschweren, bitten wir Sie noch um etwas Geduld! Leider gibt es z.Z. Lieferschwierigkeiten bei QD's. Deshalb sind die Freiprogramme noch nicht ausgeliefert worden. Wegen der großen Menge bestellter Disketten und Kassetten kann es auch hier gelgentlich zu Engpässen kommen. ■

```

120 'OUT@ $bbpp,var
130 'bb B-Register
140 'pp Portadresse
150 'var Variable
155 CLS
160 PRINT"Randfarbe
    eingeben (0...F) ";
170 GET A$:IF A$="" GOTO170
180 A=VAL("$"+A$):PRINTA$
200 OUT@$6CF,A:GOTO160
```

## Die Problematik der Unterprogramm-Technik (Schnittstellen)

In Fortsetzung meines Beitrages in Heft 1/87 soll nun auf die Probleme eingegangen werden, die sich bei dem Aufruf von externen Upro's (mittels CHAIN) ergeben können.

Für alle, die mit dem Begriff "saubere Schnittstellen" nichts anfangen können, hier zunächst einmal die allgemeine Definition dazu.

Schnittstellen im Softwarebereich sind Übergabe-Bereiche vom rufenden zum gerufenen Programm (und umgekehrt) und enthalten die Informationen, die für die Weiterverarbeitung gebraucht werden. In der Regel handelt es sich dabei um Datenbereiche, deren Indizes bei Tabellen, und/oder Programm-Zustands-Anzeiger (Schalter).

Doch nun zur Programmierungstechnik selbst. Bei der Anwendung des CHAIN-Befehls ergeben sich folgende Schwerpunkte:

- Die Wirkung des Befehls an sich
- Der Speicherbedarf für die Ausführung
- Die Verknüpfung mit dem externen Programm
- Der modulare Test der zugehörigen Programme

Auf diese Punkte will ich im Folgenden noch näher eingehen.

### Die Wirkung des CHAIN-Befehles

Da die Funktion CHAIN alle (bekannten) Variablen und deren Inhalte erhält, scheint seine Anwendung problemlos zu sein. Aber "der Teufel steckt bekanntlich im Detail" und ehe man es merkt, geht das System bei der Anwendung "in den Wald" und der Ungeübte assoziiert zum "Zauberlehrling".

Abhilfe schafft hier eine Parameterleiste, in der die übergebenen Felder enthalten sind, und die vor und nach dem Aufruf auf den Monitor ausgegeben werden.

### Der Speicherbedarf für die Übergabe

Man achte darauf, daß im Speicher außer dem rufenden Programm noch mindestens 2 KB für das Chainen zur Verfügung stehen. Ist dies nicht der Fall, muß das rufende Programm evtl. segmentiert (zweigeteilt) werden.

### Die Verknüpfung mit dem externen Upro

#### 1. Die Datenübergabe

Bei Übergabe von Tabellen müssen die DIM-Anweisungen bzgl. des Namens und der Dimensionen gleichlautend sein. Handelt es sich dabei um große Bereiche, sollte man diese als Zwischenspeicher auf Diskette anlegen und so übergeben.

Die DIM-Anweisung im gerufenen Programm muß übersprungen werden, da dieses eine Löschung des Bereiches einschließt. Man legt dieses deshalb immer an den Programmanfang, um sie ggf. in Abhängigkeit von dem Eingang erwähnten Zustands-Anzeiger des rufenden zu überspringen. Bei DATA-Anweisungen in beiden Programmen darf der RESTORE-Befehl nach der jeweiligen Verarbeitung nicht vergessen werden.

#### 2. Zustands-Anzeiger

Rufft man ein Upro von mehreren Programmen auf /z. B. bei gemeinsam benutzten Lese- und Schreibroutinen), enthält der Zustands-Anzeiger die Information, wohin das Unterprogramm nach Beendigung seiner Aufgabe zurückkehren soll.

Die Anzeiger sollten sowohl im rufenen als im gerufenen Programm in Inhalt und Wirkung gut erklärt sein.

### Modularer Test des Programmsystems

Darunter versteht man, daß jedes externe Upro für sich alleine getestet wird, indem man die Parameterleiste mit den Übergabewerten zu Beginn eines Upros simuliert, d. h. mit konstanten Werten füllt. Die anderen Upros sind zu dem Zeitpunkt schon getestet oder als Treiber-Module angelegt (Upros, die nur die Parameterleiste auf den Monitor ausgeben).

Zur Demonstration der Vorgehensweise der Unterprogramm-Technik dient das abgedruckte Programm-Listing.

In meinen nächsten Beiträgen werde ich anhand eines von mir selbst erstellten Programm-Systems über die Entwicklung, Programmierung, Integrations-Test und Dokumentation referieren.

Walter Beinert

```
10 REM *****
20 REM *
30 REM * Beispiel Unterprogr. Technik *
40 REM *
50 REM *****
60-REM
70 DIM Ps(30) :REM PROGRAMME
80 DIM Fs(30,3) :REM FUNKTIONEN
90 DIM Hs(30,3) :REM HERKUNFT
100 DIM Bs(30,3) :REM BEMERKUNGEN
110 REM
120 RESTORE 1790 :REM POSITION DATAS
130 REM :REM TABELLEN FUELLEN
140 FORI=1TO30
150 READPs(I)
160 NEXTI
170 REM
180 FORI=1TO30
190 FORK=1TO3
200 READPs(I,K)
210 NEXTK,I
220 REM
230 FORI=1TO30
240 FORK=1TO3
250 READHs(I,K)
260 NEXTK,I
270 REM
280 FORI=1TO30
290 FORK=1TO3
300 READBs(I,K)
310 NEXTK,I
```

```

320 REM
330 REM *****
340 REM *
350 REM * Erstellen Auswahl-Menue *
360 REM *
370 REM *****
380 REM
390 CURSOR2,1
400 COLOR,7,1
410 GOSUB3000 :REM RAHMEN-MENUE
420 CURSOR2,1
430 PRINTK1;K2; :REM KOPF-ZEILEN
440 GOSUB560 :REM PROGRAMM-ZEILEN
450 CURSOR5,23
460 INPUT"Ende / Auswahl / Liste = (E/A/L)";:AS
470 IFAs="E"THEN END :REM ENDE
480 IFAs="A"GOTO530 :REM AUSWAHL
490 IFAs<"L"GOTO450 :REM EINGABE
500 PRINT/PCHR$(27,66,3)
510 K=7
520 GOTO1270 :REM AUSGABE DRUCKER
530 COLOR,7,2
540 GOSUB730 :REM AUSGABE MONITOR
550 CURSOR2,1
560 FORI=5TO19
570 CURSOR3,I
580 IFI=4(10THENPRINT" ";
590 PRINTI-4;" ";:Ps(I-4);
600 CURSOR21,1
610 PRINTI+11;" ";:Ps(I+11)
620 NEXTI
630 RETURN
720 REM
730 GOSUB3000 :REM RAHMEN-MENUE
740 REM
750 PRINT" Programm - Auswahl"
760 REM
770 GOSUB560 * :REM PROGRAMM-ZEILEN

790 INPUT"Bitte die laufende Programm-Nr.:"N
800 CURSOR5,23
810 INPUT"Help = H / Start = S / Ende = E":AS
820 IFAs="E"GOTO1550 :REM AUFRUF HAPRO
830 REM
840 IFAs="S"GOTO890 :REM START UPROS
850 REM
860 IFAs="H"GOTO910 :REM HELP UPROS
870 REM
880 GOTO340 :REM AUSW. UPROS
890 Zs=As
900 GOTO1560 :REM AUFRUF UPROS
910 COLOR,7,0
920 GOSUB3000 :REM RAHMEN MENUE
930 CURSOR5,1
940 PRINT"Programminfo - ":Ps(N)
950 CURSOR5,4
960 PRINT"Programm-Funktion"
970 CURSOR5,5
980 PRINT"-----"
990 CURSOR5,6
1000 PRINTFs(N,1)
1010 PRINT":Fs(N,2)
1020 PRINT":Fs(N,3)
1030 PRINT":-----"
1040 PRINT": "Programm-Herkunft"
1050 PRINT": "-----"
1060 PRINT":Hs(N,1)
1070 PRINT":Hs(N,2)
1080 PRINT":Hs(N,3)
1090 PRINT":-----"
1100 PRINT": "Programm-Bemerkung"
1110 PRINT": "-----"
1120 PRINT":Bs(N,1)
1130 PRINT":Bs(N,2)
1140 PRINT":Bs(N,3)
1150 CURSOR5,23
1160 INPUT" Auswahl = A Start = S "::AS
1170 IF As="S"GOTO890
1180 IF As="A"GOTO340
1190 GOSUB1210
1200 GOTO1560
1210 K=1
1220 PRINT/P:PRINT/P
1230 PRINT/P"Programmliste ";K1;": ";K2;
1240 PRINT/P"-----"
1250 PRINT/P:PRINT/P
1260 RETURN
1270 FORI=1TO30
1280 IFPs(I)<>"Reserve"GOTO1300
1290 I=30:GOTO1470
1300 K=K+1:IFK>7GOSUB1210
1310 PRINT/P"Programm: ";:Ps(I);
1320 PRINT/PTAB(25)"Funktion: ";
1330 PRINT/PTAB(40):Ps(I,1)
1340 PRINT/PTAB(40):Ps(I,2)
1350 PRINT/PTAB(40):Ps(I,3)
1360 IFBs="E"GOTO1410
1370 PRINT/PTAB(25)"Herkunft: ";
1380 PRINT/PTAB(40):Hs(I,1)
1390 PRINT/PTAB(40):Hs(I,2)
1400 PRINT/PTAB(40):Hs(I,3)
1410 IFBs(I,1)="-"GOTO1470
1420 PRINT/PTAB(25)"Hinweis: ";
1430 PRINT/PTAB(40):Bs(I,1)
1440 PRINT/PTAB(40):Bs(I,2)
1450 PRINT/PTAB(40):Bs(I,3)
1460 PRINT/P
1470 NEXTI

1480 REM
1490 REM *****
1500 REM *
1510 REM * Beispiel Aufruf ext. Upro *
1520 REM *
1530 REM *****
1540 REM
1550 CHAIN "AUTO RUN"
1560 REM
1570 IFN>1GOTO1610
1580 REM
1590 CHAIN "Psy & IQ"
1600 REM
1610 IFN>2GOTO1650
1620 REM
1630 CHAIN "Sex & Strip"
1640 REM
1650 IFN>3GOTO1690
1660 REM
1670 CHAIN "Bio & Comp"
1680 REM
1690 IFN>3GOTO1710
1700 REM
1710 CHAIN "AUTO RUN"
1720 REM
1730 REM *****
1740 REM *
1750 REM * Beispiel DATA Anweisungen *
1760 REM *
1770 REM *****
1780 REM
1790 DATA Psy & IQ
1800 DATA Sex & Strip
1810 DATA Bio & Comp
1820 DATA Reserve
1830 DATA Reserve
1840 DATA Reserve
1850 DATA Reserve
1860 DATA Reserve
1870 DATA Reserve
1880 DATA Reserve
1890 DATA Reserve
1900 DATA Reserve
1910 DATA Reserve
1920 DATA Reserve
1930 DATA Reserve
1940 DATA Reserve
1950 DATA Reserve
1960 DATA Reserve
1970 DATA Reserve
1980 DATA Reserve
1990 DATA Reserve
2000 DATA Reserve
2010 DATA Reserve
2020 DATA Reserve
2030 DATA Reserve
2040 DATA Reserve
2050 DATA Reserve
2060 DATA Reserve
2070 DATA Reserve
2080 DATA Reserve
2090 DATA Ego-Test,Reaktions-Test,Intelligenz-Test
2100 DATA Sex-Handbuch,Strip für Anfänger,Reserve
2110 DATA Bio-Rhythmen,Monats-Zyklus,Diaet-Plan

2120 REM
2130 REM *****
2140 REM *
2150 REM * Die folg. Aus-Definition *
2160 REM * der jeweiligen Bereiche *
2170 REM * mit '*' ist bei Tabellen *
2180 REM * unbedingt zu beachten! *
2190 REM *
2200 REM *****
2205 REM
2210 DATA .....
2220 DATA .....
2230 DATA .....
2240 DATA .....
2250 DATA .....
2260 DATA .....
2270 DATA .....
2280 DATA .....
2290 DATA .....

2300 DATA Software-User-Club,bzw. Webe-Soft,bzw. unbekannte Urheber
2310 DATA Software-User-Club,bzw. Webe-Soft,bzw. unbekannte Urheber
2320 DATA Software-User-Club,bzw. Webe-Soft,bzw. unbekannte Urheber
2330 DATA .....
2340 DATA .....
2350 DATA .....
2360 DATA .....
2370 DATA .....
2380 DATA .....
2390 DATA .....
2400 DATA .....
2410 DATA .....
2420 DATA Eingaben vom Anwender,Rechner MZ-700,Entspr. ext. Einheiten
2430 DATA Eingaben vom Anwender,Rechner MZ-700,Entspr. ext. Einheiten
2440 DATA Eingaben vom Anwender,Monatszyklus / Diaetplan,nicht realisiert

2450 DATA .....
2460 DATA .....
2470 DATA .....
2480 DATA .....
2490 DATA .....
2500 DATA .....
2510 DATA .....
2520 DATA .....
2530 DATA .....
2540 DATA .....
2550 DATA .....
2560 DATA .....
2570 DATA .....
2580 DATA .....
3000 REM
3010 REM :REM RAHMEN-MENUE

20 REM *
30 REM * Beisp. Upros 2.System-Stufe *
35 REM *
40 REM * (Hier Psy & IQ) *
50 REM *
50 REM *****
70 REM
100 CURSOR5,1
110 PRINT"Bitte waehlen Sie Ihre Funktion"
120 PRINT" Test - Funktionen"
130 REM
140 PRINT"-----"
150 REM
160 PRINT" 1 Persoenlichkeits-Test"
170 PRINT" 2 Intellegenz-Test"
180 REM
190 PRINT" 3 Reaktions-Test"
200 REM
210 PRINT" 4 Ende des Programms"
220 REM
230 ASS="A":GOSUB1000
240 IF(Asc("1")+Asc("4"))GOTO230
250 REM
260 IFAs="4"GOTO390
270 REM
280 IFAs="1"GOTO340
290 REM
300 IFAs="2"GOTO380
310 REM
320 IFAs="3"GOTO360
330 REM
331 REM *****
332 REM *
333 REM * Aufruf Upros 3.System-Stufe *
335 REM *
336 REM *****
337 REM
340 CHAIN "Psycho-Test"
350 REM
360 CHAIN "Reakt-Test"
370 REM
380 CHAIN "IQ-Test"
390 REM
400 CHAIN "AUTO RUN"

10 REM *****
11 REM *
14 REM * IQ-Test Upro der 3. Stufe *
15 REM * mit Upros der 4. Stufe *
20 REM * IQ-Text0 *
22 REM * IQ-Text0 *
24 REM * IQ-Text0 *
26 REM *
27 REM *****
30 REM
150 IFPs="4"GOTO950 :REM TEXT0 GELESEN
160 IFPs="5"GOTO1060 :REM TEXT1 GELESEN
170 IFPs="6"GOTO1160 :REM TEXT2 GELESEN
180 DIM Fqs(10,2) :REM FRAGEN-TAB.
190 DIM Bts(8,14)
200 DIM Vts(11,4) :REM VERGL.-TAB.
210 DIM Ats(10,4) :REM AUSW.-TAB.
220 DIM Nts(30) :REM NR. -TAB.
230 DIM Wf(5) :REM FAKTOR-TAB.
440 RESTORE 3830 :REM DATA POSITION
450 FORV=1TO11
460 FORV=1TO4
470 READVts(X,Y) :REM INIT VT-TAB.
480 NEXTV,X
890 GOSUB1250 :REM LESEN TEXTE

```

```

910 REM TEST-ABSCHNITT-1
920 REM *****
930 REM
940 CHAIN "IQ-Text0"
945 REM
950 TB=VAL(TIS)
960 TXs=LEFTS(TIS,2)+MIDS(TIS,3,2)+RIGHTS(TIS,2)
970 GOSUB1350 :REM FRAGE 1 - 10
980 REM :REM ANTW. 1 - 10
990 TE=VAL(TIS)
1000 GOSUB2590 :REM ZWISCHENZEIT
1010 REM
1020 REM TEST-ABSCHNITT-2
1030 REM *****
1040 REM
1050 CHAIN "IQ-Text1"
1055 REM :REM AUFR. 4. STUFE
1060 TB=VAL(TIS)
1070 GOSUB1770 :REM FRAGE 11 - 18
1080 REM :REM ANTW. 11 - 18
1090 TE=VAL(TIS)
1100 GOSUB2590 :REM ZWISCHENZEIT
1110 REM
1120 REM TEST-ABSCHNITT-3
1130 REM *****
1140 REM
1150 CHAIN "IQ-Text2"
1155 REM :REM AUFR. 4. STUFE
1160 TB=VAL(TIS)
1170 GOSUB1350 :REM FRAGE 19 - 28
1180 REM :REM ANTW. 19 - 28
1190 TE=VAL(TIS)
1200 GOSUB2590 :REM ZWISCHENZEIT
1210 REM
1220 GOSUB2800 :REM RESULTATE
1230 REM
1240 PSS="3" :REM PROG.SCHALTER
1245 CHAIN "IQ-AUTO"
1250 REM :REM AUF 7. EBENE
1255 REM
1260 REM EINLESEN TEXTE
1270 REM *****
1280 REM
1290 RESTORE 3810 :REM DATA POSITION
1300 FORX=1TO30
1310 READNTS(X) :REM ZEILEN-TAB.
1320 NEXTX
1330 RETURN
1340 REM
1350 REM ANBIETEN FRAGEN 1/3
1360 REM *****
1370 REM
1380 RETURN
1390 REM
1400 REM ANBIETEN FRAGEN 2
1410 REM *****
1420 REM
1430 RETURN
1440 REM
1450 REM ZWISCHENERGEBNIS ZEIT
1460 REM *****
1470 REM
1480 REM
1490 REM
1500 REM
1510 REM ANZEIGEN RESULTATE
1520 REM *****
1530 REM
1540 REM
1550 REM CONSOLE
1560 REM RESTORE
1570 REM RETURN

```

```

10 REM *****
20 REM *
30 REM * IQ-Text0 Upro der 4. Stufe *
40 REM *
50 REM *****
60 REM
70 DIM FQs(10,2)
80 PSS="4" :REM PROG.SCHALTER
90 REM
100 REM :REM TEXT ABSCHN-1
110 REM
120 FORW=1TO10
130 FORY=1TO2
140 READFQs(W,Y) :REM FRAGENTABELLE
150 NEXTY,W
160 FORW=1TO10
170 FORY=1TO4
180 READATs(W,Y) :REM ANWORTTABELLE
190 NEXTY,W
200 REM
210 CHAIN "IQ-Test"
220 REM :REM ZUR 3. STUFE
230 REM
240 REM *****
250 REM *
260 REM * Hier folgen die DATA'S *
270 REM *
280 REM *****
290 END

```

Oliver Roth und Andreas Jerrentrup haben die Grafik-Routine von Andre Kohley in Heft 2 ausführlich studiert. Sie wollten jedoch nicht nur Bilder in schwarz/weiß, sondern in Farbe abspeichern. Die folgenden Routinen dienen deshalb zum Abspeichern und Laden von ganzen Bildschirmseiten in vier Farben (Bildschirmmodus M1)

### SCREENSAVE:

```

10 LIMIT$BEO0
20 DIM C(3)
30 POKE$FE10,$DB,$EO,$JE,$I,$DJ,$CD,$CD,$J2,$FE,$DB,$E1,$21,$0,$DE,$11,$0,$BE,$1
,$0,$20,$ED,$E0,$DB,$E0,$JE,$2,$D3
40 POKE$FE2B,$CD,$CD,$J2,$FE,$DB,$E1,$C9,$21,$0,$80,$11,$0,$DE,$1,$0,$20,$ED,$80
,$C9
50 USR($FE10)
60 INPUT"BILDNAME:";A$
70 FOR L=0 TO 3
80 PRINT "PAL-Wert";L;";"
90 INPUT C(L)
100 NEXT
110 POKE$FE01,C(0),C(1),C(2),C(3)
    'In CO-C3 stehen die aktuellen Werte von PAL 0-3
120 A$="7SBEO0FE04BEO0;" + A$:USR($15B7,A$)
130 LIMITMAX

```

### SCREENLOAD:

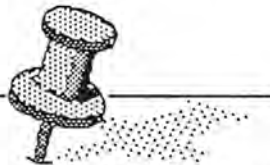
```

10 LIMIT$BEO0
20 CLS:INPUT"BILDNAME:";A$
30 A$="L:" + A$:USR($15B7,A$)
40 POKE$FE40,$DB,$EO,$JE,$2,$DJ,$CC,$CD,$62,$FE,$DB,$E1,$21,$0,$BE,$11,$0,$DE,$1
,$0,$20,$ED,$80,$DB,$E0,$JE,$1
50 POKE$FESA,$DJ,$CC,$CD,$62,$FE,$DB,$E1,$C9,$21,$0,$DE,$11,$0,$80,$1,$0,$20,$ED
,$80,$C9
60 USR($FE40)
70 PAL0,PEEK($FE01):PAL1,PEEK($FE02):PAL2,PEEK($FE03):PAL3,PEEK($FE04)
80 LIMITMAX

```

Nachdem das zu speichernde Bild erzeugt worden ist, sollte die Routine SCREENSAVE mit CHAIN aufgerufen werden. Es wird dann nach dem Bildnamen und den vier Farbwerten gefragt (PAL 0-3). Dann wird das Bild automatisch gespeichert.

SCREENLOAD braucht nur den Bildnamen, es erzeugt das Bild in den Farben, die beim Abspeichern mit SCREENSAVE eingegeben wurden.



## Wann schicken Sie uns Ihren Beitrag?

Auf vielfachen Wunsch würden wir gern eine Serie über die verschiedenen, für den MZ-800 lieferbaren *Programmiersprachen* bringen. Wer hat sie benützt und kann nützliche Informationen liefern, für welche Anwendungen sie sich am besten eignen?

Wer hat an seinem MZ-700/800 nützliche Hardware-Zusätze angebracht und kann darüber berichten?

Nutzen Sie Ihren Computer beruflich? Dann senden Sie uns bitte einen Anwenderbericht!

Vielen Dank im voraus! *Ihr MZ-Verlag Harald Schicke*

## Club-Porträt

Der **Software-User-Club** stellt sich vor:

Wir sind ein geräte-unabhängiger Computer-Club mit derzeit 30 Mitgliedern aus dem gesamten Bundesgebiet und Sitz in Karlsruhe. Unsere Mitglieder sind Schüler, Studenten und Lehrer gleichermaßen wie Ingenieure, Kaufleute und Techniker, Unternehmer und Hobbyisten.

Wir bieten den Kontakt zu Herstellern, Anwendern und Club's im Bundesgebiet und im europäischen Ausland; die organisatorische Voraussetzung für die Vereinverwaltung unter EDV-Einsatz (Mitglieder/Beitrag/Kasse/Protokoll); ein im zweimonatigen Turnus erscheinendes Club-Info als Erfahrungsaustausch; eine club-eigene Fachbücherei und eine umfangreiche Programmbibliothek.

Wir streben an die Eintragung ins Vereinsregister mit dem Status der Gemeinnützigkeit; die Herausgabe einer ordentlichen Clubzeitung; die Einrichtung einer Mailbox.

Wir suchen aktive und passive Mitglieder, Clubs, Vereinigungen oder Gruppen, die sich uns in Form von Aktionsgruppen anschließen wollen, ohne dabei ihre Selbständigkeit aufgeben zu müssen.

Beitragsklassen:

- A = Selbständige DM 80,- jährl. ①
- B = Unselbständ. DM 60,- jährl. ②
- C = Schüler/Stud. DM 50,- jährl. ③
- D = Gruppen p. P. DM 40,- jährl. ④
- E = Nurinfo-Bezie. DM 30,- jährl. ⑤

- ① + DM 40,- Aufnahmegebühr
- ② + DM 20,- Aufnahmegebühr
- ③ + DM 10,- Aufnahmegebühr

Club-Anschrift:  
Software-User-Club  
Walter Beinert  
Dornröschenweg 15  
7500 Karlsruhe 21 ■

## Sherlock-Holmes

Wir freuen uns sehr, daß Sie uns so eifrig schreiben. Normalerweise erhält jeder Einsender von Manuskripten, Frei- oder Prüfprogrammen noch am Tag des Eingangs eine Bestätigungskarte. Manchmal hören Sie danach nichts mehr... Das verursacht viele schriftliche oder telefonische Rückfragen. Meist ist des Rätsels Lösung ganz einfach. Deshalb beachten Sie bitte:

Wenn Sie uns schreiben, geben Sie bitte auf allen Unterlagen Ihren Namen und möglichst Ihre Anschrift sowie Ihre Telefonnummer an. Alle Unterlagen, d. h. zunächst mal der Brief selbst. Oft fehlt hier bereits die Adresse und Unterschriften sind nur schwer zu entziffern. Meist wird aber der Briefumschlag direkt beim Öffnen weggeschmissen... Aber auch auf den Listings, auf Kassetten, Disketten usw., kurz, auf allem, was Sie uns schicken, sollte Ihre Anschrift, zumindest aber Ihr Name vermerkt sein. Es macht uns eine Höllenarbeit, mit kriminalistischem Spürsinn, die Unterlagen zuzuordnen, wenn nichts genaueres vermerkt ist.

Deshalb unsere dringende Bitte: **Bevor Sie etwas an uns schicken, kontrollieren Sie doch, ob überall Ihr Name und Ihre Anschrift vermerkt sind.** Auch in Ihren Programmen sollte zumindest Ihr Name auftauchen. Schreiben Sie bitte auch in Ihrem Schreiben, zu *welchem Zweck* Sie uns etwas zuschicken, damit es keine Mißverständnisse gibt.

Sein Sie uns bitte auch nicht böse, wenn wir einige Zeit benötigen, um Ihre Programme zu testen. Je besser die Unterlagen, die Sie uns zuschicken, desto kürzer die Zeit bis zur Antwort. Zu jedem Programm gehört *unbedingt* eine Beschreibung und eine Angabe, welche Hardware benötigt wird. ■

## MZ-800

Den Drucker KX-P1080 von Panasonic habe über ein 26poliges Flachbandkabel angeschlossen. Am Drucker sind daher nur die Pins 1-13 und 19-31 auf dem 36poligen Centronicsstecker belegt. Die Leitungen am Druckerstecker Pin 10, 12 und 31 müssen aufgetrennt werden. Mit dieser Verbindung ist ein problemloser Druckbetrieb im MZ-800-BASIC möglich.

Claus Torstrick

\*\*\*

Mit INIT"CRT:M4" lassen sich am MZ-800 im 80er Modus vier Farben darstellen. Man muß sie nur mit PAL definieren, z.B. PAL1,4;PAL2,2;PAL3,6. Somit erhält man mit PRINT(1) die Farbe 4 (grün) usw.

Alfred Groiss

\*\*\*

POKE\$5D4,\$1 schaltet beim MZ-800 auf den zweiten Zeichensatz. Ein Beispiel:

```
10 POKE$5D4,$1:PRINT[2]
   "Text"
```

Mit POKE \$5D4,\$0 setzt man wieder auf normal.

Wenn man den zweiten Zeichensatz zusammen mit der SYMBOL-Anweisung benutzen will, geht man wie in der folgenden Programmzeile vor:

```
20 POKE$4DD0,$1:SYMBOL[1],
   10,10,"Text",2,2
```

POKE\$4DD0,\$0 schaltet wieder auf den ersten Zeichensatz zurück.

W. T. de Jong

\*\*\*

# Fragen & Antworten

## Fragen

### MZ-700 → 800

Bei der Benutzung der BASIC-Systeme MZ-700 und MZ-800 habe ich das Problem, daß ich MZ-700 Programme nicht mit dem MZ-800-BASIC laden kann. Mein Drucker, ein Panasonic KX-P1080, druckt im MZ-700-BASIC aber keine kleinen Buchstaben, dies funktioniert nur im MZ-800-BASIC nach der richtigen Initialisierung. Es dürfte auch für andere Anwendungen förderlich sein, alte MZ-700 Programme in das MZ-800-BASIC zu übertragen, um z. B. die Grafikmöglichkeiten zu nutzen.

Claus Torstrick, Niflandring 19,  
2000 Hamburg 56

Ihr Problem ist leicht lösbar sofern Sie über eine Quick-Disk verfügen. Benützen Sie das mitgelieferte Utility TRANS, um das Programm auf eine QD oder auf eine andere Kassette zu bringen. Danach läßt es sich leicht einladen. Nun können Sie die nötigen Änderungen vornehmen, damit es im 800er BASIC läuft. Wenn Sie es nur listen wollen, brauchen Sie gar nichts zu ändern.

☺☺☺

### 700er Spielprogramme auf MZ-800

Bei einigen Spielprogrammen für den MZ-700 gibt es mit dem MZ-800 das Problem, daß nur der Header der Spiele geladen wird. Danach läuft die ganze Kassette durch.

Claus Torstrick

Meist handelt es sich dabei um 700er Maschinensprachprogramme. Wenn Sie die DIP-Schalter auf der Rückseite des MZ-800 in der Stellung für MZ-700 lassen (entgegen der Bedienungsanleitung!) sollte alles klar gehen!

## Antwort

### Steuerung eines Schrittmotors mit dem MZ-800 unter Verwendung des Fischer Interface

Es ist ein Schrittmotor mit zwei getrennten Wicklungen zu verwenden, da sonst das Interface zerstört wird. Der Motor ist an zwei der vier Motorausgänge anzuschließen. Das nachfolgende Programm Schrittmotor spricht die Ausgänge M4 und M3 an. Falls zwei Schrittmotore betrieben werden sollen, wie dies z.B. beim Betrieb eines Plotters der Fall ist, muß das Programm auf die Ausgänge M2 und M1 ausgedehnt werden.

Das Programm fragt, wieviele Schritte der Motor machen soll. Eingabe einer negativen Zahl bewirkt umgekehrte Drehrichtung. Das Fischer Interface ist in einer Version für den MZ-800 erhältlich.

Dr. Albert Lorenz

```
300 X=SIZE:LIMIT $FE00: IF (X+2)=SIZE THEN 320
310 PRINT "FISCHER WURDE NOCH NICHT GESTARTET":LIMIT MAX:END
320 IN=$FE00:M1=$FE05:M2=$FE14:M3=$FE23:M4=$FE32:RE=0:EI=0:LI=5:AU=10
330 ER=$FE63:E1=$FEF1:E2=$FEF2:E3=$FEF3:E4=$FEF4:E5=$FEF5:E6=$FEF6:E7=$FEF7
340 EB=$FEF8:EX=$FEF9:EY=$FEFA:PX=$FEB9:PY=$FE92:USR(IN)
350 DIM A(4)
360 A(1)=M4+RE:A(3)=M4+LI
370 A(2)=M3+RE:A(4)=M3+LI
380 INPUT"S=":S
390 IF S<0 THEN I=4:S=-S:GOTO 440
400 FOR J=1 TO S
410 IF I=4 THEN I=0
420 I=I+1:USR(A(I))
430 NEXT J:GOTO 380
440 FOR J=1 TO S
450 I=I-1:USR(A(I))
460 IF I=1 THEN I=5
470 NEXT J:GOTO 380
```

### AIP

Was heißt AIP in AIP-Logo?

Jürgen Schulz,

Ostpreußendamm 36,  
1000 Berlin 45

## Fragen

### BASICODE

Wer hat Erfahrung mit BASICODE?  
Claus Torstrick

☺☺☺

### Video-Digitalisierer

Gibt es für den MZ-800 einen Video-Digitalisierer?

Tobias Keppler,

Alfred-Faust-Str. 73, 2800 Bremen

☺☺☺

### Fischer Technik Interface für MZ-800

Das Interface ermöglicht die Steuerung von Fischer Technik Modellen, aber auch z.B. von Schrittmotoren, wie das untenstehende Programmbeispiel von Dr. Albert Lorenz zeigt. Es wird einschließlich Software auf Kassette zum Preis von DM 259,- (Bestellnr. X 176) geliefert. Die Lieferzeit beträgt ca. vierzehn Tage. Der Aufpreis für die Floppy- (F 176) oder QD-Version (Q 176) beträgt DM 45,-.

### Hilfe!

Ich habe auf meinem Monitor Taxan Vision III statt 16 nur 8 Farben! Wie kann man im K&P-Diskbasic (800) den Tongenerator steuern, ähnlich im S-Basic mit POKE 4513,N:POKE 4514,N?

Georg Sassermann,

Hauptstr. 88, A-6471 Arzl/ Pitztal,  
☎0043/54123001



# Turbo-Lader für das 800er BASIC

Haben Sie sich auch schon über die langen Ladezeiten des 800er BASIC geärgert? Dann hat Christian Czech eine Lösung für Sie!

Vor allem die Anfänger, aber auch die Profis unter Ihnen werden das folgende Programm zu schätzen wissen.

Fast jeder wird sich schon einmal darüber geärgert haben: Ich spreche von der langen Ladezeit des 800er BASIC in der Kassettenversion (fast fünf Minuten). Dem Frust des öden Wartens, bis der Rechner endlich das erlösende "Beep" von sich gibt, kann ein Ende gesetzt werden: Alles was man dazu benötigt, ist ein SHARP MZ-800 mit Datenrekorder, etwas Zeit und Raffinesse.

Die Profis unter Ihnen mögen mir verzeihen, wenn ich etwas weiter aushole, denn der Grundgedanke des Programms soll ja auch für den Anfänger nachvollziehbar sein.

Der MZ-800 benutzt zum Abspeichern und Laden von Daten auf bzw. von Kassette die sogenannte Pulsweitenmodulation; dies bedeutet nichts anderes, als einen langen Impuls für ein gesetztes Bit (1) und einen kurzen Impuls für ein zurückgesetztes Bit (0).

Das Maschinenprogramm im ROM verwendet dafür vier Verzögerungsschleifen, die jeweils die Länge des Impulses bestimmen. Im Klartext: Kürzt man die Schleifenwerte im selben Verhältnis (!), so verkürzen sich die Impulse und somit auch die Dauer des Lade- bzw. Abspeichervorgangs. Obwohl der MZ-800 zwei Monitore im ROM verankert hat – 700er Monitor von 0000H - 0FFFH und 800er Monitor von E000H - FFFFH – werden immer dieselben Schleifen angesprochen. Sie befinden sich alle im 700er Monitor. Folgende Ver-

zögerungsschleifen sind ausschlaggebend (n=normal, v=verkürzt):

DELAY 1: Von 0759H - 075FH  
Schleifenwert bei 075AH: 1BH n  
09H v

DELAY 2: Von 0760H - 0766H  
Schleifenwert bei 0761H: 19H n  
08H v

DELAY 3: Von 09A9H - 09AFH  
Schleifenwert bei 09AAH: 73H n  
20H v

DELAY 4: Von 0A4AH - 0A4EH  
Schleifenwert bei 0A4BH: 52H n  
1BH v

Um nun die Ladezeit des 800er BASIC von fast fünf Minuten auf ca. zwei Minuten und 20 Sekunden (!) zu verkürzen, müssen folgende Punkte ausgeführt werden:

- 1 Kopieren des 700er Monitors vom ROM in das darunterliegende RAM (da die Schleifenwerte nur im RAM verändert werden können!)
- 2 Dritteln der vier Schleifenwerte
- 3 Laden des BASIC (das aber vorher mit höherer Aufzeichnungsrate abgespeichert werden muß!)

Die Punkte 1 und 2 übernimmt das Maschinenprogramm.

Nun geht man folgendermaßen vor: Geben Sie das untenstehende Maschinenprogramm im Monitor ein (aber bitte noch nicht starten!!) und speichern Sie es auf Kassette ab.

```
M BF00:3E 08 D3 CE 21 00 00 11 00 C0
      01 00 10 ED B0 D3 E0 21 00 C0
      11 00 00 01 00 10 ED B0 3E 09
      32 5A 07 3E 08 32 61 07 3E 26
      32 AA 09 3E 1B 32 4B 0A C3 7C E9
```

S...Filename: Turbo-Basic

Top ad.: BF00

End ad.: BF32

Exc. ad.: BF00

Für alle Wißbegierigen: Das Programm finden Sie auch noch einmal in Assemblerschreibweise mit Kommentar (S. 10).

Als nächstes verwenden Sie bitte die Kopieroutine im 800er Monitor (Anhang G im Bedienerhandbuch) und laden Sie damit das BASIC (mit G E807). Verändern Sie nun die Adresse BF30H: Statt C3 7C E9 nur noch C9 (=Return) und starten Sie jetzt das Programm mit G BF00. Nun steht der 700er Monitor im freien RAM mit den verkürzten Schleifenwerten (Vorsicht: Bei Betätigung der Reset-Taste befindet sich in dem erwähnten Bereich wieder das ROM mit den normalen Schleifenwerten!). Geben Sie nun G E80A ein und kopieren Sie das BASIC gleich nach dem vorher abgespeicherten Maschinenprogramm auf Kassette. Fertig!

Schalten Sie nun den System- schalter 1 auf der Rückseite des Rechners auf ON (für MZ-800-Modus), spulen Sie zurück und beginnen Sie – wie immer – den Ladevorgang des BASIC mit "C" und "PLAY". Der Rechner lädt jetzt zuerst das kurze Maschinenprogramm, startet dies und lädt gleich danach den BASIC-Interpreter, nur jetzt viel schneller!

Die gleich Prozedur können Sie auch mit dem "S-BASIC" durchführen, das dann nur noch eine Minute und dreißig Sekunden benötigt, um im Speicher zu stehen.

Übrigens: "LOAD errors" sind ebenso selten wie bei normalem Ladevorgang.

Christian Czech, Henkelstr. 33,  
8960 Kempten



Turbo-Basic Maschinenprogramm in Assemblerschreibweise (⇐ Seite 9)

```

ORG BF00H
LD A,08H ; Akku laden
OUT (CE),A ; Akku an Displaymoderegister
LD HL,0000H ; HL=Quelladresse
LD DE,C000H ; DE=Zieladresse
LD BC,1000H ; BC=Länge
LD IR ; übertragen
OUT (E0),A ; 0000H - 0FFFH ROM aus - RAM einbankten
LD HL,C000H ; ...
LD DE,0000H ; ...
LD BC,1000H ; ...
LD IR ; ...
LD A,09H > Schleifenwert 1 verändern
LD (075AH),A
LD A,08H > Schleifenwert 2 verändern
LD (0761H),A
LD A,26H > Schleifenwert 3 verändern
LD (09AAH),A
LD A,1B > Schleifenwert 4 verändern
LD (0A4BH),A
JP E97C ; Sprung zum IPL
    
```

\*\*\*

Ich weiß nicht, ob Ihr auch schon auf dieses Problem gestoßen seid, aber ich habe schon oft in einem Programm eine Variable benötigt, die ständig ihren Inhalt zwischen zwei Zahlenwerten wechselt (z.B. zwischen 0 und 1 oder -2 und 2 oder 123.54 und 0.22 usw.). Deshalb habe ich mir eine einfache Formel erstellt, die mir diesen Wunsch erfüllt:

## 1. Zahlengenerator

Das folgende Programm bewirkt, daß der Inhalt der Variablen V ständig zwischen den Zahlenwerten 1 und 2 wechselt.

```

10 INPUT "ZAHLENWERT 1 ";Z1
20 INPUT "ZAHLENWERT 2 ";Z2
30 V=Z2:Z=-(Z1+Z2)
100 V=-(Z+V)
110 PRINT V
120 GOTO 100
    
```

Man könnte dieses Problem auch mit einer IF...THEN-Anweisung lösen, dann würde die Zeile 100 wie folgt aussehen. Man sollte aber bedenken, daß eine IF...THEN-Anweisung viel Zeit vom

Prozessor in Anspruch nimmt und das Programm durch die längere Zeile einen höheren Speicherplatzbedarf hat.

```
100 IF V=Z1 THEN V=Z2:ELSE V=Z1
```

## 2. Abkürzen der IF...THEN-Anweisung

Die IF...THEN-Anweisung kann abgekürzt werden, indem man das THEN durch einen Doppelpunkt ersetzt bzw. in manchen Fällen ganz wegläßt! Beispiele:

```

10 IF Z<7 THEN PRINT A$ →10 IF Z<7:PRINT A$
20 IF Z=3 THEN GOSUB 200 →20 IF Z=3:GOSUB 200
30 IF Z<=0 THEN 40:ELSE END →30 IF Z<1:ELSE END
40 IF A>=0 THEN END:ELSE RUN →40 IF A<0:RUN
50 END →50 END
    
```

## 3. Gleichungen, Ungleichungen

```

10 INPUT "ZAHLENWERT 1, ZAHLENWERT 2 ";Z1,Z2
20 INPUT "ZAHLENWERT 1 = ZAHLENWERT 2 ";Z1=Z2
30 INPUT "ZAHLENWERT 1 <= ZAHLENWERT 2 ";Z1<=Z2
40 INPUT "ZAHLENWERT 1 < ZAHLENWERT 2 ";Z1<Z2
50 INPUT "ZAHLENWERT 1 <> ZAHLENWERT 2 ";Z1<>Z2
60 INPUT "ZAHLENWERT 1 > ZAHLENWERT 2 ";Z1>Z2
70 INPUT "ZAHLENWERT 1 >= ZAHLENWERT 2 ";Z1>=Z2
80 RUN
    
```

Wenn die Bedingung erfüllt wird (wahr ist), ist das Ergebnis -1, sonst 0! Damit kann man IF...THEN-Anweisungen sparen. Beispiel:

```

10 INPUT "ZAHL 2;Z
100 IF Z<100:Z1=34:ELSE Z1=0
110 PRINT Z1
120 RUN
    
```

Zeile 100 kann durch folgende Zeile ersetzt werden:

```
100 Z1=-(Z<100)*34
```

Da IF...THEN-Anweisungen in BASIC viel Zeit in Anspruch nehmen, ist es oft vorteilhaft, wenn man auf sie verzichten kann.

## 4. Videoramerweiterung

Wenn Ihr mal ein Programm schreiben wollt, in dem man wahlweise die Videoramerweiterung benutzen kann und Ihr dem User die lästige Abfrage, ob er seine Videoramerweiterung eingebaut hat, ersparen wollt, könnt Ihr mit einem einfachen PEEK-Befehl den Computer abfragen, ob diese eingebaut ist oder nicht.

```
10 IF PEEK(4249)=1;INIT"CRT:M2":ELSE INIT"CRT:M1"
    
```

→Seite 12

## Anmerkungen zu Heft 2

Seite 8 "Cursortzeitgeber": Verstehen das Programm nicht! Eine bessere Darstellung und etwas Erklärung täte not. Was soll "&"?

Seite 9 "Rundung": Dümmer geht es wohl nicht? Kann der Verfasser nicht in einem Schritt durch 1000 dividieren? "Basic-Compiler": Man sollte vor allem wissen, wo die Haken sind. Wenn Sie testen, lassen Sie z.B. mein Programm von S. 11 laufen. Warum kostet die Diskette DM 60,- mehr als die Kassette? (Anm. d. Red.: Wissen wir leider auch nicht!)

Seite 10 "Schrittmotor": Seit kurzem experimentiere ich mit gutem Erfolg mit den z.Z. sehr preisgünstigen Schrittmotoren. Hierzu lege ich ein Programm bei (S. 11). Auf diesem Gebiet ist noch vieles möglich. Es sieht so aus, daß sich bald die DM 250,- für das Interface einsparen lassen.

Seite 17: Mit dem K&P-Disketten-Basic funktionieren die Pokes nicht. Mit dem Kassetten-Basic geht es. Offenbar liegen die Adressen anders. Wie ist es bei dem Original-SHARP-Basic? (Anm. d. Red.: Funktioniert!) Ich verstehe nicht, was PRINT gegen PRINT pokes bedeuten soll.

Seite 19: Auf Ihren Maus-Test bin ich sehr gespannt. Erzielt man über den JOY-Stecker wirklich ein mit anderen Mäusen vergleichbares Resultat?

Albert Lorenz, 5000 Köln 90

\*\*\*

## Spitze!

Zuerst möchte ich sagen, daß ich es "spitze" finde, daß es endlich eine Zeitschrift extra für den SHARP MZ-700/800 gibt, und hoffe, daß sie sehr erfolgreich wird.

In dem Bericht von Edgar Lefgrün ab Seite 16 in der Mai-Ausgabe ist mir folgendes aufgefallen:

Er behauptet, daß die Farbwerte 8 bis 15 Wiederholungen der Farbwerte 0 bis 7 sind; das stimmt zwar,

jedoch sind die Farbwerte 8 bis 15 heller als die Farbwerte 0 bis 7! Ich vermute, daß er seinen MZ-800 an einen RGB-Monitor angeschlossen hat, der Helligkeitsunterschiede nicht wahrnehmen kann! Ein RGBI-Monitor (mit Intensitätsbit) oder ein Farbfernseher stellt alle 16 Farbwerte, entsprechend der Tabelle im Handbuch auf Seite 6-66 bzw. im Anhang A, dar!

Waldemar Schell

7950 Biberach/Riß

\*\*\*

## Kein PC-1600

Ich möchte Ihnen noch sagen, daß ich die ersten zwei Ausgaben von Magazin 700/800 sehr gut fand. Aber ich möchte auch demnächst auf Beiträge betreff den PC-1600 verzichten, da ich diese Zeitschrift als MZ-800 Besitzer abonniert habe.

Jetzt vielleicht noch einen Rat. Drückt niemals Listings mit mehr als ca. 4000 Byte ab, wie es bei Fischel (Alles über SHARP) gemacht wird, da man beim Abtippen meist die Lust verliert oder auch zuviele Fehler macht.

Ulrich Christ, 5900 Siegen

\*\*\*

## Machen Sie weiter so!

Nach zwei Heften auch von mir ein "Machen Sie weiter so!". Der Tip "RESET-Schalter" auf Seite 10, Heft 2/87 war bereits DM 6,- (7,50) wert!

Zu meinem Leserbrief (Heft 1). Das Farb-Thema war bereits überholt, da inzwischen der V-RAM-Zusatz (für DM 90,- bei K&K!) eingebaut war. Alle 16 Farben kommen nun gleichzeitig. Drei Leser hatten geantwortet. Leider damals noch nicht Hardware-Tip in Heft 2 gelesen!

Jürgen Schulz, 1000 Berlin 45

## Toll!

Ich finde das Magazin 700/800 wirklich toll und deshalb will auch ich mit meinen Tips an der Gestaltung des Hefts teilnehmen.

Ich stimme dem Leserbrief von Willi Plüschke vollkommen zu. Ein neuer MZ-800 könnte wie auf dem beigefügten Zettel aussehen.

Florian Krüsch

8960 Kempten



## Gratulation!

Wir haben kürzlich die zweite Ausgabe Ihres Magazin 700/800 erhalten. Zu diesem Magazin möchten wir Ihnen gratulieren, besonders zu der Idee mit den Freiprogrammen. Wir werden Sie in Zukunft mit eigenen Programmen unterstützen.

Frankenberger Computerclub,

Oliver Roth, 5900 Siegen

\*\*\*

Frank Diercks aus 2208 Glückstadt bemängelt, daß nur Beiträge von mindestens einer Seite honoriert werden. Das stimmt nicht: Alle veröffentlichten Beiträge außer Leserbriefen und sehr kurzen Tips werden honoriert!

\*\*\*

Fortsetzung von Seite 10:

Wenn die Speicherstelle 4249 bzw. 1099Hex eine 1 enthält, dann ist die Videoramenerweiterung eingebaut, sonst nicht.

Übrigens habe ich festgestellt, daß man auch mit einem Farbfernseher sehr gut im 80-Zeichen-Modus arbeiten kann, wenn man die Farbwerte schwarze Schrift auf grauem Hintergrund (PAL 0,8; PAL 3,0 im Modus 4 bzw. PAL 0,8; PAL 1,0 im Modus 3) einstellt.

Ich hoffe, daß Euch die Tips helfen werden.

**Waldemar Schell**  
Hermann-Volz-Str. 57  
7950 Biberach/Riß



Waldemar Schell ist der Autor unserer Programme *Skorpel*, Bestellnr. K 130, DM 24,80 und *Cycle*, Bestellnr. K 127, DM 19,80

\*\*\*

## Ausgabe der 512 Zeichen aus dem ROM (MZ-800)

```

10 ON ERROR GOTO 50'READ
   Error
20 DIM A(7):CLS:CURSOR0,22
30 A=$E000:LIMIT A
40 READ B:POKE A,B:A=A+1:
   GOTO 40
50 USR($E000)
60 FOR I=0 TO 511
70 FOR J=0 TO 7
80 X=PEEK($F000+J+8*I)
90 A(J)=128*(X AND 1)+32*
   (X AND 2)+8*(X AND 4)+2
   *(X AND 8)+(X AND 16)/2
   +(X AND 32)/8+(X AND 64)
   /32+(X AND 128)/128
100 NEXT J
110 POSITION 10*(I MOD
   32),10*INT(I/32)+8
120 PATTERN8,CHR$(A(7),
   A(6),A(5),A(4),A(3),
   A(2),A(1),A(0))
130 NEXT I:END
    
```

## Bewegung

Wie kann man einen Invader (gemäß unten stehendem Unterprogramm) in BASIC zum Laufen bringen?

**Jürgen Schulz,**  
Ostpreußendamm 36,  
1000 Berlin 45

(Sharp-Basic)

Unterprogramm

```

"Invader"
510 CURSOR 8,12
520 PRINT CHR$(204)+CHR$(200)+CHR$(204)
530 CURSOR 8,13
540 PRINT CHR$(200)+CHR$(200)+CHR$(200)
550 CURSOR 8,14
560 PRINT CHR$(193)+CHR$(216)+CHR$(213)
570 CURSOR 8,15
580 PRINT CHR$(216)+CHR$(32)+CHR$(216)
590 RETURN
    
```

## Joystickabfrage beim MZ-800 unter S-BASIC

Diese Routine von Jan Zamojski ist eine verbesserte Fassung der Hilfsroutine von Christian Fenske aus Heft 2/87.

Poke \$2FB0,\$F0,\$2F,\$21,0,  
\$30,\$F5,\$E6,\$0F,\$77,\$F1,\$E6  
,\$10,\$23,\$77,\$C9  
Als Assemblerlisting würde das so aussehen:

```

IN (A),F0 ;WERT VOM JOY-
           ;STICKPORT EINLE-
           ;SEN
CPL       ;komplementieren
LD HL,3000;Adresse zum Ab-
           ;legen der Werte
PUSH AF  ;und auf Stack
           ;retten
AND 0FH  ;Richtungsbits
    
```

```

140 DATA1,0,$10:' LD BC,
   1000H ;lang
150 DATA$21,0,$10:' LD HL,
   1000H ;von
160 DATA$11,0,$F0:' LD DE,
   F000H ;nach
170 DATA$DB,$E0:' IN
   (E0H),A ;Character-
   ROM einschalten
180 DATA$ED,$B0:' LDIR
           ;Dotmuster nach
           F000 übertragen
190 DATA$DB,$E1:' IN
   (E1H),A ;Ch-ROM aus
200 DATA$C9:' RET
    
```

Edgar Schatter ■

Task:

F	F	F
C	204	204
F	F	F
200	200	200
4	5	4
49	216	213
5	5P	5
216	32	216

```

;separieren
LD (HL),A ;und in $3000 ab-
           ;legen
POP AF    ;Wert restaurie-
           ;ren
AND 16H  ;Feuerknopf sepa-
           ;rieren
INC HL   ; Zeiger auf Ad-
           ;resse für
           ;Feuerknopfwert
LD (HL),A ;Zustand des Feu-
           ;erknopfes in
           ;$3001 ablegen
RET      ;das war's
Nach Aufruf mit USR($2FB0) steht
unter Adresse $3000 eine Zahl, die
die Richtung kennzeichnet, in die
der Joystick bewegt wurde, und
zwar: 1 für vor, 2 für zurück, 4 für
links, 8 für rechts, 5 für links vor, 6 für
links zurück, 9 für rechts vor, 10 für
rechts zurück. In Adresse $3001
steht entweder 0, falls der Feuer-
knopf nicht betätigt wurde odert
eine 16, falls doch. Ich glaube,
daß mein Vorschlag eine Berbes-
sereung bringt, weil nur noch zwei
Adressen (gegenüber fünf, die
auch keine Schrägbewegungen
erfassen, es sei denn, man bildet
Summen) abgefragt werden
müssen. In einem BASIC-Pro-
gramm kann man dann einfach
mit "ON PEEK($3000)GOTO" bzw.
"IF PEEK($3001) THEN" verzwei-
gen. Die Routine ist auch kürzer
und belegt keinen Speicherplatz
im BASIC-Bereich. ■
    
```

Eine der klassischen Anwendungen für Heimcomputer und PC's ist die Textbearbeitung. Sehr viele Anfragen, die wir dazu erhalten, beziehen sich darauf.

Wir wollen die Frage an Sie weitergeben: Welche Erfahrungen haben Sie mit der Textbearbeitung gemacht? Welche Programme benutzen Sie? Was gefällt Ihnen daran, was sollte verbessert werden? Welches Textprogramm ist für welchen Zweck geeignet?

Das ideale Textprogramm schlechthin wird es wohl nie geben. Zu unterschiedlich sind die Bedürfnisse der Anwender.

Der eine schreibt nur manchmal. Er kann sich keine komplizierten Befehle merken, die meist für ein exquisites Textprogramm nötig sind. Er möchte seine Texte am Bildschirm erstellen, damit er sie nachträglich kontrollieren und evtl. abspeichern kann. Die nachträgliche Kontrolle vor dem Ausdrucken findet er vielleicht so nützlich, weil er nur schlecht tippen kann.

Der andere möchte das Programm beruflich nützen. Da kommt es vor allem auf professionelle Details an, die dem Leser (leider) meistens nicht auffallen, weil er von Satzmaschinen schon sehr verwöhnt ist. Dazu gehört z.B. die automatische Silbentrennung (ein ganz schwieriges Problem in der deutschen Sprache!), aber auch die Möglichkeit, Text im Blocksatz und in Proportional-schrift auszugeben (auch sehr rechenintensiv!) sowie die Belegung von Funktionstasten mit beliebigen Floskeln oder schwierigen Wörtern, die häufig vorkommen. Statt das Wort einzugeben, drückt man auf die Funktionstaste. Einzelne Wörter oder gar ganze Absätze (Textbausteine) werden dann ausgegeben.

Das älteste professionelle Textprogramm, Wordstar, läuft auch auf dem MZ-800 (unter C/PM). Es läuft auch auf MS-DOS-Rechnern (IBM-Kompatible). Überall gelten die gleichen Befehle – aber sie sind lausig schwer zu merken. So schreibt uns z.B. Albert Lorenz in einem Brief: "Dieser Brief ist mit meinem super-simplen Textprogramm (9 Zeilen!) auf dem MZ-800 und Silverreed EX 32 geschrieben. Früher habe ich mich an Wordstar versucht. Ist für den Privat-Bedarf viel zu kompliziert. Oder das Textprogramm von K&P verlangt ein spezielles Interface! Glatt unbrauchbar." Dem Brief ist das einfache Textprogramm nicht anzusehen!

Dr. Werner Groß hat sich ebenfalls sein eigenes Textprogramm erstellt. Es heißt MZ-BRIEF und hat diverse Unterprogramme. Eins heißt TEXT-LESEN. Es steht Ihnen künftig auch auf unseren Freidisketten zur Verfügung.

Beim MZ-700/800 stellt sich natürlich auch die Frage nach der Druckausgabe. Hier ist noch viel Arbeit zu leisten, denn wir benötigen noch diverse Druckertreiber, um speziell die Probleme mit den Umlaufen zu lösen. Wenn Sie sich solche Druckertreiber erstellt haben, wäre es ganz ideal, Sie könnten Sie auch anderen Anwendern zur Verfügung stellen. Eigentlich sollte jedes Programm, das Drucker benötigt, in einem speziellen Programmteil Einstellmöglichkeiten für diverse Drucker bieten. Erst dann ist es anwendergerecht. Schließlich hat fast jeder einen anderen Drucker.

Deshalb noch einmal unsere Bitte: Schreiben Sie uns Ihre Erfahrungen mit den verschiedenen Textprogrammen oder senden Sie uns Ihre eigenen (sofern sie nicht zu lang sind) zur Veröffentlichung zu. ■

In vielen Haushalten gibt es heute einen Video-Rekorder. Der MZ-700/800 ist für den Hobby-Filmer ein nützliches Zubehör. Aus dem MZ-800 lassen sich Bild und Ton ganz einfach in den Videorekorder übernehmen.

Der MZ-Verlag Harald Schicke hat das bereits einmal vorge-macht: Mit seinem Verlagsprogramm auf VHS-Kassetten (Bestellnummer V 162, DM 25,-).

So wird's gemacht: Beim Bild ist es ganz einfach. Sie schließen den Videorekorder (Video in) am Computer dort an, wo Sie sonst den Fernseher anschließen. An den Videorekorder (Video out) schließen Sie den Fernseher an. Am Videoschraubchen an der Rückseite des MZ-800 und am Videorekorder stellen Sie die Frequenz richtig ein. Das ist natürlich ein bißchen Fummelei. Sie drehen so lange, bis Sie ein gutes Bild am Fernseher (der muß natürlich zuvor auf die Frequenz des Videorekorders eingestellt sein) sehen.

Beim Ton ist es etwas komplizierter. Dafür müssen wir den MZ-800 öffnen und die Kabel, die zum eingebauten Lautsprecher führen, nach außen bringen (Achtung! Durch den Eingriff kann die Garantie verloren gehen!). Dort kann man entweder einen Umschalter anbringen oder einfach einen für den Videorekorder passenden Stecker. Dieser wird in Audio in des Videorekorders gesteckt. Ohne Umschalter wird der Lautsprecher umgangen. Dann brauchen Sie künftig einen Kopfhörer, um den Ton aus dem MZ-800 zu hören. Mit Umschalter wechseln Sie einfach zwischen dem internen Lautsprecher und dem Ausgang für Kopfhörer bzw. Videorekorder. Nun stehen Ihnen viele Möglichkeiten der Nutzung zur Verfügung! MZ-800 und Video – eine gute Kombination! (Wer berichtet über Erfahrungen?) ■

# Konvertierung

**Kompatibel und Konvertierung sind Zauberworte der Computersprache. Denn sie ermöglichen es, Programme auf verschiedenen Computern zu benutzen (kompatibel) oder sie so umzuwandeln, daß sie benützbar sind (Konvertierung).**

Einige Anfragen zu diesem Thema haben uns veranlaßt, ein paar Möglichkeiten dafür aufzuzeigen.

SHARP hat darauf geachtet, daß Programme, die auf den Vorgängermodellen des MZ-800 lauffähig waren, auch auf dem MZ-800 zu starten sind. Dies wird durch das Konzept des Clean-Computers (sauberer Computer = Computer ohne eingebaute Programmiersprache) unterstützt. Was bedeutet das? Wenn man den MZ-800 startet, erkennt man zunächst nur die Nachteile dieses Konzepts. Es meldet sich IPL. Bevor man loslegen kann, muß erst eine Programmiersprache geladen werden.

Bei anderen Computern wird sofort das BASIC geladen, das im ROM (read only memory = Nur-Lese-Speicher) vorhanden ist. Das spart Zeit, wenn man nur dieses BASIC benötigt. Beim MZ-800 können aber auch leicht andere Programmiersprachen verwendet werden. Das hängt davon ab, welche Kassette oder Diskette wir einlegen.

So kann man in den MZ-800 auch z.B. das BASIC des MZ-700 (wird mitgeliefert) einladen oder das des MZ-80 K. Dann sind die BASIC-Programme dieser Rechner ohne Einschränkung lauffähig (=kompatibel).

Selbstverständlich gibt es dabei Einschränkungen: Die früheren Programme (und das BASIC) nutzen nicht die Möglichkeiten des MZ-800. MZ-800-Programme würden auch nicht auf den alten

Rechnern laufen (das nennt man aufwärtskompatibel). MZ-700 Programme kann man grundsätzlich laufen lassen, wenn man im Gegensatz zum Bedienerhandbuch den Rechner immer im MZ-700-Modus läßt. Die neueren Programme für den MZ-800 erkennen den Modus selbständig. Die 700er Programme natürlich nicht. Als die Programmierer tätig wurden, kannten sie noch nichts vom MZ-800.

Wenn man Programme, die man auf anderen Rechnern erstellt hat, an den MZ-800 anpassen möchte, so nennt man das Konvertierung.

Eine solche Konvertierung hat der MZ-800 eingebaut. So kann man im MZ-800 BASIC Programme einladen, die im BASIC des MZ-700, MZ-80A und MZ-80K geschrieben worden sind.

Manche 700er Programme lassen sich nicht im 800er BASIC laden. Dafür gibt es einen Trick. Benützen Sie das Hilfsprogramm TRANS, das Sie mit der Quick Disk erhalten haben. Dann lassen sich alle 700er Kassettenprogramme leicht im 800er BASIC einladen. Sie müssen dann nur einige spezifische Befehle ändern. Schon läuft das Programm.

Ebenso einfach lassen sich Programme vom MZ-80 K übernehmen. Auch Programme vom MZ-80 A lassen sich konvertieren. Die Mühe ist allerdings größer.

Ebenso wie beim Einladen von MZ-80 K BASIC-Programmen meldet der Rechner nach dem Laden ...converting...Doch dabei werden alle Befehle in REM's geändert. Jetzt müssen Sie diese Befehle umschreiben. Schon läuft das Programm auf Ihrem MZ-800!

Mit einem Hilfsprogramm gelingt Ihnen sogar noch mehr! Dafür gibt

es den **6502 Crossassembler** (Bestellnummer K 152, DM 110,-) bzw. Q 152, DM 125,-).

Mit diesem Crossassembler sind Sie in der Lage, auf Ihrem MZ-700/800 im 6502 Assemblercode zu programmieren. Dadurch können Sie z.B. Programme vom Apple, Commodore oder Atari auf Ihrem SHARP implementieren. Es werden sämtliche 6502 Befehle verarbeitet und direkt in Z-80 Maschinensprache übersetzt. Das bedeutet, daß eine besonders schnelle Ausführungszeit erreicht wird.

Zusätzlich zum Crossassembler ist ein Editor zur Erstellung und Bearbeitung von Programmen implementiert. Des weiteren bietet der Crossassembler auch Labels und Pseudobefehle. Die fertigen (kompilierten) 6502-Programme können auf Kassette bzw. Quick-Disk abgespeichert werden und sind später ohne den Assembler direkt vom Monitor oder IPL aus lauffähig.

Leider müssen Sie die Programme Byte für Byte per Hand eingeben, da die Speichermedien nicht kompatibel sind. Die Ausführung auf dem MZ-800 ist in der Regel auch langsamer als auf den 6502-Computern, da ja alles erst noch übersetzt werden muß.

Auch zu diesem Thema gehört der Hinweis auf Heft 1/87, Seite 14. Dort haben wir ein Utility (Dienstprogramm) abgedruckt, mit dem man Sprites aus Commodore 64-Listings auf dem MZ-800 nutzen kann.

In Kürze hoffen wir, Ihnen ein Programm vorstellen zu können, mit dem Bilder aus dem Commodore 64 direkt in den MZ-800 eingelesen werden können. ■

# Hardcopy-Programme für MZ-1P16

Ich ärgerte mich schon lange, daß ich eine Grafik, die ich auf dem Bildschirm erarbeitet hatte, mit dem Plotter MZ-1P16 nicht ausdrucken konnte. Deshalb habe ich die folgenden vier Programme geschrieben.

Programm 1 ist für die 40-Zeichen-Betriebsart (320x200 Bildschirmpunkte) in zwei Farben (Untergrundfarbe + 1 Farbe für Schrift und Grafik) gedacht.

```
60000 'Hardcopy 800/1
60010 '200*320 Punkte s/w
60020 PMODE GR:HSET
60030 FOR I=1 TO 16 STEP 2
60040 PMOVE 78-I,2+I
60050 PLINE 402+I,2+I,402+I,
-204-I,78-I,-204-I,78-
I,2+I
60060 NEXT I
60070 FOR X=0 TO 199
60080 PMOVE 80,0-X
60090 FOR Y=0 TO 319
60100 A = POINT(Y,X)
60110 IF A=0 THEN PMOVE
80+Y+1,0-X:ELSE PLINE
80+Y+1,0-X
60120 NEXT Y:NEXT X
```

Für den Ausdruck wird ca. eine dreiviertel Stunde benötigt. Nur der schwarze Stift wird benutzt.

Programm 2 (Hardcopy 800/2, wir zeigen nur die Änderungen gegenüber Programm 1) ist ebenfalls für 320x200 Bildschirmpunkte geschrieben, aber für ein vierfarbiges Bild. Dabei ist Palettencode 0=weiß, 1=blau, 2=rot, 3=schwarz.

```
60000 'Hardcopy 800/2
60010 '200*320 Punkte bunt
60110 IF A=0 THEN PMOVE
80+Y+1,0-X
60120 IF A=F GOTO 60160
60130 IF A=1 THEN PCOLOR 1
60140 IF A=2 THEN PCOLOR 3
60150 IF A=3 THEN PCOLOR 0
60160 IF A>0 THEN PLINE
80+Y+1,0-X
60170 F=A
60180 NEXT Y:NEXT X
```

Wieviel Zeit der Plotter für diesen Ausdruck benötigt, kann ich nicht genau sagen, da ich noch nie ein ganzes Bild ausgeplottet habe. Es hängt davon ab, wie oft die Farben wechseln. Ich glaube aber, daß die Zeit zwischen einer Stunde bei einem einfachen Bild und drei Stunden bei einem komplizierten Bild (z.B. ein Fraktal +S. 5, Heft 2/87) beträgt. Hierfür werden die Plotterstifte schwarz, rot und blau benötigt.

Programm 3 (Hardcopy 800/3) ist zum Ausdruck von Bildern in der Grafik-Betriebsart M3, das heißt 640x200 Bildschirmpunkte, in zwei Farben gedacht.

```
60000 'Hardcopy 800/3
60010 '200*640 Punkte s/w
60020 PMODE GR:HSET
60030 FOR I=1 TO 16 STEP 2
60040 PMOVE 138-I,2+I
60050 PLINE 342+I,2+I,342+I,
-644-I,138-I,-644-I,
138-I,2+I
60060 NEXT I
60070 FOR X=0 TO 639
60080 PMOVE 142,0-X
60090 FOR Y=199 TO 0 STEP -1
60100 A = POINT(X,Y)
60110 IF A=0 THEN PMOVE 138+
(200-Y),0-X:ELSE PLINE
138+(200-Y),0-X
60120 NEXT Y:NEXT X
```

Für den Ausdruck werden ca. anderthalb Stunden und der schwarze Plotterstift benötigt.

Programm 4 (Hardcopy 800/4; Änderungen zu 800/3) ist zum Ausdruck von Bildern in der Grafik-Betriebsart M4, das heißt 640x200 Bildschirmpunkte, in vier Farben gedacht.

```
60000 'Hardcopy 800/4
60010 '200*640 Punkte bunt
60110 IF A=0 THEN PMOVE
138+(200-Y),0-X
60120 IF A=F GOTO 60160
60130 IF A=1 THEN PCOLOR 1
60140 IF A=2 THEN PCOLOR 3
```

```
60150 IF A=3 THEN PCOLOR 0
60160 IF A>0 THEN PLINE
138+(200-Y),0-X
```

```
60170 F=A
```

```
60180 NEXT Y:NEXT X
```

Für den Ausdruck gilt das gleiche wie bei Programm 2, nur daß die Zeitspanne von zwei bis sechs Stunden gehen dürfte. Es ist zu beachten, daß im Bildschirmmodus M4 nur gearbeitet werden kann, wenn man die Grafikerweiterung MZ-1R25 hat.

Falls man ein Bild hat, in dem die Farben sehr oft in einer Bildpunktreihe wechseln, sollte man Programm 2 und 4 ändern. Versuchen Sie es einmal selbst oder schauen Sie in die neue Liste der Frei-Programme! Durch diese Änderungen wird das Bild wahrscheinlich erheblich schneller ausgeplottet.

Nun noch ein paar Tips zur Anwendung. Um das Bild eines Programms ausplotten zu können, muß man erst das passende Hardcopy-Programm mit MERGE hinzuladen. An passender Stelle des Programms, in dem das Bild vorkommt, muß GOSUB 60000 eingegeben werden. Außerdem muß man darauf achten, daß volle Plotterstifte im Plotter sind. Und nun viel Spaß beim Ausplotten!

Ulrich Christ, 5900 Siegen



Für den MZ-700 haben wir das Programm Screencopy (K 121, DM 49,80 bzw. Q 121, DM 59,80) im Programm. Screencopy ermöglicht eine farbige Bildschirmdrucke mit allen Zeichen (1. und 2. Zeichensatz) des MZ-700 auf den Plottern MZ-1P15/16 und CE-516P. Screencopy belegt den Speicherbereich EBE6 H bis FEFF H (K121) bzw. DBE0 H bis EFO0 H (Q 121). Es läuft nicht nur mit S-BASIC, sondern mit allen Programmiersprachen, die diesen Bereich nicht benutzen.

## LIMIT-BEFEHL

DER MAXIMAL ZULÄSSIGE WERT FÜR DEN LIMIT-BEFEHL BETRÄGT LAUT DEM HANDBUCH (6-99) \$FF00. MÖCHTE MAN NOCH EINIGE BYTES HERAUSPRESSEN, SO KANN DAS MIT POKE \$1080, \$FF,\$FF GEÄNDERT WERDEN. DAS ZURÜCKSETZEN GESCHIEHT MIT POKE \$1080,\$00,\$FF.

## ON ERROR GOTO-BEFEHL

BEI DEN ERPROBUNGEN VON BASIC-PROGRAMMEN, IN DENEN DIESER BEFEHL MEHRMALS VERWENDET WIRD, IST ES OFT ERFORDERLICH DEN BEFEHL UNWIRKSAM ZU SCHALTEN. DIES IST DURCH REM ODER APOSTROPH MÖGLICH. EINE ANDERE METHODE ERLEDIGT DAS MIT POKES.

- UNWIRKSAM DURCH  
POKE \$6956,\$C3,\$FE,\$67
- WIRKSAM DURCH  
POKE \$6956,\$CA,\$E4,\$64

## DER BASIC-ANFANG BEIM MZ-2Z046

DER BASIC-SPEICHER BEGINNT BEI ADRESSE \$A473. NACH NEW ON BEGINNT ER BEI \$A017.

## DIE STEPRATE BEIM MZ-2Z046

DIE LADEGESCHWINDIGKEIT DER FLOPPY KANN GEÄNDERT WERDEN.

- POKE \$37DF,\$0 NORMAL
  - POKE \$37DF,1 BIS 3 ERHÖHTE LADEGESCHWINDIGKEITEN
- AUSWIRKUNGEN AUF SCHREIB/LESEFEHLER HABE ICH NICHT UNTERSUCHT. FÜR DIESE INFORMATION DANKE ICH HERRN SCHNEIDER VON DER FIRMA MAGNUM SOFTWARE-PRODUKTION IN HAMBURG.

## DIE VIDIORAM-ERWEITERUNG BEIM MZ-2Z046

DURCH PRINT PEEK(\$1099) KANN FESTGESTELLT WERDEN, OB DIE VIDIORAM-ERWEITERUNG INSTALLIERT IST. IST DER WERT '0' SO IST KEINE ER-

WEITERUNG INSTALLIERT. IST DER WERT '1' SO IST DIE ERWEITERUNG INSTALLIERT.

## FREI NUTZBARE ADRESSEN IM BASIC-INTERPRETER MZ-2Z046

ES GIBT MINDESTENS ZWEI BEREICHE IM INTERPRETER, DIE DER OPERATOR RELATIV GEFÄHRLOS NUTZEN KANN.

- VON \$558B BIS \$57FF MIT 629 BYTES
- VON \$6787 BIS \$67A3 MIT 28 BYTES

ACHTUNG: EINIGE SOFTWAREHERSTELLER NUTZEN DIE BEREICHE ZUR ABLAGE VON MASCHINENPROGRAMMEN.

## UNTERDRÜCKEN DES PROGRAMMABBRUCHS DURCH SHIFT UND BREAK

- POKE \$0D23,\$E7 VERHINDERT DEN PROGRAMMABBRUCH.
- POKE \$0D23,\$E8 ERMÖGLICHT DEN PROGRAMMABBRUCH.

BEI DER VERHINDERUNG WIRD AUSSERDEM NOCH DAS ZEITWEILIGE STOPPEN VON LIST DURCH BREAK UNTERBUNDEN. DAS ENTGÜLTIGE ANHALTEN VON LIST MIT SHIFT UND BREAK IST MÖGLICH.

## UNTERBINDEN VON SHIFT UND HOME

- POKE \$0992,\$2E ERMÖGLICHT SHIFT UND HOME.
- POKE \$0992,\$C9 UNTERBINDET SHIFT UND HOME.

## VERHINDERN VON SHIFT UND CLEAR

- POKE \$061D,\$CD ERLAUBT SHIFT UND CLEAR.
- POKE \$061D,\$C9 VERBIETET SHIFT UND CLEAR.

## VERHINDERUNG DER CURSORSTEUERUNG NACH LINKS BEI INPUT

- POKE \$09FE,\$2A ERMÖGLICHT DIE LINKSSTEUERUNG.
- POKE \$09FE,\$C9 VERHINDERT DIE LINKSSTEUERUNG.

## BEEP BEI DER AUSGABE VON READY

- POKE \$587A,\$CD,87,\$67
- POKE \$6787,\$CD,\$56,\$6A,\$11,\$F0,\$63,\$C9

## DIR-BEFEHL

DIE FORM DIR #n,FDx DES BEFEHLS WIRD IN KEINEM MIR BEKANNTEN HANDBUCH ERWÄHNT. IM SHARP COMPUTER MAGAZIN NR.3 VERÖFFENTLICHTE HERR RAINER SCHÄFER VON DER FIRMA SDS COMPUTER SERVICE EIN DIRECTORY-PROGRAMM, IN DEM DIESER BEFEHL GENUTZT WURDE. NACHFOLGEND FINDEN SIE EINE WEITERENTWICKLUNG DES PROGRAMMS, DAS EIN DIRECTORY UNTER BASIC-KONTROLLE AUSGIBT, DIE FREIE DISKETTENKAPAZITÄT DURCH STRING-VERARBEITUNG ERMITTELT UND DER VARIABLEN A ZUWEIST.

```
10 WOPEN#1,'RAM:DIRECTORY'  
20 DIR#1,FD1  
40 ROPEN#1,'RAM:DIRECTORY'  
50 INPUT#1,J$  
60 IF EOF(#1) GOTO 90  
70 A=A+1  
80 GOTO 50  
90 CLOSE#1  
100 DIM J$(A-1)  
110 ROPEN#1,'RAM:DIRECTORY'  
120 INPUT#1,J$  
130 IF EOF(#1) GOTO 180  
131 IF J$(B)= GOTO 120  
140 PRINT J$(B)  
150 B=B+1  
170 GOTO 120  
180 CLOSE#1  
190 DELETE 'RAM:DIRECTORY'  
220 A=VAL(MID$(J$(0),20,3))  
*1000  
230 PRINT 'FD:','A;' BYTES  
FREE'  
IST KEIN RAM-BOARD VORHANDEN,  
SO KANN FDX: EINGESSETZT WERDEN.
```

## FREIE DISKETTENKAPAZITÄT

ES GIBT WAHRSCHEINLICH SCHON MEHRERE BEKANNTE METHODEN, DIE FREIE DISKET-



TENKAPAZITÄT ZU ERMITTELN.  
HIER STELLE ICH IHNEN EINE  
WEITERE VOR.

10 POKES1DB3, \$22, \$87, \$67,  
\$C9

20 DIRFD1

30 POKES1DB3, \$DF, \$16, \$21,  
\$40

40 A=PEEK(\$6788)\*256+PEEK  
(\$6787)\*1000

50 PRINT A; ' BYTES FREE'

ANSTELLE VON DIRFD1 KANN  
DIRFD2 BIS FD4 ODER DIRRAM  
VERWENDET WERDEN.

## PROZEDUR ZUR HERSTEL- LUNG EINES FORMATUTILI- TY 2X40 TRACKS

DEN BASIC-INTERPRETER MZ-  
2Z046 EINLADEN. MIT BYE IN  
DEN SOFTWARE-MONITOR GEHEN.  
MIT LB000:FD1:FDCOPY DAS  
SHARP-UTILITY EINLADEN. MIT  
\*MCC89 CR DIE SPEICHER-  
STELLE DES VERGLEICHSBYTES  
ANWÄHLEN. ES ERSCHEINT:  
CC89=23. MIT 28 CR DEN  
INHALT DER SPEICHERSTELLE  
ÄNDERN. SHIFT UND BREAK  
DRÜCKEN. DAMIT IST DIE UM-  
STELLUNG AUF 80 TRACKS  
VOLLZOGEN. MIT \*MC9B3 WÄH-  
LEN WIR DIE SPEICHERSTELLE  
FÜR DIE KAPAZITÄTSANGABE  
AN. ES ERSCHEINT: :C9B3=5F.  
WIR ÄNDERN AUF :C9B3=FF  
DER FOLGENDE VORGANG DIEN-  
T DER ÄNDERUNG DES MENÜS DES  
PROGRAMMS MIT \*MC84D CR  
DIE ERSTE ZU ÄNDERNDE SPEI-  
CHERSTELLE DER MENÜS ANWÄH-  
LEN. ES ERSCHEINT: C84D=FE.  
MIT 00 CR, 00 CR, 00 CR,  
00 CR, 00 CR DIE KOPIER-  
FUNKTION AUS DEM MENÜ HER-  
AUSNEHMEN, SHIFT UND BREAK  
DRÜCKEN.

DIE NÄCHSTEN ARBEITSGÄNGE  
DIENEN DER ÄNDERUNG DER  
MASKE. MIT \*MD1C7 CR DIE  
ERSTE SPEICHERSTELLE DER  
MASKENKOPFZEILE ANWÄHLEN.  
ES ERSCHEINT: D1C7=46. MIT  
20 CR, 46 CR, 4F CR, 52 CR,  
4D CR, 41 CR, 54 CR, 55 CR,  
54 CR, 49 CR, 4C CR, 49 CR,

54 CR, 59 CR, 20 CR, 32 CR,  
58 CR, 34 CR, 30 CR, 20 CR,  
54 CR, 52 CR, 41 CR, 43 CR,  
4B CR, 53 CR, 00 CR, 00 CR,  
00 CR, 00 CR, 00 CR, 00 CR,  
00 CR, 00 CR DIE KOPFZEILE  
ÄNDERN, SHIFT UND BREAK  
DRÜCKEN. MIT \*MD226 DIE  
ERSTE SPEICHERSTELLE DER  
ZEILE C...COPY DISK AN-  
WÄHLEN. ES ERSCHEINT:  
D226=43. MIT 20 CR, 20 CR,  
20 CR, 20 CR, 20 CR, 20 CR,  
20 CR, 20 CR, 20 CR, 20 CR,  
20 CR, 20 CR, 20 CR, 20 CR,  
20 CR DIE ZEILE MIT LEER-  
ZEICHEN FÜLLEN.

NUN WIRD EINE MASCHINENROU-  
TINE IN DEN SPEICHER GE-  
BRACHT, DIE SPÄTER DAS PRO-  
GRAMM IN SEINEN ABLAUFBE-  
REICH LÄDT. MIT \*MD3C4 CR  
DIE ERSTE SPEICHERSTELLE  
ANWÄHLEN. ES ERSCHEINT  
:D3C4=DF ODER :D3C4=00.  
MIT 21 CR, 00 CR, B0 CR,  
11 CR, 00 CR, 00 CR, 01 CR,  
00 CR, 24 CR, ED CR, B0 CR,  
C3 CR, AF CR, 22 CR DIE MA-  
SCHINENROUTINE EINGEBEN,  
SHIFT UND BREAK DRÜCKEN.

DER LETZTE VORGANG DIEN-  
T ZUM ABSPEICHERN DES PRO-  
GRAMMS. IM LAUFWERK MUSS  
EINE FORMATIERTE UND NICHT  
SCHREIBGESCHÜTZTE DISKETTE  
SEIN.

MIT \*SB000D400D3C4:FD1:FOR-  
MAT 2X40 TR. CR DAS UTILI-  
TY ABSPEICHERN.

GEHEN SIE MIT \*R ZURÜCK  
IN DEN BASIC-INTERPRETER.  
BEFEHLEN SIE RUN "FORMAT  
2X40 TR." CR UND ÜBERPRÜFEN  
SIE DAS PROGRAMM.

DER MZ-800 UND DER  
PRINTER/PLOTTER CE-516P  
ÄNDERUNG DES ZEICHENBEREI-  
CHES: WIRD MIT DEM MZ-800  
DER CE-516P ANSTELLE DES  
MZ-1P16 VERWANDT, STIMMEN  
DIE ANGABEN IM RECHNERHAND-  
BUCH (6-85) NICHT MEHR. UM  
DIE IM PLOTTER-HANDBUCH,  
S.64, ANGEGBENEN WEITEN ZU

ERREICHEN, MUSS DER BASIC-  
INTERPRETER GEÄNDERT WER-  
DEN.

POKESXXXX, \$E5, \$21, \$C0, \$3  
X-KOORDINATE MAX 960

POKESYYYY, \$E5, \$21, \$4A, \$5  
Y-KOORDINATE MIN -1354

DISK-BASIC MZ-2Z046

XXXX=A0AE - YYYY=A0A8 FÜR  
DAS QUICK-DISK-BASIC DÜRF-  
TE DAS GLEICHE ZUTREFFEN.

KASSETTEN-BASIC 1Z016

XXXX=A035 - YYYY=A02F

KASSETTEN-BASIC 1Z013

XXXX=4E80 - YYYY=4E6F

ORIGINALBYTEKOMBINATIONEN -  
XXXX, \$E5, \$21, \$E0, \$1

YYYY, \$E5, \$21, \$E7, \$3

DAS FOLGENDE PROGRAMMZEILE  
LÄSST DEN PLOTTER EIN RECH-  
TECK VON 194 x 267 MM GRÖS-  
SE ZEICHNEN.

10 PLINE 0,0,0,-1354,960,  
-1354,960,0,0,0

DER MZ-800 UND DER  
PRINTER/PLOTTER CE-516P  
DIE ANDEREN ZEICHENSÄT-  
ZE DES CE-516P

DURCH DAS UMSTELLEN DER  
DIP-SWITCHES AM PLOTTER  
WERDEN DREI WEITERE ZEI-  
CHENSÄTZE ZUGÄNGLICH, BEI  
DENEN DIE BUCHSTABEN e, t  
NICHT VERFORMT SIND, WIE  
BEIM MZ-ZEICHENSATZ.

- ZEICHENSATZ PC-1500:  
SWITCH 2+7 UNTEN, DER REST  
OBEN

- ZEICHENSATZ IBM/PC-5000:  
SWITCH 2+6 UNTEN, DER REST  
OBEN

- ZEICHENSATZ PC-1350/2500:  
SWITCH 2+6+7 UNTEN, DER  
REST OBEN

BEI DEN DREI GENANNTEN ZEI-  
CHENSÄTZEN ERFOLGT NACH  
DRUCKENDE ZWAR DER WAGEN-  
RÜCKLAUF, ABER KEIN PAPIER-  
VORSCHUB. WIRD ZUSÄTZLICH  
DER SWITCH 4 NACH UNTEN GE-  
STELLT, GESCHIEHT AUCH  
DIES. VOR DER VERWENDUNG  
DER ZUSÄTZLICHEN ZEICHEN-  
SÄTZE MUSS DIE RICHTIGE BE-  
TRIEBSART DES RECHNERS MIT

INIT"LPT:M0,S2" EINGESTELLT WERDEN, UM DEN RECHNER VOM SHARP- AUF DEN ASCII-CODE UMZUSTELLEN. DAS GILT EBENSOWEIL FÜR DIE BENUTZUNG ALS PLOTTER IN DIESEM MODUS, DA KLEINBUCHSTABEN ALS STEUERZEICHEN BENUTZT WERDEN, WIE AN DEM FOLGENDEN BEISPIEL GEZEIGT WIRD:

```
10 INIT"LPT:M0,S2"  
20 PRINT/P CHR$(27);"b"  
30 FOR A=0.4 TO 2.2 STEP  
   0.2  
40 PRINT/P"C400,-400,"  
   100/A;" ,0,360,1,0,";A  
50 NEXT A  
60 PRINT/P CHR$(27);"a"
```

DAS PROGRAMM LÄSST DEN PLOTTER DIE AUF SEITE 83 ABGEBILDETE FIGUR ZEICHNEN. DIE KLEINEN BUCHSTABEN ALS STEUERZEICHEN KÖNNEN DURCH IHREN ASCII-CODE ERSETZT WERDEN

```
20 PRINT/P CHR$(27);  
   CHR$(562)  
60 PRINT/P CHR$(27);  
   CHR$(561)
```

DIE ZEICHENSÄTZE KÖNNEN MIT DEM FOLGENDEN PROGRAMM GEDRUCKT WERDEN:

```
3 'INIT'LPT:M0,S0':'BEI  
  BEDARF  
4 'INIT'LPT:M0,S1':'BEI  
  BEDARF  
5 'INIT'LPT:M0,S2':'BEI  
  BEDARF  
6 'INIT'LPT:M0,S3':'BEI  
  BEDARF  
10 FOR I=0 TO 255  
20 PRINT/P CHR$(I);  
30 NEXT I  
40 PRINT/P:'ZURÜCK AN DEN  
   LINKEN RAND  
50 END
```

JE NACH INITIALISIERUNG WERDEN BEIM AUSDRUCKEN DER ZEICHENSÄTZE ZUSÄTZLICH KLEINERE BIS GRÖßERE UNTERSCHIEDE INNERHALB EINES SATZES ERKENNBAR, BESON-

DERS VON S0, S1, S3 ZU S2. DAS SEMIKOLON IN ZEILE 20 VERHINDERT DEN PAPIERVORSCHUB; STEHT AN DER STELLE EIN KOMMA, WIRD DER TABULATORABSTAND GESETZT. HINWEISE AUF DIE NUTZUNG DER ALTERNATIVEN ZEICHENSÄTZE ERHIELT ICH VON HERRN GERD KRAUS VON DER FIRMA BAVARIA-SOFT IN INGOLDSTADT MIT DEM RK-INFO NR. 2 UND HERRN DETLEF KORHON VON DER FIRMA SHARP ELECTRONICS IM SHARP COMPUTERMAGAZIN NR.1. ICH DANKE DAFÜR.

## Praxistest HBC-BASIC- Compiler

### Bemerkungen zum Extended Basic Compiler XBC S 1.0 von Edgar Lefgrün

Die nachfolgenden Angaben gelten für die FD-Version. Für andere Speichermedien mag entsprechendes gelten, es wurden aber keine Tests vorgenommen.

Als ich den XBC erhielt, nahm ich an, ich könne meine im Disk-Basic geschriebenen Programme kompilieren wie S-Basic-Programme mit dem S-Basic-Compiler. Das erwies sich als Irrtum. Der XBC kann die normalen Basic-Programme nicht verarbeiten.

Die Quelltexte müssen mit dem Zeileneditor geschrieben werden. Dabei muß die vom normalen Basic abweichende Satzbauweise beachtet werden. Die im Editor vergebenen Zeilennummern dienen zur Hilfe der Editor-Befehle wie bei CP/M. Sie sind nicht Bestandteil des Quelltextes. Bei ihnen kann also nicht nach Zeilennummern gesprungen wer-

den. Das geschieht nach Label. Der Quelltext kann in einer Struktur programmiert werden, die an Pascal erinnert. Der Befehlssatz ist wesentlich größer als der des normalen Basics. Er enthält z. B. die Schleifen REPEAT...UNTIL, WHILE...WEND, LOOP...END...LOOP, DO...END DO.

Den Anlaß zum Niederschreiben dieser Bemerkungen gaben die vielen Anregungen von Lesern beim MZ-Verlag. Sie hatten Probleme mit der Bedienung des Compilers. Die hatte ich zu Beginn auch und habe daraufhin nachfolgende Vorgehensweise entwickelt.

Nach dem Laden des XBC von der Diskette wird mit dem Editor der Quelltext geschrieben, mit 'JT' das einwandfreie Kompilieren getestet und mit 'J' ein Problemlauf durchgeführt. Dann wird der Quelltext mit XSF "NAME" auf der Diskette abgespeichert und mit 'D' der Editorinhalt getilgt. Anschließend wird das folgende Hilfsprogramm geschrieben.

```
1 DEFAULT "F" <CR>  
2 <Shift + Break>
```

Es wird mit 'J' gestartet und anschließend mit 'D' getilgt. Nun kann mit dem Kompilieren begonnen werden. Mit JSRF "NAME" wird das Laufzeitsystem auf der Diskette abgespeichert und mit XLF "NAME" der Quelltext von der Diskette in den Editor geladen. Dann wird mit 'JS' kompiliert und das OBJ-Programm als "NAMEC" auf der Diskette abgespeichert.

Einige Befehle sind aus einem mir unbekanntem Grund nun blockiert. Es ist anscheinend unmöglich, den Editor zu benutzen. Auf die Befehle 'XDF' und 'D' erzeugen nur Fehlermeldungen. Zur Abhilfe wird die Reset-Taste betätigt und der Compiler erneut geladen.

→Seite 19

Um Ihnen zu helfen, wenn Sie etwas suchen, kaufen oder verkaufen wollen, bieten wir Ihnen den Kleinanzeigenmarkt. Für bis zu fünf Zeilen zahlen Sie nur DM 10.--, Händler zahlen nur DM 20.--. Wenn Sie Gewerbetreibender sind, beachten Sie bitte, daß das aus Ihrer Kleinanzeige aus wettbewerbsrechtlichen Gründen deutlich hervorgehen muß. Jede Zeile darf bis zu 27 Anschläge haben. Bitte beachten Sie, daß Kleinanzeigen nur bei Vorkasse berücksichtigt werden können.

**Verkaufe Quick Disc für MZ-700.** Preis VS. Hans-Joachim Vorbeck, Bergstr. 17, 5340 Bad Honnef 1.  
☎ tagsüber: 02223/71561

**Suche Packet-Radio für MZ-700/800.** Georg Sassermaun, Hauptstr. 88, A-6471 Arzl/Pitztal, ☎0043/54123001

**Verkaufen DIN A4-Plotter CE-516P,** wenig gebraucht, für DM 578.-- (jetziger Neupreis: DM 698.--). MZ-Verlag Harald Schicke, Postfach, D-2110 Buchholz 5

## Freiprogramme

Es sind wieder sehr viele Freiprogramme eingegangen. Vielen Dank! Leider konnten sie bis Redaktionsschluß noch nicht alle katalogisiert werden. Dafür gibt es dann in Heft 4/87 (erscheint schon Ende August!) eine ganz lange Liste der neuesten Freiprogramme, die wir auch mit einigen Erläuterungen versehen wollen, wie von vielen gewünscht worden ist. Eine Erläuterung schon vorab: TTI heißt Tips, Tricks & Informationen. Sie stammen von Edgar Lefgrün, der die Freiprogramme zusammenstellt und dank seiner großen Erfahrung über einen riesigen Schatz an Tips verfügt. ■

Fortsetzung von Seite 18

Mit 'XFF' kann das Utility-Menü angewählt werden und die Files "NAME" und "NAMEC" auf eine andere Diskette kopiert und anschließend das Bootprogramm erzeugt werden. Im Prinzip steht dies alles in der Anleitung zum XBC. Möglicherweise ist sie aber nicht deutlich genug, anders kann ich mir die vielen Anfragen nicht erklären.

Verbesserungsvorschläge und Kritiken senden Sie bitte an den MZ-Verlag und/oder an:

Edgar Lefgrün, Torneiweg 3, 2400 Lübeck, ☎ 0451/36228.

XBC BASIC-Compiler, K 177 DM 199.--, Q 177 DM 214.--, F 177 DM 259.--. (K=Kassette, Q=Quick-Disk, F=5,25" Floppy-Disk). (Um weiteren bösen Briefen vorzubeugen: Wir wissen nicht, warum die Preise für die verschiedenen Speichermedien so unterschiedlich sind!) ■

## Grafisch orientierte Benutzeroberfläche

Neid ist keine nette Eigenschaft, doch wer von uns hätte nicht schon einmal mit tränenden Augen betrachtet, was es für andere Computer alles gibt. Wir MZ-700/800-Anwender können uns da nur schamvoll in die Ecke stellen und unser trauriges Schicksal beklagen, oder...?

Fragen wir doch noch einmal nach den Ursachen für diese bedauerliche Entwicklung. An der Hardware liegt es nicht. Unser MZ-731 und unsere MZ-800 laufen seit Jahren ohne Störung, manchmal sogar Tag und Nacht. Da gibt es keine Überhitzung. Auch die QD's funktionieren einwandfrei. Die Floppy arbeitet makellos.

Wir wissen, woran es liegt: Andere Hersteller haben ihre Geräte gepuscht. Sie haben für Zeitschriften, Bücher und Software gesorgt und auf diese Weise ihre Geräte massenhaft verkauft, selbst wenn die Hardware nicht so gebrauchstüchtig ist.

So gibt es von einem Wettbewerber allein 500.000 Computer im deutschen Markt. Von unseren Geräten gibt es nur wenige tausend Stück. Davon steht ein Großteil ungenutzt irgendwo im Keller. Die meisten Anwender haben aus lauter Verzweiflung den Rechner entweder in die Ecke gestellt oder sich im Programmieren geübt.

Deshalb gibt es z.B. auch noch keine grafisch orientierte Benutzeroberfläche für den MZ-800 oder viele andere Dinge. Für die paar Stück, die verkauft werden, lohnt sich die Entwicklung nicht (und wenn es entsprechendes trotzdem gibt, muß es leider teurer sein).

Dabei weist Jan Zamojski in einem Leserbrief an den MZ-Verlag darauf hin, daß sich der MZ-800 für so eine Benutzeroberfläche sehr gut eignen würde. Wörtlich schreibt er: "Schließlich ist der MZ-800 ein durchaus würdiger Gegenspieler des (ohnehin vielfach überschätzten) C64. Man denke nur an die Grafikfähigkeiten und den 80-Zeichen-Modus. Eine grafisch orientierte Benutzeroberfläche ließe sich beim 800er wesentlich einfacher implementieren als das 64er GEOS, wenn man von den Hardwarevoraussetzungen ausgeht (nur der ständige Interrupt des C64 fehlt, aber auch das wäre zu lösen)."

Er schlägt vor, so etwas gemeinsam zu entwickeln und im Magazin 700/800 vorzustellen. Er will gerne daran arbeiten. Wer macht mit? ■

# Input-Routine M2-700/800

## INPUT ist einer der ganz wichtigen Befehle in BASIC, denn welches Programm würde keine Eingabe des Benutzers verlangen?

Deshalb haben wir in der letzten Ausgabe des Magazin 700/800 eine INPUT-Routine von Andre Kohley veröffentlicht. Leider kann aber auch ein gutes und übersichtliches Programm seine Tücken haben, wenn man es einer strengen Prüfung unterzieht. Jan Zamojski hat ein Haar in der Suppe gefunden und zeigt mit seiner Routine, wie man es entfernen kann.

Abgesehen davon, daß die INPUT-Routine in Heft 2/87 keine Editierung mit DEL bzw. INST innerhalb des Eingabefeldes zuläßt, führt die DIM-Anweisung in Zeile 10030 spätestens beim zweiten Aufruf innerhalb eines Programms zu einem "Illegal data error", sofern der erste Eingabestring nicht zufälligerweise gleich der längste ist.

In Anwenderprogrammen steht man häufig vor dem Problem, vom Bediener eine Eingabe anzufordern, die eine bestimmte Länge nicht überschreiten und die womöglich nur aus bestimmten Zeichen bestehen darf (z.B. Zahleneingabe).

Der normale INPUT-Befehl löst das Problem nicht, denn hierbei kann der Bediener mit den Cursorstasten auf dem ganzen Bildschirm herumfahren oder einfach nur 'CR' drücken. Beides führt dazu, daß eine eventuell vorhandene Bildschirmmaske zerstört wird und daß sich Eingabefehler und dadurch verursachte Programmabstürze häufen.

Benötigt wird also die Möglichkeit, den Eingabevorgang auf einen vorher festgelegten Bildschirmbereich zu beschränken, allerdings unter Beibehaltung aller Editierungsmöglichkeiten, die auch beim INPUT zur Verfügung stehen, d.h. Cursorbewegung innerhalb des Eingabefeldes, DEL- und INST-Funktion usw. Die Eingabe soll durch 'CR' beendet werden, und zwar bei beliebiger Cursorposition innerhalb des Eingabefeldes (wie INPUT).

Von der nachfolgenden Routine werden alle diese Bedingungen erfüllt. Zusätzlich besteht die Möglichkeit, in das Eingabefeld einen Vorgabestring einzusetzen, der dann einfach mit 'CR' übernommen werden kann, oder das gesamte Eingabefeld mit SHIFT/CLR zu löschen. Ein weiterer Vorteil: Die Routine ist kurz und kann daher problemlos in alle Programme integriert werden.

In Heft 4/87 werden wir sehen, wie sich die Eingaberoutine so erweitern läßt, daß sie nicht nur eine sondern mehrere Eingabezeilen verarbeitet.

```
10 '
20 ' Listing 1
30 '
40 ' Programm zum Austesten
50 ' der Eingaberoutine
60 '
70 POKE $3000,$1A,$CD,$B6,$4,$12,$13,
  $10,$F8,$C9
80 '
90 ' LD A,(DE);CALL $04B6;LD (DE),A;INC
  DE;DJNZ -8;RET
100 ' Die POKE-Zeile muß am Anfang des
  Programms stehen (hier Zeile 70)
110 '
120 CLS:CURSOR 10,11:FOR I=1 TO 15:
  PRINT"-";:NEXT
130 VSS="TESTEINGABE"
140 EX=10:EY=10:LE=15:GOSUB 240
150 PRINT:PRINT" Eingabe: ";EI$:END
155 '
160 ' EINGABEROUTINE
170 ' EINGANGSPARAMETER:
180 ' EX - Eingabespalte
190 ' EY - Eingabezeile
200 ' LE - Eingabelänge
210 ' VSS - Vorgabestring (kann leer sein)
220 ' AUSGANGSPARAMETER:
230 ' EI$ - eingegebener Sting
235 '
240 CONSOLE EY,1,EX,LE+1
250 CURSOR EX,EY:PRINT LEFT$(VSS,LE);
260 EN=EX+LE-1:AN=53248+40*EY
270 IF PEEK(84)>EN THEN CURSOR EN,EY
280 COLOR PEEK(84),EY,1,7
290 GET T$:IF T$="" THEN 290
300 TA=ASC(T$)
310 IF TA=13 THEN 400
320 IF TA=22 THEN CURSOR EX,EY:PRINT
  SPC(LE);:CURSOR EX,EY:GOTO 280
330 IF (TA=20)+(TA=16) THEN IF
  PEEK(84)>380
340 IF TA=19 THEN 370
350 IF TA=24 THEN IF PEEK(AN+EN)=0 THEN
  380
360 IF TA<32 THEN 290
370 IF PEEK(84)>EN THEN PRINT CHR$(20);
380 COLOR PEEK(84),EY,7,1
390 PRINT T$;:GOTO 270
400 COLOR PEEK(84),EY,7,1:EI$="":FOR I=EX
  TO EN
410 EI$=EI$+CHR$(PEEK(AN+I)):NEXT
420 USR($3000,EI$)
430 IF EI$=SPC(LE) THEN 270
440 CONSOLE:RETURN
```