

MIKRO  
+ KLEIN

# COMPUTER



## **KLEINCOMPUTER aktuell**

INS 8073 – Basic on chip  
Computerneuheiten

## **HOBBY MIT MIKROS**

OSI-Board-Erweiterung  
Spezialzeichen auf Matrixdrucker

## **GEWUSST WIE**

Schnelle Grafik auf Sorcerer  
Pascal-Spielprogramm  
V24 für Kassettenrekorder



**sfr. 1.484,-\***

## **Kennen Sie einen vergleichbaren Teleprinter zu einem vergleichbaren Preis?**

Er hat mehr Funktionen als jeder andere Teleprinter seiner Klasse (45 Zeichen/Sek.).

Er ist mit nahezu jedem Computersystem voll kompatibel.

Er ist absolut zuverlässig.

Er ist extrem einfach zu bedienen (Schreibmaschinen-Design).

Er ist jederzeit verfügbar.

Er kostet sfr. 1.484 im Rahmenvertrag\*.

Er heißt LA34 AA . DECwriter IV.

Er kommt von einem der führenden Hersteller (Jahresproduktion: über 300.000 Terminals). Und das sagt eigentlich alles.

### **Technische Details:**

- Echte Durchschnitts-Geschwindigkeit von 30 Zeichen/Sek.
- Aufholgeschwindigkeit: 45 Zeichen/Sek.
- Hohe Druckqualität durch 7x9 Punkt-Matrix.
- Kompletter ASCII-Satz mit 128 Zeichen.
- 5, 6, 6.6, 8.2, 10, 12, 13.2 oder 16.5 Zeichen/Zoll.

- Bis zu 132 Zeichen bei normaler Zeichendichte.
- Bis zu 218 Zeichen bei enger Zeichendichte.
- Einstellbare Zeilenschaltung: 2, 3, 4, 6, 8, oder 12 Zeilen/Zoll.
- Einzelblätter, Papierrollen oder Endlosformulare auf Normalpapier.
- Leicht auswechselbare Farbbandkassette.
- Gewicht: nur 10 kg.
- Tragtasche erhältlich.

Mich interessiert der LA34 AA  
 Bitte schicken Sie mir weitere Unterlagen.

Name: \_\_\_\_\_

Firma: \_\_\_\_\_

Abt.: \_\_\_\_\_

Anschrift: \_\_\_\_\_

Telefon: \_\_\_\_\_

Dieter Hohl

Digital Equipment Corporation AG

Schaffhauserstrasse 144, 8302 Kloten/ZH MKC

**digital**

Digital Equipment Corporation AG

**So denkt die Zukunft.**



bitte  
frankieren

Verlag SCC AG  
Mikro- und Kleincomputer  
Seeburgstrasse 12

6002 Luzern

bitte  
frankieren

Dialog Computer Treuhand AG  
«Kurswesen»  
Seeburgstrasse 18

6002 Luzern

bitte  
frankieren

Name

Vorname

Firma oder Beruf

Strasse

PLZ/Ort

Telefon P

G

Verlag SCC AG  
Mikro- und Kleincomputer  
Seeburgstrasse 12

6002 Luzern

**Auflage 10000 Exemplare**

☎ **041 - 31 45 45**

**Mit einem Inserat erreichen Sie mehr als 10 000 interessierte und engagierte Personen – direkt zu Hause!**

# 81-2



April 1981  
Erscheint 6mal pro Jahr  
3. Jahrgang

Die Fachzeitschrift für «Personal Computing» informiert über Heimcomputer, Mikrocomputer für Hobby und Beruf, programmierbare Taschenrechner und Kleincomputer für «Small Business»

Offizielles Organ des  
**Schweizer Computer Club**  
6002 Luzern PC 60-26496  
Jahresabonnement  
Schweiz: Fr. 36.– plus einmaliger  
Clubbeitritt Fr. 20.– (Firmen Fr. 50.–)  
Ausland (inkl. Porto): Fr. 44.–

#### Redaktion

Leopold Asböck  
Ernst Erb  
Eric Hubacher, El. Ing. HTL  
Dr. Bruno Stanek

Nachdruck, auch auszugsweise sowie Vervielfältigungen jedwelcher Art nur mit schriftlicher Genehmigung des Verlags und unter voller Quellenangabe.

#### Manuskripte

Mit der Annahme von Manuskripten hat der Verlag das Recht zum Abdruck in seinen Organen und zur Übersetzung in andere Sprachen erworben. Für die Veröffentlichung wird keine Gewähr oder Garantie übernommen, auch nicht dafür, dass die verwendeten Schaltungen, Firmennamen und Warenbezeichnungen usw. frei von Schutzrechten Dritter sind. Die Verwendung der Informationen erfolgt auf eigenes Risiko. Mit Verfassernamen gekennzeichnete Beiträge geben nicht unbedingt die Meinung der Redaktion wieder. Copyright by SCC Lucerne, aber Speicherung in Datenverarbeitungsanlagen für den eigenen Gebrauch erlaubt.

#### Verlag, Redaktion, Inserate

Verlag SCC AG  
Seeburgstrasse 12, 6006 Luzern  
Tel. 041 - 31 45 45, Tx 72227 (dcl ch)  
Postcheck-Konten:  
Luzern 60 - 27181  
Stuttgart 3786 - 709

#### Verlagsleitung

Hans-Jürgen Ottenbacher

#### Herausgeber

Ernst Erb, 6045 Meggen

## INHALT

	Editorial	5
<b>KLEINCOMPUTER AKTUELL</b>	INS 8073 – Basic on chip	7
	PET jetzt mit PASCAL	11
	Computerneuheiten	15
<b>SMALL BUSINESS</b>	Recht und Computerbuchhaltung	17
<b>LEHRGÄNGE</b>	Turtlegraphics	21
	Plotroutinen für Typenradprinter	23
<b>PPC</b>	Textverarbeitung mit PCC?	29
	Neue Befehle für HP 41	33
	PPC-Plotter	37
<b>HOBBY MIT MIKROS</b>	OSI-Board-Erweiterung	43
	2m Contest	45
	SORCERER-Tips	47
	Spezialzeichen auf Matrixdrucker	49
	Joystik-Programmierung	51
<b>GEWUSST WIE</b>	Toolkit, 3. Teil	53
	Schnelle Grafik auf Sorcerer	59
	Pascal-Spielprogramm	61
	V24 für Kassettenrekorder	63
	Eigene BASIC-Befehle	67
<b>LISTINGS</b>	Programme zum Eintippen	69
<b>Clubinformationen</b>		73
<b>News... News...</b>	Aktuelle Meldungen aus der Welt der Mikros und Kleincomputer	75
<b>Vorschau</b>		82

# Programmier-Kurse

## BASIC-Schnupperkurs

(Kurs Nr. 124) Samstag, 16. Mai  
Kurskosten Fr. 70.-/Mitglied Fr. 50.-

## BASIC-Fortsetzungskurs

(Kurs Nr. 319) Donnerstag/Freitag, 11. - 12. Juni  
Kurskosten Fr. 340.-/Mitglied Fr. 310.-

## Programmiertechniken für kommerzielle Anwendungen

(Kurs Nr. 417) Montag - Mittwoch, 22. - 24. Juni  
Kurskosten Fr. 360.-/Mitglied Fr. 330.-  
Tageskurse jeweils von 9.00 - 12.00 und 13.30 - 17.00 Uhr

**NEU!** Teilnehmerzahl beschränkt!

# SUPERBRAIN-Workshop

für Erstanwender

Technologie - Datenträger - Peripherie - Betriebssystem -  
Tips & Tricks - Diskussion mit Profis  
(Kurs Nr. 901) Montag, 27. April, 13.30 - ca. 17.00 Uhr  
(Kurs Nr. 902) Montag, 25. Mai, 13.30 - ca. 17.00 Uhr  
Kurskosten: Für Besitzer von **DCT-SUPERBRAIN** ist der erste Workshop für eine Person **gratis**, jeder weitere Teilnehmer Fr. 75.-  
Andere Superbrain-Besitzer Fr. 150.- dito, jedoch SCC-Mitglied Fr. 125.-



## Schweizer Computer Club

Sekretariat Kurswesen  
Seeburgstrasse 18, 6002 Luzern  
Telefon 041 - 31 45 45

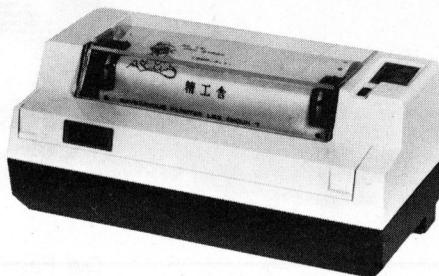
Kursanmeldung mit Karte auf Seite 1.

# SEIKOSHA

## Matrixdrucker

Fr. 1170.-

inkl. WUST



### Spezifikationen:

30 Charakter/Sek.  
80 Zeichen/Zeile  
5X7 Matrix  
128er Charaktersatz  
Spreizschrift  
Graphics  
Normalpapier  
(max. 3 Kopien)  
Tractorführung

### Anschlussmöglichkeiten

RS-232 C/Current Loop  
PET 2001  
APPLE II  
IEEE-488 (HP)  
PC-8001 (NEC)  
TRS-80 (TANDY)  
HITACHI LEVE III

### Dimensionen

127X328X171 mm

### Gewicht:

2,5 kg

**KONTRON AG**  
**DATASYSTEMS**

8048 Zürich, Bernerstrasse-Süd 169  
Telefon 01-62 82 82, Telex 57439  
1066 Epalinges, 10, ch. des Croisettes  
Téléphone 021-33 15 35, Télex 26 398

**[Für Lösungen mit Flair]**

# LOG-ON COMPUTER

## Sinnvolle Computer-Systeme für den Kleinbetrieb

- Finanzbuchhaltung
- Debitorenbuchhaltung
- Lagerverwaltung
- Adressverwaltung
- Textverarbeitung usw.



zum Beispiel:

komplette Anlage inkl. Massen-Speicher, Drucker,  
Buchhaltungsprogramm und Garantie Fr. 9900.-

## LOGON AG

Baslerstrasse 145 8048 Zürich, 01 62 59 22  
Konsumstrasse 1 8630 Rüti ZH, 055 31 72 30

## Machen Sie Ihr Lieblingshobby zum Beruf

Wir sind ein erfolgreiches Generalunternehmen für Datenverarbeitung. Wir verkaufen unseren Kunden schlüsselfertige EDV-Applikationen mit Hard- und Software.

Da wir uns auf Expansionskurs befinden, suchen wir für die Programmierung unser Dialogcomputer in Business-Basic einige interessierte

## Programmierer und Analytiker/Programmierer

Eventuell wären wir auch bereit, geeignete Anfänger auszubilden.

Sind Sie dabei? Wenn ja, rufen Sie uns an oder schreiben Sie uns kurz.

### Sengstag AG

Generalunternehmung für Datenverarbeitung

Haldenstrasse 1, 8344 Adetswil  
Tel. 01/939 18 28

# Editorial

Lieber Computerfreund

Aktivität, Kreativität, Durchhaltewillen und sachliches Leistungsvermögen machen den beruflich Erfolgreichen aus - hört man immer wieder sagen. Ich meine, dass diese Leistungsmerkmale für einen erfolgreichen Sachbearbeiter wohl in hohem Masse vorhanden sein sollten. Für ein erfolgreiches Teamwork, für Kaderstellungen, für das Zusammenspiel von Faktoren, welche eine Karriere ausmachen, sind jedoch noch ganz andere Charaktermerkmale wichtig.

Dies gilt auch für eine erfolgreiche Organisation oder Firma als Gesamtes. Sie sind schwer zu fassen und fussen auf Grundwerten, die ein Mensch erhalten hat - nicht auf seiner Aus- (und Ein-)Bildung. Es hängt von einer gewissen Neurosefreiheit ab - die in unserer heutigen Leistungsgesellschaft schwer zu finden ist - ob ein Mensch diese Faktoren mitbringt oder in jahrelanger (jahrzehntelanger) Arbeit an sich selbst bilden kann. Gute oder ausserordentliche Leistungen sollten weder dem lieben Geld, noch dem Prestige, noch einem Fanatismus entspringen, sondern der Freude an der sachlichen Leistung; aus Freude, etwas Gutes entstehen zu sehen oder dem Ziel einer gewissen Unabhängigkeit.

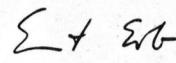
Warum solche Gedanken in einer Computerfachzeitschrift? Weil hier sonst immer das Rationale vorherrschen muss - denn dies ist tatsächlich die Grundvoraussetzung. Vergessen wird darob aber gerne, dass die weiteren, hintergründigen (nicht so rasch testbaren oder sichtbaren) Werte noch mehr zählen. Haben Sie sich schon Gedanken gemacht, warum die Japaner mehr Autos produzieren als die USA, warum USA-Giganten den Videomarkt kampflos preisgaben, warum die Chips aus Japan um ganze Grössenordnungen zuverlässiger sind? Sollten wir das Gute, das Zuverlässige nicht mehr vermehrt auf unserem Kontinent suchen - auch was die Wirtschaft betrifft?

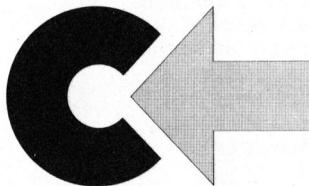
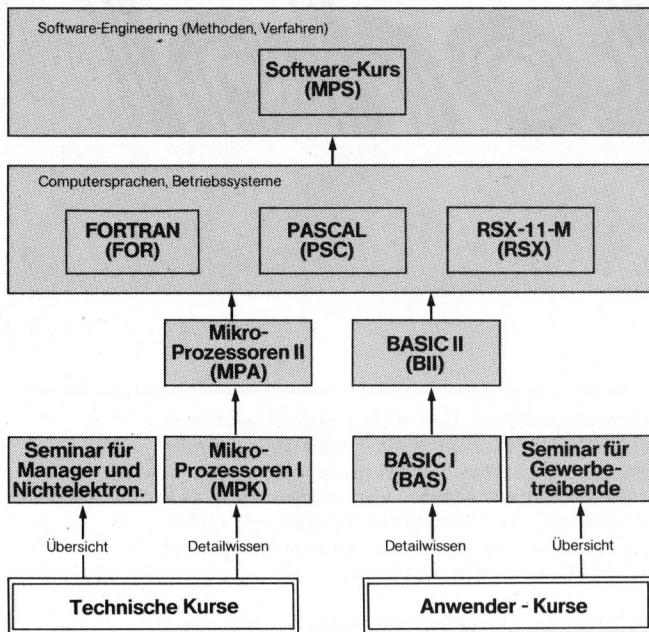
Im kleinen beschäftigt uns, warum in unserer neuen Branche so viele Produkte oder Dienstleistungen angeboten werden, bevor sie geboren sind. Kürzlich wurde eine "Vollgrafik" auf Superbrain angeboten. Erstens heisst dieses Wort nichts, denn Sie können auf jedem Bildschirm "Vollgrafik" zeigen und beim Superbrain besonders mit den "OUT's". Zweitens konnten von jener Firma seit Wochen nicht einmal die Darstellungen mit diesen "OUT's" gezeigt werden, geschweige denn die Umlaute auf dem Bildschirm. Trotzdem: Der SCC hat an der DIDACTA in Basel den DCT-Superbrain mit hochauflösender Grafik gezeigt - ohne sie vorher gross anzukündigen.

Es ist doch schon schwierig genug, in dieser Branche Personen zu finden, die obige Merkmale einigermassen erfüllen und dazu noch Sachkenntnis mitbringen, sodass man nicht noch Wunschdenken zu entwickeln braucht, um den Markt künstlich anzuheizen und extreme Forderungen stellen zu lassen. Wir haben einen Ruf zu verlieren - auch wenn wir ebenfalls dem Spruch unterliegen: "Wo viel Sonne ist, da ist auch Schatten".

Uebrigens suchen wir nach wie vor einen geeigneten Chefredaktor, der nicht unbedingt viel selbst zu schreiben braucht, aber die Entwicklungen überblicken, unsere vielen (potentiellen) Autoren anregen und koordinieren kann und einiges oben erwähntes mitbringt. Nicht nur das gute Salär und die "Plattform" sollen ihn ansprechen, sondern ebenso sehr das Gefühl, Interessantes aufgebaut zu haben.

Viel Erfolg mit Computer wünscht Ihnen im Namen der Redaktion

  
Ernst Erb



## Computerschule Zürich Digicomp AG

Birmensdorferstr. 94  
8003 Zürich  
Tel. 01/66 12 13  
Telex 812035

## Dabei sein . . .

setzt eine entsprechende Ausbildung voraus. Unsere Lehrkräfte zeigen Ihnen gerne, was Mikroprozessoren und Mikrocomputer sind, was sie können, wo sie eingesetzt werden sollten, wo Probleme liegen. Unsere Kursangebote ist strukturiert: Neben Grundkursen (MBK, BAS) führen wir laufend diverse Aufbaukurse durch. Die Kursdaten teilen wir Ihnen gerne mit.

### Technische Kurse

- **Mikroprozessoren I (MPA):** Fachkurs für Elektroniker (14 Abende oder 5 Tage). Voraussetzung: Digitaltechnik
- **Mikroprozessoren II (MPA):** Fortsetzungskurs zu MPK (14 Abende oder 4 Tage). Voraussetzung: Kenntnisse entsprechend Grundkurs
- **Seminar für Manager und Nichtelektroniker (MMA):** 1-Tages-Seminar, Orientierung über Mikroprozessoren

### Anwender-Kurse

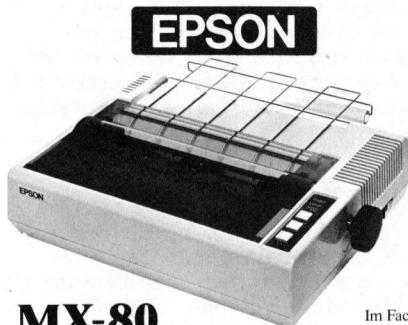
- **BASIC-I (BAS):** Einführungskurs für EDV-Anfänger. Erlernung der Programmiersprache BASIC.
- **BASIC-II (BII):** Fortgeschrittenen-Kurs für Anwender, welche BASIC bereits gut kennen.
- **Seminar für Gewerbetreibende (SKC):** Orientierung über Möglichkeiten des praktischen Einsatzes heutiger Kleincomputer

### Fortgeschrittenen-Kurse

(setzen Vorkenntnisse voraus, nicht für Anfänger geeignet, Englischkenntnisse vorteilhaft)

- **PASCAL (PSC):** Programmiersprache PASCAL in Theorie und Praxis
- **FORTTRAN (FOR):** Programmiersprache FORTRAN für technische Anwender
- **RSX-11-M (RSX):** PDP-11-Betriebssystem RSX-11-M für System-Spezialisten
- **Software-Kurs (MPS):** Software-Engineering, Software-Verfahren, Methodik, Organisation (für den fortgeschrittenen Anwender resp. Programmierer)

Fordern Sie unser Kursprogramm, Detail-Beschreibungen obiger Kurse sowie Anmeldekarten bei unserem Sekretariat an (Tel. 01 66 12 13).



**MX-80**

Im Fachhandel erhältlich

**Vergleichen Sie selbst ...**

**Welcher andere Drucker bietet Ihnen:**

- |  |   |
|--|---|
| <ul style="list-style-type: none"> <li>■ Impact Dot Matrix</li> <li>■ Bidirektionaler Druck</li> <li>■ Druckwegoptimierung</li> <li>■ 80 Zeichen/Sekunde</li> <li>■ 9x9 Matrix (echte Unterlängen) bei Zeichen</li> <li>■ 6x12 Matrix bei graphischen Symbolen</li> <li>■ 40, 66, 80, 132 Zeichen/Zeile (A4 hoch)</li> <li>■ Deutscher, Französischer, Englischer, Amerikanischer Zeichensatz umschaltbar</li> </ul> | <ul style="list-style-type: none"> <li>■ Lebensdauer 100 Mio. Zeichen</li> <li>■ Self Test Mode</li> <li>■ Tractor Feed (verstellbar)</li> <li>■ Papierendschalter mit Alarm</li> <li>■ On Line/Off Line-Schalter</li> <li>■ Form Feed, Horizontal-, Vertikal-Tabulator</li> <li>■ Farbbandkassette</li> <li>■ 12 verschiedene Schriftarten wählbar</li> <li>■ Parallel TTL (Centronics) Interface</li> <li>■ Optionen: RS232C/current loop, IEC Bus</li> <li>■ Apple II, TRS-80</li> </ul> |
|--|---|

Der Druckkopf kann mit einem Handgriff ausgewechselt werden und kostet Sie weniger als 50.- Franken!

Alle diese Leistungen bietet Ihnen nur EPSON.  
Angekündigt ... und bereits ab Lager lieferbar.

Exklusiv-Vertretung für die Schweiz + Liechtenstein.

**ADCOMP AG**



Computer-Systems - Components  
- Software - Education

ADCOMP AG, Steinwiesenstr. 3, 8952 Schlieren Tel. 01/730 48 48, Telex 58 657  
ADCOMP AG (Software + Education), Oberrasse 32, 8400 Winterthur, Telefon 052/22 32 73

## Kleincomputer-Software

- Auswahl des richtigen Computers
- Preiswerte und individuelle Problemlösungen
- Garantierte Wartung aller Programme

Roman Kaiser  
EDV-Beratung/Software  
Würzenbachstrasse 62  
6006 Luzern  
Telefon 041 31 49 69

**CP/M  
48 KB RAM  
700 KB DualDisk**

**zum sensationellen Preis von  
für den  
kompletten  
Computer**

**Fr. 4'850.-**

Telefon (01) 44 86 86  
Rosengartenstrasse 5, Zürich

**COMPUTERWARE**

# Kleincomputer aktuell



## INS 8073 – Basic on chip

Leopold ASBÖCK

Einchipcomputer finden überall dort Einsatz, wo es auf geballte Intelligenz auf kleinsten Raum ankommt, beispielsweise in Messgeräten, Maschinensteuerungen oder Computerperipherie. Die zugehörigen Programme werden in Maschinensprache entwickelt, was erheblichen Aufwand und wenig Komfort bedeutet. Neu auf dem Markt ist der INS8073 von National Semiconductor, ein Einchipcomputer, der einen Tiny Basic-Interpreter mitintegriert hat.

Mit dem wachsenden ROM-Umfang der Einchipcomputer war die Implementierung einer höheren Programmiersprache in 4 KByte-ROMs zu erwarten. National Semiconductor hat mit dem INS8073 den ersten Schritt in diese Richtung getan. In 2.5 KByte des ROMs residiert ein BASIC-Interpreter, der speziell darauf ausgelegt ist, die Programmierung in Maschinensprache zu ersetzen. Dadurch werden auch teure Entwicklungssysteme entbehrlich, miteingeschlossen aufwendige Software wie Editor, Assembler oder Debugginghilfen.

Für Programmerstellung und Fehlersuche lassen sich Zeitgewinne bis zu 90 % erzielen, was eine drastische Senkung hoher Softwareerstellungskosten bedeuten kann.

Mit weniger als 20 ICs lassen sich komplette Systeme aufbauen, die mit Videoterminal oder Teletype mit Lochstreifenleser in wählbaren Baudraten verkehren. Die neu erstellten BASIC-Programme können auf Lochstreifen oder Kassetten ausgegeben werden oder "onboard" in 2716-EPROMs geschrieben werden und sind sofort im Echtzeitbetrieb zu testen, was selbst mit teuren und komfortablen Entwicklungsgeräten manchmal ein Problem ist.

Im Echtzeitbetrieb wird ab externem ROM oder RAM kein Maschinencode mehr ausgeführt, sondern der Computer interpretiert die gespeicherten BASIC-Programme. Für zeitkritische Anwendungen wird durch einen LINK-Befehl die Verbindung zur Maschinensprache ermöglicht.

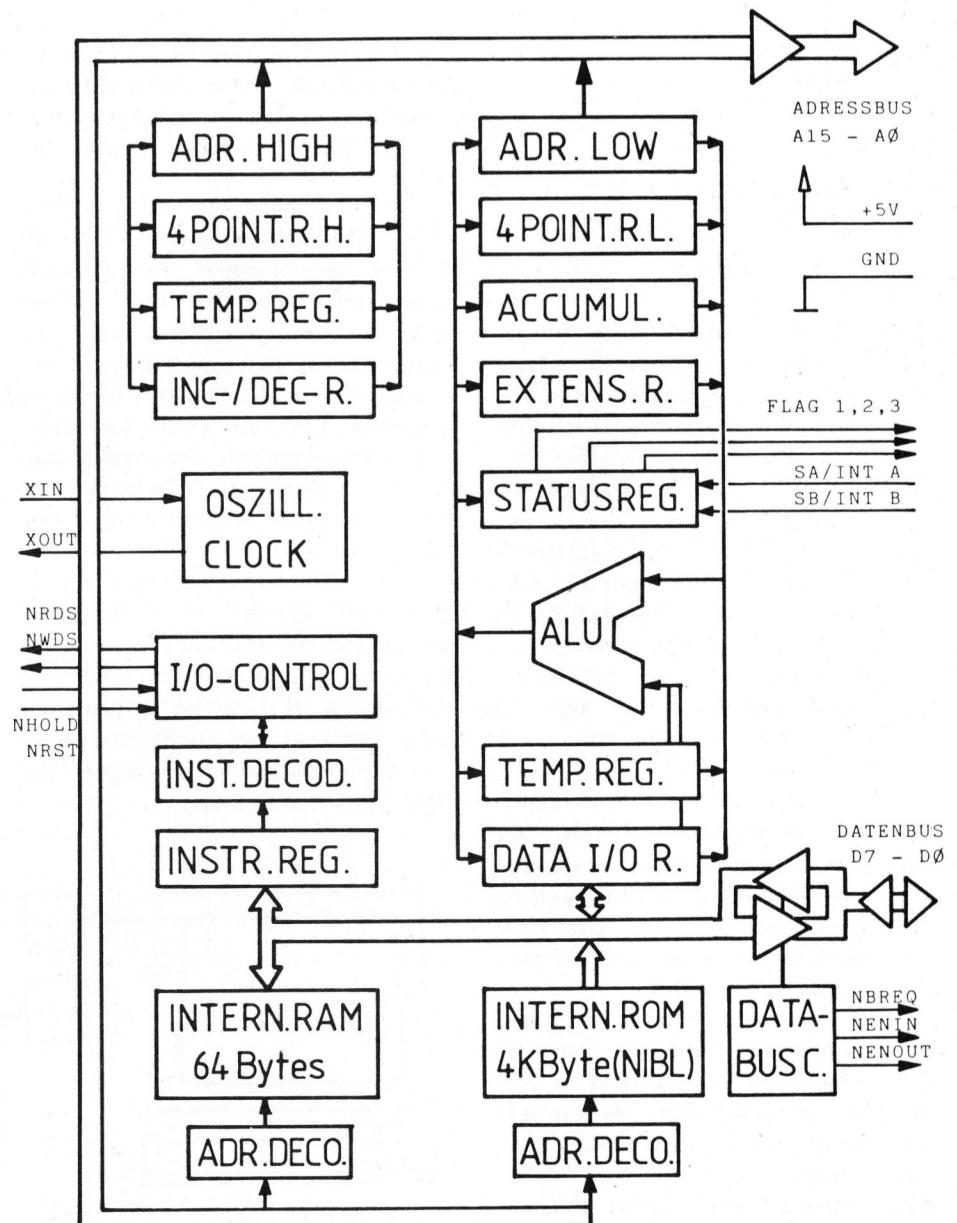


Abb.1 Interner Aufbau des INS 8073

# Kleincomputer aktuell

## INTERNE STRUKTUR

Der INS8073 wird in einem 40-pin-Gehäuse produziert und beinhaltet neben dem Tiny Basic-Interpreter im ROM noch 64 Bytes RAM, eine 8-bit-ALU (Arithmetic Logic Unit), einen 8-bit-Akkumulator sowie 16 weitere Register. Zusätzlich sind die notwendigen Logikeinheiten für Input-/Outputkontrolle, Buskontrolle, Dekodierung von Adressen und Instruktionen sowie ein Clockgenerator vorhanden (Abb.1).

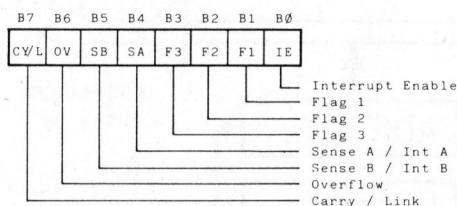


Abb.2 Statusregister

Der Computer kann über 16 Adressleitungen total 64 KByte Speicher- oder Peripherie-Adressen ansprechen, die Daten werden über acht Datenleitungen mit separaten RD- und WR-Strobes ein- und ausgegeben. Die übrigen Anschlüsse sind Interrupt-, Kontroll- und Input-/Outputlinien, über die ein serieller Datenverkehr mit vier wählbaren Baudraten ohne externen Peripheriebaustein erfolgen kann. Die notwendigen Ein-/Ausgaberoutinen sind im Interpreter bereits vorhanden.

Falls noch die entsprechenden Buffer angeschlossen werden, hat man eine vollwertige RS232-Schnittstelle zur Verfügung. Die Baudraten können mit 110, 300, 1200 oder 4800 Baud software- oder hardwaremässig gewählt werden.

Verzichtet man auf eine serielle Datenübertragung, so stehen die hierfür vorgesehenen Anschlüsse als zusätzlicher Interrupteingang bzw. Flagausgang zur freien Verfügung. Kontrolliert werden die beiden Sense-/Interrupteingänge A und B sowie die Flagausgänge 1, 2 und 3 über das Statusregister (Abb. 2), das mittels der Funktion "var-STAT"

gelesen oder in das mit "STAT=expr" geschrieben werden kann. So setzt zum Beispiel die BASIC-Anweisung "STAT=STAT OR 14" die drei Flagbits auf logisch "1".

Die beiden Interrupteingänge sind maskierbar (Statusbit 0), die beiden Bits für CARRY und OVERFLOW sind für BASIC-Anwendungen bedeutungslos, da sie vom Interpreter kontrolliert werden.

## EXTERNE HARDWARE

Mit zwei ICs, dem INS8073 und einem 1Kx8-RAM sowie einem Videoterminal oder Teletype lassen sich bereits BASIC-Programme entwickeln.

Der Interpreter benötigt mindestens 256 Bytes an externem RAM als "scratchpad memory". Pro Kilobyte BASIC-Programm speichern. Der RAM-Bereich muss zusammenhängend sein und kann maximal 60 KByte umfassen (Adresse 1000 bis FFBE hexadezimal). ROM-residente Programme sollten bei Adresse 8000 beginnen, da der Computer selbsttätig feststellt, ob ab Adresse 8000 ein ROM- oder RAM-Bereich vorliegt. Im ersten Fall startet er automatisch die Programmausführung bei Adresse 8000, was im Echtzeitbetrieb von Bedeutung ist, da im allgemeinen kein Terminal zur Verfügung steht und ein Autostart ohne Eingabe von "RUN" erfolgen muss.

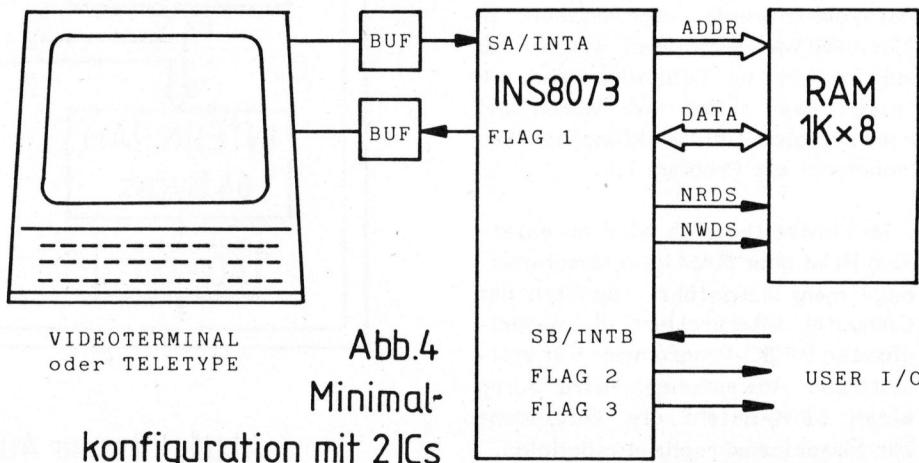


Abb.4 Minimal-Konfiguration mit 2 ICs

Im zweiten Fall stellt der Computer den zur Verfügung stehenden RAM-Bereich fest und springt dann in den Command-Modus. Zur Neueingabe eines Programms ist eine Initialisierung mit dem Befehl "NEW addr" vorzunehmen. Der Computer speichert die Programmstartadresse und markiert mit dem Delimiter 7F (hexadezimal) das Ende des BASIC-Programms. Somit können sich auch mehrere Programme gleichzeitig im Speicher befinden, mit "NEW addr RUN" wird das Programm ab Adresse "addr" ausgeführt.

NENOUT	1	40	VCC (+5V)
NENIN	2	39	SB/INT B
NBREQ	3	38	SA/INT A
NRDS	4	37	NRST
NHOLD	5	36	FLAG 3
NWDS	6	35	FLAG 2
XIN	7	34	FLAG 1
XOUT	8	33	D0
A15	9	32	D1
A14	10	31	D2
A13	11	30	D3
A12	12	29	D4
A11	13	28	D5
A10	14	27	D6
A9	15	26	D7
A8	16	25	A0
A7	17	24	A1
A6	18	23	A2
A5	19	22	A3
GND	20	21	A4

Abb.3 Pinbelegung des INS8073

## SYSTEM MIT 2 ICs

Wie in der Abb. 4 ersichtlich, erhält man bereits mit dem INS8073 und einem RAM ein funktionstüchtiges "Entwicklungssystem", ein Teletype mit Lochstreifeneinheit ermöglicht sogar Speicherung der Programme auf Lochstreifen.

# Kleincomputer aktuell

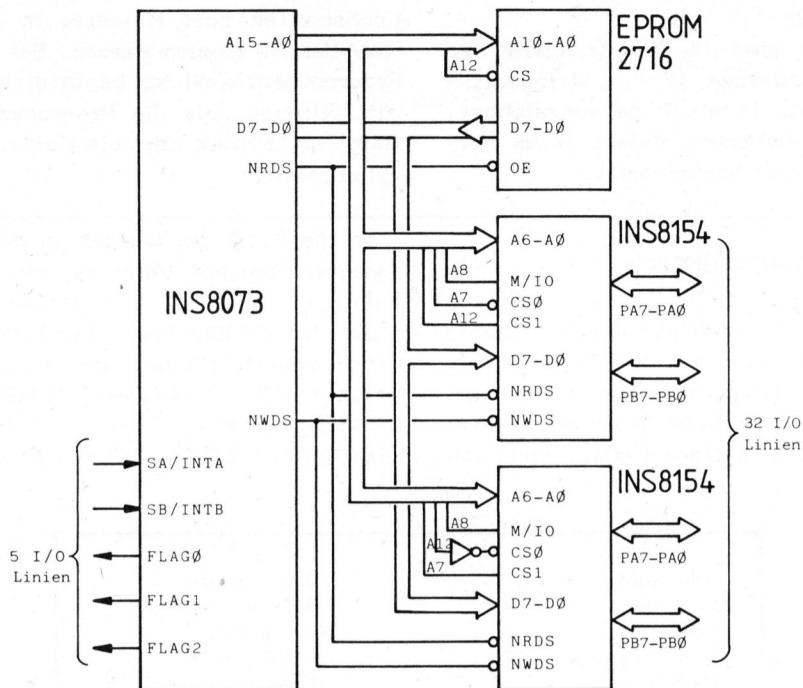


Abb.5 Echtzeitsystem mit 2K ROM und 37 I/O Linien

Ein praktisch einsetzbares Echtzeitsystem besteht aus vier LSI-ICs und einigen weiteren Komponenten und bietet neben einem 2K-ROM-Bereich die notwendigen 256 Bytes für den Interpreter sowie 37 Input/Outputlinien (Abb. 5).

Auf einer Karte mit den geringen Dimensionen von 12 x 18 cm lässt sich ein komfortables Entwicklungssystem erstellen, das neben 4 KByte RAM, 4 KByte ROM und 40 Input/Outputlinien noch Interfaces für Videoterminal, Teletype und Kassettengerät bietet und über eine Programmierereinheit für 2716-EPROMs verfügt (Abb. 6). Die Komponentenzahl von 19 ICs lässt sich noch reduzieren. Ein entsprechender Schaltungsentwurf mit ausführlicher Beschreibung ist im User Manual zum Tiny Basic Interpreter zu finden.

## BASIC-INTERPRETER

Der Interpreter ist speziell auf den Bedarf als Maschinensprachenersatz konzipiert und verfügt über einen leistungsstarken Befehlssatz. Man darf natürlich keine trigono-

metrischen Funktionen oder Gleitkomma-Arithmetik erwarten. Trotzdem findet man Befehle, die selbst in komfortablen BASIC-Interpretern vermisst werden, wie zum Beispiel DO-UNTIL-Schleifen.

Arbeitet man im Manual zum INS-8073 die Beschreibung der in Abb. 8 aufgeführten Befehle durch, so staunt man, wieviel einfacher sich Programme mit diesem "nur"-2,5K-

Interpreter im Vergleich zur Maschinensprache entwickeln lassen. Besonders einfach ist der Zugriff auf jede Speicherstelle und jedes einzelne Bit, was für entsprechende Anwendung (Bitmanipulation bei Steuerungen und Überwachungen) besonders wichtig ist. Zwölf Fehlermeldungen erleichtern das Auffinden von Programmfehlern.

Aus der BASIC-Befehlsliste sollen einige Anweisungen näher erklärt werden, die meisten Standardbefehle sind allgemein bekannt und bedürfen wohl keiner Beschreibung.

RND (X,Y)

ermittelt eine Zufallszahl im Intervall von X bis Y.

MOD (X,Y)

berechnet den Rest der Ganzzahldivision X/Y.

DO...UNTIL expr

bewirkt eine Programmschleife, die solange ausgeführt wird, bis die Bedingung expr erfüllt ist. Acht Schleifen können ineinandergeschachtelt werden.

LINK addr

führt einen Sprung in ein Maschinenprogramm ab Adresse addr aus. Nach dem RET-Code erfolgt die Rückkehr ins BASIC-Programm.

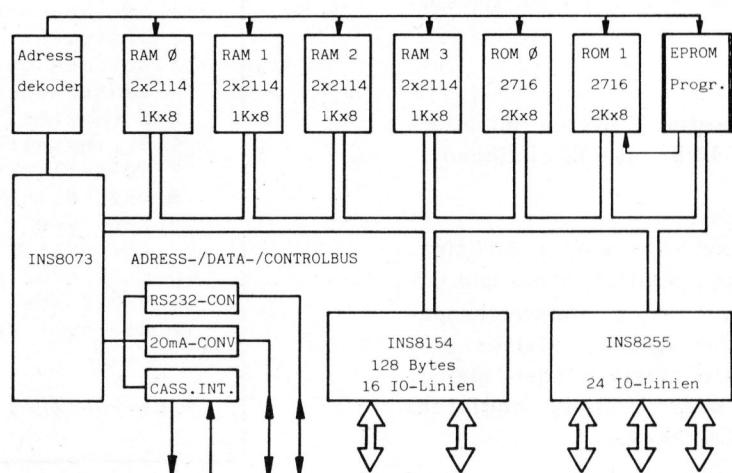


Abb.6 Komplettes Entwicklungssystem (NSC Tiny Basic Card)

# Kleincomputer aktuell

## DELAY expr

gestattet eine präzise Verzögerung in Vielfachen einer Millisekunde.

## ON 1 (oder 2) Zeilennummer

bewirkt einen Sprung (GOSUB) zur angegebenen Zeilennummer bei Interrupt auf Anschluss INTA oder INTB.

## STAT=expr, bzw. var=STAT

schreibt den Wert von expr in das Statusregister, bzw. liest den Inhalt des Statusregisters in die Variable var.

## INC(X), DEC(X)

werden in Multiprozessorsystemen mit gemeinsamen RAM-Bereichen eingesetzt und sind nichtinterruptbare Inkrementier-/Dekrementier-Ladebefehle.

## TOP

gibt jeweils die kleinste nichtbelegte RAM-Adresse an. Beispiel: PRINT TOP ; A=TOP+16 etc.

## "at sign" @

entspricht dem Ausdruck "at the location", beispielsweise bewirkt @4096=33 die Speicherung von 33 im Speicherplatz 4096.

@A=2\*B+1 speichert das Ergebnis des rechtseitigen Ausdruckes in der Speicherstelle mit der Adresse in A.

## VARIABLE

26 Variable (A...Z) stehen zur Verfügung.

## KONSTANTE

Dezimalkonstante dürfen im Intervall von -32767 bis +32767 liegen.

## OPERATOREN

Arithmetikoperatoren für Addition, Subtraktion, Multiplikation und Division: +, -, \*, /. Vergleichsoperatoren für gleich, kleiner als, grösser als, kleiner oder gleich, grösser oder gleich, ungleich: =, <, >, <=, >=, <>.

## LOGIKOPERATOREN

AND, OR, NOT

## STRINGS

Strings sind in Anführungszeichen eingeschlossen (" "). Stringvariable sind durch \$ gekennzeichnet. Zur Speicherung belegt jedes Zeichen einen Speicherplatz.

## UTILITY-ROUTINEN

Häufig benötigte Maschinenprogramme, die mit dem Befehl LINK aufgerufen werden, können in einem EPROM oder ROM deponiert werden oder bei seltenem Gebrauch von

Lochstreifen oder Kassette in den RAM-Bereich gelesen werden. Bei der Programmentwicklung benötigt man oft Utilities, die die Programmierung von EPROMs oder die Kontrolle erleichtern.

Solche Programme werden in 8070-Assemblerform mit Maschinencode zur Verfügung gestellt und finden in einem 2716-EPROM Platz. Die EPROM Programming Software umfasst die Befehle COPY, PROGRAM, VERIFY, ERASE CHECK, FILL, DUMP, LOAD, ASCII LOAD, WRITE, READ.

NEW addr	GOTO expr
RUN	GOSUB expr
CONT	RETURN
LIST (expr)	INPUT var
REM	INPUT \$var
CLEAR	LINK expr
(LET) var=expr	ON expr1 expr2
(LET) STAT=expr	DELAY expr
(LET) \$addr=expr	STOP
(LET) \$var="string"	STAT
(LET) \$addr1=\$addr2	TOP
PRINT expr	INC (X)
PRINT "string"	DEC (X)
PRINT \$addr	MOD (X,Y)
IF expr (THEN) statem	RND (X,Y)
FOR var=expr1 TO expr2	AND, OR, NOT
(STEP expr3)	+ , - , * , /
NEXT var	< , > , <> , -
DO	<= , >=
UNTIL expr	

## BASIC-Befehle und Funktionen des INS8073

- 1 Speicherbereich überschritten
- 2 Statement unzulässig
- 3 Zeichen unzulässig
- 4 Syntaxfehler
- 5 Formatfehler
- 6 Stringdelimiter " fehlt
- 7 GO-Zeilennummer existiert nicht
- 8 RETURN ohne GOSUB
- 9 Mehr als 8 FOR-NEXT oder DO-UNTIL Schleifen verschachtelt
- 10 NEXT ohne FOR
- 11 UNTIL ohne DO
- 12 Division durch Null

Fehlermeldungen des Microinterpreters

Abb.8 Befehle, Funktionen und Fehlernummern

# Kleincomputer aktuell

## PET jetzt mit PASCAL

Jack MEYER

Das lange Warten hat sich gelohnt: Nun ist ein PASCAL-Compiler auch für den PET/CBM erhältlich. Mit diesem Compiler schaffen Sie den Sprung von BASIC in die strukturierte Computersprache PASCAL. Ein ROM, eine Diskette und ein (in englischer Sprache abgefasstes) Manual bieten die Grundlage für den Einstieg in diese Sprache, die immer mehr Liebhaber findet.

Das über 100-seitige Manual ist relativ einfach geschrieben, so dass es auch mit weniger guten Englischkenntnissen gelesen werden kann. Aehnlich einem Lehrgang beginnt es einfach und für jeden verständlich, wird komplexer und dringt immer tiefer in die Materie ein. Die Blätter sind zudem so eingehaftet, dass sie auch nach 20-maligem Vor- und Rückwärtsblättern nicht herausfallen.

Der Einbau und die Inbetriebnahme des TCL PASCAL (TRANSAM COMPONENTS LIMITED PASCAL) Compilers ist sehr einfach. Das mitgelieferte Sicherheits-ROM wird in den mittleren freien ROM-Sockel (Adr. A000-AFFF) gesteckt. Die Software, d.h. die Pascal Diskette wird vom Drive eingelesen, mit RUN gestartet, und fertig ist das Wunderwerk: Unser CBM "spricht" PASCAL.

### ALLGEMEINES ZUM COMPILER

Als erstes sei hier erwähnt wie der Compiler zu laden ist, und wie man ihn später anwendet.

Man lädt ihn wie ein BASIC-Programm mit

```
LOAD"*",8
```

Nach RUN (Return) ist er bereits geladen. Die Anwendung sei an einem sehr einfachen Programm beschrieben:

```
10 begin
20 write ("Hallo Pascal-Freund")
30 end.
```

Nach Eintippen der ersten Zeile und einem CR (=Return) antwortet unsere Maschine bereits mit

```
20
```

(Diese Zeilennummerierung ist für den Programmablauf bedeutungslos, dient aber zur Lokalisierung von Fehlern.) Nach dem "end." schreibt der Computer

```
40
```

Nun geben wir eine leere Zeile ein (nur Return drücken), und damit ist unser Programm im Computer. Um es ablaufen zu lassen, haben wir nichts anderes zu tun als

```
r (Return)
```

zu drücken. Der Compiler übersetzt nun unser Programm in einen Maschinencode und lässt es ablaufen. Auf dem Schirm erscheint

```
compiling
program 0 0509
0 error (s)
compilation complete.
Hallo Pascal-Freund
```

Unser Programm ist nun im Maschinencode ebenfalls im Speicher und nach weiterem Drücken von r (Return) läuft es blitzartig ab, so schnell wie Maschinenprogramme eben laufen. Einzig die Uebersetzung von PASCAL in Maschinensprache braucht etwas Zeit. Aber die wird ja nur einmal benötigt, weil der Code auch gespeichert bleibt. Als Beispiel sei hier ein Programm erwähnt, das die Primzahlen von 2 bis 127 ausrechnet und auf den Schirm gibt.

```
program Primzahlen;
const n = 127;
var sieb : set of 2..n;
    Zahl, i: integer;
begin
    sieb := [2..n];
    for Zahl := 2 to n do
    if Zahl in sieb then
        begin writeln (Zahl);
            for i := 2 to n div Zahl do
                sieb := sieb - [i*Zahl]
            end
        end.
```

Die Compilation benötigt rund 8 s. Die Abarbeitung dieses Programmes braucht jedesmal nur noch 3 s. Ein ähnliches Programm in BASIC würde jedesmal 6 s benötigen.

## Textverarbeitung?

Rationalisieren Sie Ihre Textverarbeitung mit dem

**DCT-Super-Star**  
**Fr. 18'095.— inkl. Wust**  
für ein sofort einsatzfähiges Anwender-System!



Bestehend aus dem superschnellen **DCT-Superbrain** mit 64 K, zwei integrierten Floppies, automatische Floppyabschaltung, dem exklusiven Schönschreibdrucker **NEC-Spinwriter**, der benutzerfreundlichen Software **TEXT-STAR** mit deutscher Anleitung.



DIALOG COMPUTER  
TREUHAND AG  
Seeburgstrasse 18  
6002 Luzern  
☎ 041-3145 45

# Kleincomputer aktuell

## RESIDENT-MODUS/DISK-MODUS/ SYNTAX

Alles bisher Beschriebene funktioniert einwandfrei mit dem residenten Compiler in den RAMs. Einzig die File-Strukturen und natürlich die Disketten-Operationen laufen nur im Disk-mode. Aber "Disk-mode" was ist das? Nun im resident-mode ist ein etwas kleinerer und einfacherer Compiler in ca. 26K Bytes RAM gespeichert. Er erlaubt uns fast alle Operationen zu programmieren. Dieses etwas verminderte PASCAL ist etwa mit dem PASCAL-S an der ETH zu vergleichen.

Im Gegensatz dazu stehen im Disk-mode über 28K (beim 32K-PET/CBM) für eigene Programme zur Verfügung und der ganze Compiler ist auf Diskette. In diesem Modus sind auch alle möglichen PASCAL-Befehle und -Strukturen erlaubt.

Auf die allgemeine Syntax der Prozeduren, Funktionen und Files wird hier nicht eingegangen. Einzig über die rekursive Anwendung von Prozeduren und Funktionen soll schnell etwas gesagt werden. Eine rekursive Prozedur ist eine Prozedur, die sich selber aufruft. Solche Anwendungen geben bestechende (leider aber auch speicherplatzfressende) Lösungen vor allem bei Denksportaufgaben und Puzzles, wie z.B. beim "Towers of Hanoi". Dieses wohl allen bekannte Spiel mit den 3 Pfählen und den Scheiben, wobei nie eine grössere über einer kleineren liegen darf, wird durch folgendes sehr kurze Programm gelöst:

```
program Hanoi;
var ndiscs: integer;
procedure move (source, destn,
  spare: 1..3; n:integer);
begin
  if n>1 then move (source,
    spare, destn, n-1);
  writeln('Moving from', source:
    2, ' to ', destn : 2);
  if n>1 then move (spare,
    destn, source, n-1)
end;
```

```
begin
  write ('How many discs ? ');
  read (ndiscs);
  writeln;
  move (1,3,2,ndiscs)
end.
```

Die Klärung des Programmablaufes sei dem Leser überlassen.

## EDITOR-BEFEHLE

Als erstes ist hier zu erwähnen, dass im PASCAL-Modus das gesamte DOS für die Disketten mit denselben Befehlen wie in BASIC enthalten ist. Ebenso ist ein Auto-Repeat wirksam, das nach längerem Niedergedrückthalten einer Taste dieses Zeichen in rascher Abfolge auf den Schirm bringt (besonders wertvoll für Cursorsteuerung). Das TCL-PASCAL besitzt folgende Editor-Befehle:

new	wie in BASIC
list	gleich wie beim BASIC
upper, lower	zur Umschaltung für Grossbuchstaben oder Grafikzeichen
basic	bewirkt Rückkehr in den BASIC-Modus
break	wirkt wie ein brk im Maschinencode; dient zum Einsteigen in den Monitor (zurück zu PASCAL: x (Return))
auto	wie beim Toolkit; zur Aenderung der Zeilensprünge; aber Pascal numeriert auch ohne diesen Befehl die Zeilen automatisch mit 10er-Sprüngen.
number	ähnlich dem "Renumber" beim Toolkit
find	wie beim Toolkit
delete	wie beim Toolkit
change	Bsp. Change/integer/real Aendert im Programm alle "integer" in "real" um; Spezifikation zur Wahl der Zeilennummern wie bekannt z.B. bei List 100-200
hex	Gibt die entsprechende Hex-Zahl zur Dezimalzahl
decimal	Gibt die entsprechende dezimale Zahl zur Hex-Kombination z.B. decimal 5d ergibt 93

put	Speichert ein Programm auf Diskette (vgl. save)
get	Lädt ein Programm von Diskette (vgl. load)
disk	erzeugt den Disk-Modus (grosses PASCAL); Compiler auf Diskette
resident	lädt den Compiler wieder in die RAMs (kleines PASCAL)
r oder run*	wie RUN in BASIC
l*	Compiliert ein Programm und gibt es auf den Schirm
p*	Compiliert ein Programm und gibt es auf den Drucker
comp**	Compiliert das Programm und speichert den Maschinencode wieder auf Diskette
ex**	lässt ein Maschinencode-Programm von der Diskette aus laufen.
link**	erlaubt ein Zusammenfügen von Programm-Segmenten auf Diskette.
locate**	erstellt ein File auf Diskette in Maschinencode, das aber von BASIC aus aufgerufen werden kann, und wie bekannt im BASIC mit RUN gestartet wird.

\* nur im Resident-Modus

\*\* nur im Disk-Modus

## ERWEITERUNGEN ZUM TCL PASCAL

Als letztes seien noch die Erweiterungen des TCL PASCAL gegenüber dem Standard PASCAL aufgeführt

- Hex-Konstanten  
Konstanten können auch in Hexadezimaler Form (mit vorgestelltem \$-Zeichen) verwendet werden.
- erweiterte "Bildschirmbefehle"  
peek, poke wie beim BASIC (über ganzen Speicherbereich)  
origin  
setzt einen Zeiger (irgend ein Zeigertyp) auf einen Speicherplatz

# Kleincomputer aktuell

- vdu setzt ein Zeichen in bestimmter Zeile und Kolonne auf dem Schirm; Bsp: vdu (12, 20,'\$') ergibt etwa in der Mitte ein \$-Zeichen
- getkey als Prozedur aufgerufen gibt dies ein Zeichen von der Tastatur (vgl. in BASIC "get")
- c) Hexadezimaler In/Output
- wrhex schreibt eine Zahl als Hex-Kombination auf ein File (spez. auch Output)
- wrhex 2 schreibt eine Zahl (0..255) als 2er Hex-Zahl auf ein File (auch Output)
- rdhex liest eine Hex-Zahl (16 bit) von einem File (auch Input)
- d) Bit-Manipulation
- andb Angewendet auf logische Daten (16 bit):
- orb and, or, exclusive
- xorb or, complement
- notb Bsp. orb (\$FF00, \$0FF0) = FFF0
- shl (x,y) schiebt x um y bits nach links
- shr (x,y) schiebt x um y bits nach rechts
- Bsp. shl (4,4) = 64 = \$40 links gehen die bits verloren, rechts kommen 0 dazu
- e) Unterdrücken von I/O errors
- iotrap (false) unterdrückt bei read-Operationen (auch von Diskette) alle Runtime-Fehler-Meldungen
- ioerror als Funktion aufgerufen ergibt sie den Code für den aufgetretenen Fehler, bzw. 0 für keinen Fehler (Codeliste für Fehler im Manual)
- iotrap (true) stellt Originalzustand wieder her; bei
- Input-Fehler gibt der Computer eine Fehlermeldung auf den Schirm und "steigt aus dem Programm".
- f) Stop-Taste-Unterbrechung
- breaks (false) bewirkt, dass die Stop-Taste keinen Einfluss mehr hat
- breaks (true) Original-Zustand; durch Drücken der Stop-Taste kann ein Programm-Ablauf unterbrochen werden
- g) Zufallsgenerator
- random als Funktion aufgerufen ergibt dies eine Pseudo-Zufallszahl zwischen 0 und 255
- h) Pet-clock
- set time setzt die Zeit (ähnlich bei BASIC:TI\$)
- hours damit können Stunden,
- minutes Minuten, Sekunden
- seconds einzeln abgefragt werden
- i) Programmverkettungen
- chain (filename) damit ist es möglich von einem Programm aus ein anderes Programm als Subroutine aufzurufen und ablaufen zu lassen, es ist sogar möglich ein BASIC-Programm als Unterprogramm einzufügen
- k) Maschinensubroutinen
- Es ist möglich von einem Programm aus ein Unterprogramm in Maschinencode ab einer bestimmten Adresse aufzurufen. Man kann dieses Maschinenprogramm selber schreiben oder ein schon bestehendes (in den ROMs) benutzen.

## ZUSAMMENFASSUNG

Der TCL-PASCAL-Compiler läuft nun schon seit mehreren Wochen ein-

wandfrei. Im grossen und ganzen ist er ein sehr guter Zusatz zum PET/CBM. Weil er speziell auf diesen Kleincomputer zugeschnitten ist, gehen fast keine PET-Funktionen verloren.

Wie wir gesehen haben, weist dieser Compiler viele Vorteile auf. Als einziger Nachteil ist vielleicht zu erwähnen, dass das bekannte Abkürzungsverfahren, mit dem zweiten geschifteten Buchstaben, hier leider nicht funktioniert.

Für den PASCAL-Freund und PET/CBM-Besitzer ist dieser PASCAL-Compiler eine lohnende Anschaffung. Einen guten Einstieg vermittelt das in Englisch geschriebene Buch:

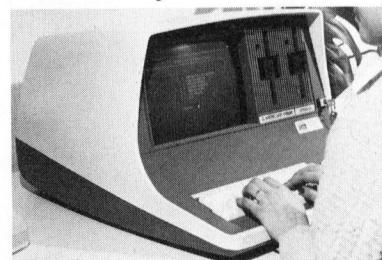
PASCAL-User Manual and Report, Niklaus Wirth, 1978, Springer Verlag

Beides, Buch und Compiler, sind beim SCC erhältlich.

## Finanzbuchhaltung?

Rationalisieren Sie Ihre Buchhaltung mit dem

**DCT-Super-Käfer**  
**Fr. 14'490.- inkl. Wust**  
 für ein sofort einsatzfähiges Anwender-System!



Bestehend aus dem superschnellen **DCT-Superbrain** mit 64 K, zwei integrierten Floppies, automatische Floppyabschaltung, dem Matrix-Drucker **EPSON MX-80**, inkl. der benutzerfreundlichen Software **FINANZBUCHHALTUNG nach Dr. Käfer** mit deutscher Anleitung.



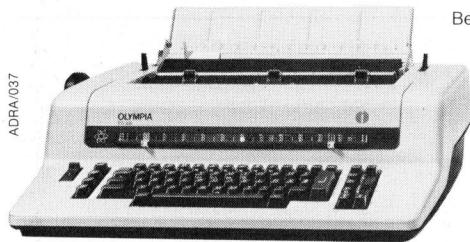
DIALOG COMPUTER  
 TREUHAND AG  
 Seeburgstrasse 18  
 6002 Luzern  
 ☎ 041-3145 45

# Olympia – und Ihr Mikrocomputer lernt gestochen scharf korrespondieren.

Olympia zeigt Ihnen, wie auch Ihr Kleincomputer gestochen scharf Briefe, Tabellen, Statistiken usw. schreiben kann. Schnell und leise dank auswechselbaren Typenrädern (verschiedene Schriftentellungen). Auf Wunsch erhalten Sie 1 Original mit 6 Kopien!

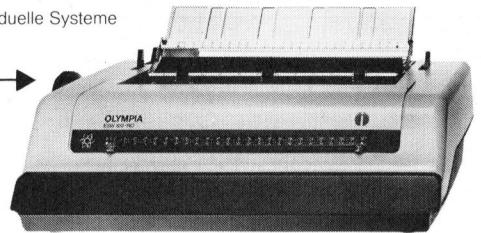
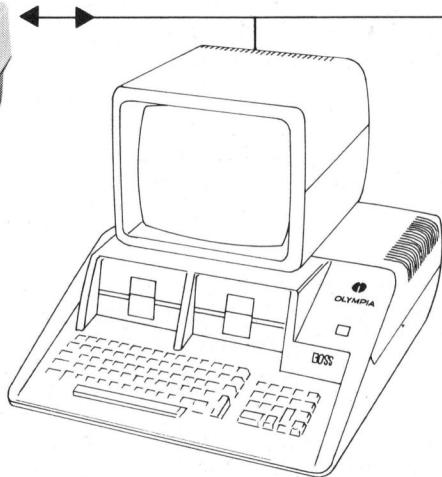
Dabei müssen Sie beim Modell ESW 100 KSW selbst auf die von normalen Schreibmaschinen her gewohnten Korrekturmöglichkeiten wie Lift-off und Cover-up nicht verzichten. Zur Wahl stehen zwei Schönschreibdrucker. **Olympia ESW 100 KSR:**

Durch einfaches Umschalten ist dieses Spitzenmodell sowohl Ein-/Ausgabeterminal wie auch «normale» Standard-Schreibmaschine. – **Olympia ESW 100 RO:** Dieses Modell ist für den professionellen Computereinsatz mit tastaturorientierten Bildschirmen.



**Typenradschreibwerk ESW 100 KSR**  
EIA RS 232 C, CCITT V24,  
Blocktastatur, 46 Schreibastern (davon 45 als Repetitionstasten) und 20 Funktions-tasten, zum Preis, der leicht über einer «normalen» Schreibmaschine liegt!

Beide Typen können als OEM-Drucker ohne Gehäuse in individuelle Systeme integriert werden.



**Typenradschreibwerk ESW 100 RO**  
Parallel, 7 bit ASCII, centronics.  
Umfangreicher Befehlssatz für Funktionen wie Walzenschnelllauf, Tabulation, Plotten usw.



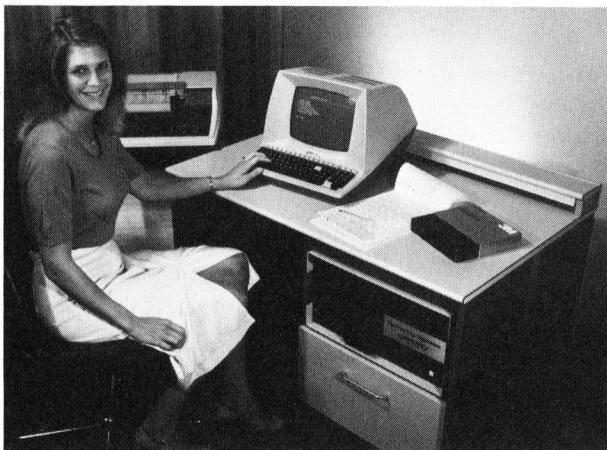
**Olympia International**

**OLYMPIA Büromaschinen AG**

8153 Rümlang, Ifangstrasse 91, 4003 Basel, Leimenstrasse 47, 3008 Bern, Freiburgstrasse 177, 1205 Genf, 9, bd. Philosophes

## Marinchip Systems

Minicomputer



Die günstigste Lösung Ihres EDV-Problems mit:

- 16 Bit CPU
- 128 KByte Speicher
- 26 MByte Hard Disk
- Printer, Videoterminal
- Buchhaltungssoftware

Fr. 41 000.–

**COMPU RENT**

Dornacherstr. 119, 4053 Basel  
Tel. (061) 35 04 70

**Dysan**  
CORPORATION  
**100% Sicherheit  
mit Dysan-Disketten**

**KUFFER ELECTRONIQUE**

Postfach 1 • CH-8965 Berikon

Tel. 057 5 22 33

6, chemin du château • CH-1245 Collonge

Tel. 022 52 33 63

Für die Realisierung mehrerer anspruchsvoller Projekte suchen wir per sofort oder nach Übereinkunft einen versierten

### APPLE-PROGRAMMIERER

der sich möglichst auch über Grosscomputer-Erfahrung ausweisen kann. Die Zusammenarbeit mit uns kann auf der Basis freier Mitarbeit oder im Angestelltenverhältnis erfolgen.

Für eine erste, unverbindliche Besprechung über unsere Ziele und Erwartungen bitten wir Interessenten, sich mit Herrn O. Herber oder Rüfenacht in Verbindung zu setzen.



**analytic ag**

Strassburgstrasse 15

8036 Zürich

Telefon 01 241 39 21

# Kleincomputer aktuell

## Computerneuheiten

Spektakuläre Computerentwicklungen wären unmöglich, würde nicht die Forschung in zahlreichen Laboratorien die technischen Grundlagen dafür schaffen. Oft überholt ein Projekt ein anderes vor seiner endgültigen Realisierung. Derzeit bemüht man sich, in "tiefgekühlten" Computern mit Josephson-Elementen die Schaltgeschwindigkeit zu vervielfachen - gleichzeitig versprechen Beobachtungen an Titaniumboritkristallen bei Raumtemperatur Zykluszeiten für Prozessoren im Nanosekundenbereich. Sollten weitere Entwicklungen Erfolg haben, darf man in Zukunft mit Computern rechnen, die einige tausendmal schneller sind als heutige Modelle.

### COMMODORE VIC 20

Für knapp \$300 bietet COMMODORE den VIC 20 an, der als "Micro-PET" bereits durch die Fachpresse geisterte. Der VIC 20 kann an einen Farbfernseher angeschlossen werden, vermag allerdings nur 23 Zeilen zu je 22 Zeichen darzustellen. Als Prozessor ist der wohlbekannte 6502 zu finden, ein hochintegrierter Videodisplaycontroller schafft die Verbindung zum Farbfernseher. Der RAM/ROM-Bereich umfasst 5 KByte bis 32 KByte und enthält eine BASIC-Version, die etwa dem PET-BASIC entspricht. Vorgesehen sind Ports für Kassettenanschluss, Speichererweiterung und ein serielles Interface für Floppy Drives sowie eine RS232C-Schnittstelle.

### NEUE HEWLETT-PACKARD-COMPUTER

Eine flexiblere Gestaltung der Peripherieeinheiten bietet der HP-83, der ähnlich dem Modell HP-85 gestaltet ist, jedoch auf eingebaute Kassettenstation und Drucker verzichtet und dadurch um ein Viertel billiger ist. Je nach Bedarf können Disk Drives, Drucker, Plotter etc. angeschlossen werden.

Neu unter den programmierbaren Taschenrechnern ist der HP-41CV, dessen Speicherbereich gegenüber dem bisherigen Spitzenmodell HP-41C verfünffacht wurde, so dass er jetzt 2000 Programmzeilen fasst.

### EPSON MX-70

Er gehört zu den billigsten Matrixdruckern am Markt, besticht aber nicht nur durch seinen tiefen Preis, sondern auch durch seine Leistungsfähigkeit: Der MX-70 von EPSON druckt 80 Zeichen pro Sekunde, jedes Zeichen in einer 5x7-Matrix. Zeilenvorschub, Formullänge und -beginn sind programmierbar. Einstellbarer Tractor und Selbsttest sind selbstverständlich.

Ein besonderes Plus ist GRAFTRAX II, ein Grafikroutine, mit deren Hilfe sich Grafik mit hoher Auflösung (480 Punkte horizontal) ausdrucken lässt.

Allmählich dürfte die Fähigkeit, hochauflösende Grafik zu drucken, zum Standard bei Matrixdruckern werden.

### 5 1/4-ZOLL FLOPPY DRIVES 70 % BILLIGER!

MICRO PERIPHERAL INC. ist die erste Firma, die nach gründlicher Ueberarbeitung des Floppy Drive Designs 5 1/4-Zoll Floppies auf den Markt bringt (MPI 61), deren Preis wesentlich unter dem heute üblichen durchschnittlichen Preis liegt.

Bei einer Speicherkapazität von 250, bzw. 500 KByte ist diese Preisreduktion durch Verzicht auf jedes überflüssige Detail in der

elektronischen und mechanischen Konstruktion möglich. Statt Servomotoren werden mechanische Positioniereinrichtungen verwendet, die Anzahl elektronischer Komponenten wurde wesentlich verkleinert. Uebrig bleibt ein Drive (Vergleich mit MPI 52), dessen Zugriffszeit 8 ms statt 5 ms beträgt, dessen Transferrate nicht an 125 KByte/Sekunde herankommt und dessen MTBF-Wert (meantime between failures) bei 2500 Stunden statt bei 9200 Stunden liegt. Die Spurfehler-rate ist mit 1 : 10 hoch 6 ungefähr gleich wie bei teureren Modellen.

Gesamthaft reichen diese Bedingungen also durchaus für den Einsatz in Personalcomputern und Small business Computern. Falls andere Firmen diesem Beispiel folgen, kann mit weiteren Preissenkungen bei Kleincomputern gerechnet werden.

### 3 1/2-ZOLL MICRO FLOPPY

SONY Corp. hat speziell für seine Textverarbeitungssysteme ein Micro Floppy entwickelt, das auf 3 1/2-Zoll-Disketten (etwa 9 cm Durchmesser) unformatiert Platz für 437,5 KByte bietet. Erzielt wird dieses hohe Speichervermögen auf dieser kleinen Diskette durch enge Schreibspuren und vergrößerte Schreibdichte.

Während es 5 1/4-Zoll-Disketten auf 48 oder 96 Spuren pro Zoll bringen, erreicht SONY durch ein neues Design des Schreib-/Lesekopfes und ein neuartiges Positioniersystem für die Diskette eine Spurdichte von 135 Spuren pro Zoll.

Mit den Erfahrungen bei der Entwicklung von Videobändern gelang auch eine bessere Diskettenbeschichtung, die eine Bitdichte von 7610 Bjts/Zoll zulässt - im Vergleich zu 5000 bis 6000 Bits/Zoll

# Kleincomputer aktuell

bei grösseren Disketten. Zusätzlich wurden die Diskettenverpackung und der Driveschlitz verbessert, um Staub besser abhalten zu können. Preislich werden die Micro Floppies ungefähr mit den 5 1/4-Zoll-Floppies gleichhalten.

Vermutlich werden in nächster Zeit leistungsstarke, sehr kompakte Textverarbeitungssysteme auf den Markt kommen. Sie werden mit Tastatur, Mehrzeilendisplay (Flüssigkristall- oder Fluoreszenzausführung) und Micro Floppies ausgestattet sein und in einem Aktenkoffer Platz finden!

## PASCAL MICROENGINE AUF S-100

Unter der Bezeichnung "PASCAL 100" bietet DIGICOMP RESEARCH eine interessante Kombination von zwei S-100-Platinen an. Auf einer Platine befindet sich die PASCAL MICROENGINE von WESTERN DIGITAL. Das sind fünf 40-polige ICs, die ein 16-bit-Prozessorsystem bilden, das den PASCAL-p-Code direkt und mit hoher Geschwindigkeit ausführt. Die Kommunikation mit dem S-100-System besorgt ein Z80-Prozessor, der auf der zweiten Platine installiert ist. Der Adressbereich umfasst 128 KByte, kann aber durch eine Option auf 1 Megabyte erweitert werden. Insgesamt können 512 I/O-Ports angesprochen werden.

Ein nicht zu verachtendes Plus ist die Softwarekompatibilität mit CP/M. Zudem unterstützt PASCAL 100 die Version III von UCSD PASCAL inklusive Bildschirmeditor, Compiler, Linker, Filer etc.

Anstrengungen werden bereits unternommen, um auch mit FORTRAN, BASIC, COBOL, LISP und ADA in p-Code zu compilieren und mit der MICROENGINE zu interpretieren.

Die Ausführungsgeschwindigkeit von PASCAL-Programmen ist bei 3 MHz Prozessortaktfrequenz etwa 10-mal schneller als bei Prozessoren, bei denen der p-Code (Pseudocode)

softwaremässig interpretiert wird und nach Behauptungen von Digicomp Research rund 1,5mal so schnell wie beim Minicomputer PDP-11/45 von DEC. Der Preis des "PASCAL 100"-Platinensatzes liegt etwa bei \$1500.

## ADA MICROENGINE

Mit einem Kleincomputer, dessen Prozessorsatz Programme, die in der Programmiersprache ADA abgefasst sind, direkt ausführt, trumpft WESTERN DIGITAL Corp. auf. Der Chipsatz, bestehend aus fünf 40-poligen ICs hat eine 16-bit-Busstruktur und ist bis auf die ROM-Inhalte mit der PASCAL MICROENGINE identisch.

Es werden aber nicht nur diese fünf ICs, sondern auch ein kompletter Computer angeboten, der im Gegensatz zur PASCAL-Version nicht auf einer einzigen Platine, sondern modular auf fünf Platinen aufgebaut ist: Prozessorplatine, Speicherplatine 128 KByte, Floppy Disk Controller, Input-/Outputplatine mit vier seriellen und parallelen Schnittstellen, Interface-Controller.

Im 10-slot-motherboard finden neben diesen fünf Grundplatinen noch weitere I/O-Karten, Harddiskcontroller, Realtimeclock-, Datenverschlüsselungs- und Fehlerkorrekturkarten Platz.

An Geschwindigkeit wird dieser ADA-Computer kaum von einem ADA-Compiler zu übertreffen sein. Der vorläufige einzige Konkurrent ist der iAPX-432 von INTEL, ein Prozessor, der eventuell ADA direkt ausführt, doch wird er auf Grund seiner 32-bit-Struktur kaum in Kleincomputern Einsatz finden.

## COMPUTER IN CHINA

Mit der grössten Bevölkerungszahl der Erde ist China auf Grund seiner Schrift vom Grosseinsatz von Compu-

tern weitgehend ausgeschlossen. Die chinesische Schrift, die trotz sprachlicher Verschiedenheiten für eine Milliarde Menschen eine Verständigungsmöglichkeit bietet, erweist sich als grosses Problem für den Einsatz von Computern.

Videoterminals kommen allmählich auf den Markt, die mit über 200 Tasten, welche fünffach belegt sind, eine Zusammensetzung von chinesischen Schriftzeichen erlauben. Auch Matrixdrucker sind in Produktion, die allein für den Zeichengenerator 2-Megabit-ROMs benötigen.

FUJITSU und TORAY Industries entwickeln nun einen Laserdrucker für chinesische und japanische Schriftzeichen, der eine Geschwindigkeit von 10'000 Zeilen pro Minute ermöglicht.

Eine Publikation des  
Schweizer Computer Club

## GBM/PET NEWS

bring's in deutscher Sprache

Unentbehrlich für jeden CBM/PET-Fan.  
Erst jetzt wissen Sie, was in Ihrem  
Commodore-Rechner steckt.

ALLES ÜBER



Da steht alles drin ...

Alle zwei Monate neue, interessante Informationen, Tricks und Kniffe ausschließlich für Ihren CBM/PET. **Mit Programmen** zum eigenen Gebrauch und Anleitungen zur Programmierung in BASIC sowie in Maschinensprache.

Verlangen Sie unverbindlich eine  
Probenummer beim

Verlag SCC AG  
Seeburgstrasse 12  
CH-6006 Luzern

# SMALL BUSINESS

## Recht und Computerbuchhaltung

Carlo BAZZANI, eidg. dipl. Buchhalter

Heute werden in zunehmendem Masse mit Kleincomputern Buchhaltungsarbeiten erledigt, für die noch vor wenigen Jahren ausschliesslich der Buchhalter oder Treuhänder zuständig war. Und gerade der gesamte Treuhand-Buchhaltungsbereich gewinnt durch den Einsatz dieser Kleincomputer an Transparenz. Die routinemässigen Arbeiten werden mehr und mehr direkt beim Handwerker oder Kleingewerbetreibenden im eigenen Büro erledigt.

Qualifizierte Buchhaltungstätigkeiten, wie sie zum Beispiel der Abschluss, die Bewertungskorrekturen, Steuerangelegenheiten und vieles andere mehr darstellen, werden aber auch in Zukunft vom Fachmann zu erledigen sein, denn ihre Indi-

vidualität erlaubt keine wirtschaftliche Erfassung in einem Computerprogramm.

Da einerseits die Buchführung für Nichtfachleute etwas Neues darstellt, und andererseits der Buch-

halter oder Treuhänder fundierte, den Gesetzen entsprechende Aufzeichnungen zur korrekten Erledigung seiner Aufgaben benötigen, versuchen wir einen kleinen Ueberblick der diesbezüglich existierenden Vorschriften und Richtlinien zusammenzufassen.

Unsere Hinweise unterteilen wir in drei Hauptpunkte:

1. Das Gesetz, die entsprechende Verordnung und die bezüglichen Richtlinien.
2. Hinweise und Anregungen zu den unter 1.) genannten Punkten.
3. Beispiel einer Buchhaltung.

Es ist nicht die Absicht dieses Artikels, alle geltenden Gesetze, Verordnungen und Richtlinien, Abschnitt für Abschnitt wiederzugeben, sondern es soll versucht werden, die Punkte zu fixieren, welche beim Kauf eines Buchhaltungsprogrammes als Anforderungsprofil genommen werden sollten.

1. Das Gesetz, die entsprechende Verordnung und die bezüglichen Richtlinien

Die Artikel 962 & 963 des Schweizerischen Obligationenrechtes regeln die grundsätzliche Pflicht zur Aufbewahrung der Geschäftsbücher und die sogenannte Editions-pflicht.

Die Verordnung über die Aufzeichnung von aufzubewahrenden Unterlagen vom 2. Juni 1976 geht tiefer, indem sie in den Artikeln 4 bis 6 und 9 wesentliche Anforderungen an ein Buchhaltungsprogramm stellt:

Kontoplan vom 30.06.81		DIALOG-COMPUTER TREUHAND AG		Seite: 1
-----		Superbrain Buchhaltung		
Konto	Bezeichnung	Budget Soll	Budget Haben	Bemerkungen
<b>Aktiven</b>				
102	Fluessige MITTEL			
1021	SPARKTO HAUS 900.436			
1022	SPARKTO GARAGE 900.612			
1040	WERTSCHRIFTEN			
1051	KTO KRT STWE GEMEINSCH.			
1060	VST-GUTHABEN			
1070	UEGRIGE AKTIVEN			
<b>Passiven</b>				
20	Fremdkapital			
2000	KREDITOREN			
21	Eigenkapital			
2100	E'FONDS HAUS			
2110	E'FONDS EINSTELLHALLE			
<b>Uebriger Aufwand</b>				
4010	REP.-AUFWAND HAUS			
4020	REP.-AUFWAND E'HALLE			
4200	BANKSPESEN			
<b>Kostenstellen</b>				
5000	Musterkostenstellen			
<b>Ausserord. Aufwand/Ertrag</b>				
70	TOTAL Einlagen G'jahr			
7010	EINLAGEN HAUS		18000	
7020	EINLAGEN EINSTELLHALLE		2250	
71	TOTAL Zi'ertrag fl.Mittel			
7110	ZINSERTRAG HAUS			
7120	ZINSERTRAG EINSTELLHALLE			
7210	WERTSCHRIFTERTRAG			
7220	KURSGEWINNE A/WERTSCHR.			
9000	E'bilanzkorrekturen			

Datum	Bu.Nr	Beleg	Konto Soll	Konto Haben	Ko. St.	Bed	Buchungstext	K Z	Betrag
22.06.80	1	1	1021	7210			CB Cps.SBG 3.25 %		162.50
22.06.80	2	1	1060	1021			CB VST A/3.25 % SBG		56.90
15.07.80	3	2	1021	7210			CB CPS. 3.25 % SBG		175.00
15.07.80	4	2	1060	1021			CB VST A/3.25 % SBG		61.25
15.07.80	5	3	1021	7210			CB CPS. 4.5 % BASEL-LAND	-	2255.00
15.05.80	6	3	1060	1021			CB VST A/4,5 % BASEL-LAND		78.75
28.08.80	7	4	1022	2000			CB C. Rebsamen	-	120.00
30.09.80	8	5	2000	1022			CB C. Rebsamen	-	120.00
05.10.80	9	6	1021	7210			CB CPS. 5 % SBG		250.00
05.10.80	10	6	1060	1021			CB VST A/ 5 % SBG		87.50
28.10.80	11	7	1021	7210			CB CPS. 5 % KANTON LUZERN		500.00
28.10.80	12	7	1060	1021			CB VST A/ 5 % KT LUZERN	-	170175.00
28.10.80	13	7	1022	2000			CB STORNO	S	-120.00
28.10.80	14	8	2000	1022			CB STORNO	S	-120.00
28.10.80	15	12	1060	1021			CB STORNO	S	-170175.00
28.08.80	16	4	1022	2000			CB CB NACHBUCHUNG		120.00
30.09.80	17	5	2000	1022			CB CB NACHBUCHUNG		120.00
28.10.80	18	7	1060	1021			CB VST A/5 % KT LUZERN		175.00
02.12.80	19	89	1021	7210			CB CPS. 4 % SBG		200.00
02.12.80	20	80	1060	1021			CB VST A/4 % SBG		70.00
10.12.80	21	9	1021	7210			CB CPS 4,5 % BASEL LAND		225.00
10.12.80	22	9	1060	1021			CB VST A/ 4,5 % BASEL LAND		78.75
31.12.80	23	10	4200	1021			CB DEPOTGEBUEHR SBG		58.00
31.12.80	24	11	1021	7110			CB ZINS SBG FUER HAUS 31.12.		306.50
31.12.80	25	11	1060	1021			CB VST A/ZINS HAUS SBG		107.30
31.12.80	26	12	1022	7120			CB ZINS SBG FUER E'HALLE		130.65
31.12.80	27	12	1060	1022			CB VST A/ZINS E'HALLE SBG		45.75
31.12.80	28	12	1021	1060			CB VST RUECKERSTATTUNG		935.55
15.03.81	29	13	1021	7210			CB CPS 5.75 % KT ZUERICH		287.50
15.03.81	30	13	1060	1021			CB VST A/5.75 % KT ZUERICH		100.65
31.02.81	31	14	1021	7210			CB CPS 4 % KT LUZERN		400.00
31.03.81	32	14	1060	1021			CB VST/A 4 % KT LUZERN		140.00
24.04.81	33	15	1021	7210			CB CPS 3.25 % SBG		162.50
24.04.81	34	15	1060	1021			CB VST A/3.25 % SBG		56.90
20.06.81	35	16	4010	1021			CB BOILERREP./UNTERSUCH		13697.30
20.06.81	36	17	4010	1021			CB FA J. OTTIGER B'LEITUNG		742.00
15.07.80	37	5	1021	7210			CB STORNO	S	-2255.00
15.07.80	38	37	1021	7210			CB CPS 4.5 % KT BASEL LAND		225.00
30.06.81	39	18	1051	7010			CB ZUWEISUNG GESCHAFTSJAHR		15000.00
30.06.81	40	19	1051	7020			CB ZUWEISUNG G'JAHR		1000.00
30.06.81	41	20	1051	7020			CB ZUWEISUNG MUELLER GJ		500.00
30.06.81	42	20	2000	9000			CB Korr. Eingangsbilanzd cb		120.00

nungen von Geschäftsbüchern für die vom Berechtigten bestimmte Zeitspanne bereitzuhalten.

Die Richtlinien für die Erstellung von Arbeitsanweisungen zur Aufzeichnung und Aufbewahrung von Unterlagen auf Datenträgern spezifizieren die Verordnung in einigen Punkten nochmals.

Ziffer 5/Ausdruck oder Aufzeichnung von Geschäftsbüchern

5.1 Die Geschäftsbücher sind periodisch, mindestens auf Ende des Geschäftsjahres auszudrucken oder auf einen Bildträger aufzuzeichnen.

5.2 Diese Ausdrücke und Aufzeichnungen bilden den Gegenstand der gesetzlichen Aufbewahrungspflicht.

5.3 Für Sammelbuchungen müssen jederzeit die Einzelbeträge nachweisbar sein.

5.4 Daten, die ausgedruckt vorliegen und so aufbewahrt werden, können auf dem Datenträger gelöscht werden.

#### Art. 4/Verfügbarkeit

Der Aufbewahrungspflichtige sorgt dafür, dass die Aufzeichnungen während der Aufbewahrungsfrist jederzeit verfügbar sind und ohne Schwierigkeiten gelesen werden können. Die Prüfung der Aufzeichnungen darf nicht schwieriger sein oder mehr Zeit beanspruchen als die Prüfung der Unterlagen.

#### Art. 9/Wiedergabe (Auszug)

Der Einsichtsberechtigte kann verlangen, dass die von ihm bestimmten Unterlagen in unmittelbar lesbarer Form vorgelegt werden. Für eine Buchprüfung ist die Aufzeich-

#### Ziffer 6/Dokumentation (Auszug)

Die Dokumentation kann in textlicher, grafischer oder tabellarischer Form erfolgen. Die Dokumentation ist so lange aufzubewahren wie die letzte mit ihr erstellte Aufzeichnung.

#### Art. 5/Angabe der Firma

Die Aufzeichnungen geben die aufbewahrungspflichtige Firma an.

#### Art. 6/Aufzeichnungsverfahren

Den Aufzeichnungen werden die folgenden Angaben beigefügt:

- Namen der mit der Aufzeichnung betrauten Personen;
- Art und Umfang der aufgezeichneten Unterlagen;
- Ort und Datum der Aufzeichnung;
- während der Aufzeichnung oder Aufbewahrung festgestellte Beschädigungen an Unterlagen und Bild- und Datenträgern.

Konto 1021 SPARKTO HAUS 900.436

Datum	Beleg	G.kto	Buchungstext	Soll	Haben
22.06.80	1	7210	Cps.SBG 3.25 %	162.50	
22.06.80	1	1060	VST A/3.25 % SBG		56.90
15.07.80	2	7210	CPS. 3.25 % SBG	175.00	
15.07.80	2	1060	VST A/3.25 % SBG		61.25
15.05.80	3	1060	VST A/4,5 % BASEL-LAND		78.75
05.10.80	6	7210	CPS. 5 % SBG	250.00	
05.10.80	6	1060	VST A/ 5 % SBG		87.50
28.10.80	7	7210	CPS. 5 % KANTON LUZERN	500.00	
28.10.80	7	1060	VST A/5 % KT LUZERN		175.00
02.12.80	89	7210	CPS. 4 % SBG	200.00	
02.12.80	80	1060	VST A/4 % SBG		70.00
10.12.80	9	7210	CPS 4,5 % BASEL LAND	225.00	
10.12.80	9	1060	VST A/ 4,5 % BASEL LAND		78.75
31.12.80	10	4200	DEPOTGEBUEHR SBG		58.00
31.12.80	11	7110	ZINS SBG FUER HAUS 31.12.	306.50	
31.12.80	11	1060	VST A/ZINS HAUS SBG		107.30
31.12.80	12	1060	VST RUECKERSTATTUNG	935.55	
15.03.81	13	7210	CPS 5.75 % KT ZUERICH	287.50	
15.03.81	13	1060	VST A/5.75 % KT ZUERICH		100.65
31.02.81	14	7210	CPS 4 % KT LUZERN	400.00	
31.03.81	14	1060	VST/A 4 % KT LUZERN		140.00
24.04.81	15	7210	CPS 3.25 % SBG	162.50	
24.04.81	15	1060	VST A/3.25 % SBG		56.90
20.06.81	16	4010	BOILERREP./UNTERSUCH		13697.30
20.06.81	17	4010	FA J. OTTIGER B'LEITUNG		742.00
15.07.80	37	7210	CPS 4.5 % KT BASEL LAND	225.00	
			Total	3829.55	15510.30
			Saldo	11680.75	
			Anfangs-Saldo	13122.40	
			Schluss-Saldo	1441.65	

Konto Bezeichnung	Soll	Haben	Aufwand	Ertrag
4010 REP.-AUFWAND HAUS	14439.30	---	14439.30	
4020 REP.-AUFWAND E'HALLE	---	---		
4200 BANKSPESEN	58.00	---	58.00	
Total Uebrigter Aufwand	14497.30	---	14497.30	
7010 EINLAGEN HAUS	---	15000.00		15000.00
7020 EINLAGEN EINSTELLHALLE	---	1500.00		1500.00
Summe TOTAL Einlagen G'jahr	---	16500.00		16500.00
7110 ZINSERTRAG HAUS	---	306.50		306.50
7120 ZINSERTRAG EINSTELLHALLE	---	130.65		130.65
Summe TOTAL Zi'ertrag fl.Mittel	---	437.15		437.15
7210 WERTSCHRIFTENERTRAG	---	2587.50		2587.50
7220 KURSGEWINNE A/WERTSCHR.	---	---		
Total Ausserord. Aufwand/Ertrag	---	19524.65		19524.65
Total und Gewinn/Verlust	14497.30	19524.65	5027.35	

### 3. Beispiel einer Buchhaltung, eine mögliche Lösung

Die Listenbilder dieser Buchhaltung stammen von einem Finanzbuchhaltungsprogramm, welches den oben genannten Anforderungen entspricht. Das Programm wurde für einen 64 K Dialog-Small-business Computer mit zwei Floppy-Laufwerken geschrieben. Mit den ungefähr 350 möglichen Konti und den 2200 Buchungen wird einem grossen Anwenderbereich Genüge getan. Programmtechnische Vorkehrungen lokalisieren und signalisieren auftretende Fehlerquellen sofort.

Journal, Kontoblätter mit separaten Seiten pro Konto, Umsatz- und Saldo-Bilanz, Gewinn- und Verlustrechnung, Budget- und Vorjahresvergleich (Option) haben informationsbewusste Unternehmer und Treuhänder jederzeit sauber auf A4-hoch-Format ausgedruckt verfügbar.

Die Dateneingabe kann während des Arbeitsprozesses am Bildschirm verfolgt werden. Durch verschiedene Bestätigungsmöglichkeiten während der Eingabe können Fehler frühzeitig erkannt werden. Korrekturen in den Files sind daher beinahe unnötig. Durch die im Programm verankerte Doppik (Soll gleich immer Haben) ist ein korrektes Journal am Ende einer Buchungsperiode immer gewährleistet.

### Ziffer 9/Buchprüfung (Auszug)

9.1 Die Ordnungsmässigkeit der Aufzeichnungen muss von einem sachverständigen Dritten leicht und zuverlässig überprüft werden können.

9.2 Die Prüfung muss direkt - vom Einzelbeleg über Journal und Konto bis zur Bilanz bzw. Betriebsrechnung und umgekehrt - sowie allenfalls indirekt - z.B. durch Vergleich - vorgenommen werden können.

9.4 Andere auf Datenträger aufgezeichnete Unterlagen sind auf Verlangen eines Berechtigten unverzüglich auszudrucken.

2. Hinweise und Anregungen zu den unter 1. genannten Punkten.

Obschon wir nur die uns wesentlich erscheinenden Ziffern und Artikel wiedergegeben haben, ist deren Anzahl nicht gering. Die Frage nach dem Sinn der Aufbewahrungspflicht stellt sich zwingend.

Die Aufbewahrungspflicht kommt erst in einem Prozessfall zum "Zuge". Die zu produzierenden Beweismittel müssen den aufbewahrten Dokumenten entnommen werden können und Auskunft geben über:

- wie, ob und wann die einzelnen Geschäfte entstanden sind
- die Entwicklung der Geschäfte
- eventuelle Friktionen bei der Abwicklung des Geschäftes

- ob und wann die ausgehandelten Verpflichtungen erfüllt wurden

Ohne eine lückenlose Aufbewahrung von Unterlagen während der vom Gesetz vorgeschriebenen Frist kann eine Beweisführung oft sehr schwer oder unmöglich werden.

Durch eine strengere Handhabung der Buchführungspflicht könnte viel menschliches Leid und volkswirtschaftlicher Schaden vermieden werden.

Da die Buchhaltung das Kernstück der Geschäftsbücher darstellt, kann durch die korrekte Buchhaltungsorganisation schon viel zum voraus festgelegt werden. Zahlreiche Daten und Dokumente müssen beschafft und notiert werden, damit die Buchhaltung richtig geführt werden kann. Die Gefahr der Ueberorganisation lauert aber nirgends so prompt wie hier.

Bilanz per 30.06.81 DIALOG-COMPUTER TREUHAND AG  
----- Superbrain Buchhaltung Seite: 1

Konto Bezeichnung	Soll	Haben	Aktiven	Passiven
1021 SPARKTO HAUS 900.436	3829.55	15510.30	1441.65	
1022 SPARKTO GARAGE 900.612	250.65	165.75	5724.20	
Kumul Flüssige MITTEL	4080.20	15676.05	7165.85	
1040 WERTSCHRIFTEN	---	---	60000.00	
1051 KTO KRT STWE GEMEINSCH.	16500.00	---	38100.00	
1060 VST-GUTHABEN	1058.75	935.55	1058.75	
1070 UEGRIGE AKTIVEN	---	---		
Total Aktiven	21638.95	16611.60	106324.60	
2000 KREDITOREN	240.00	120.00		
Summe Fremdkapital	240.00	120.00		
2100 E'FONDS HAUS	---	---		92938.70
2110 E'FONDS EINSTELLHALLE	---	---		8358.55
Summe Eigenkapital	---	---		101297.25
Total Passiven	240.00	120.00		101297.25
Total und Gewinn/Verlust	21878.95	16731.60		5027.35

# BÖRSE

Verkaufe Microcomputer DDP85, Prozessor 8085, Drive 2x600KB 64K RAM, sep. Bildschirm 1920Z ev. mit Drucker NEC 5510, für Basic (C/M/E), Cobol, Fortran, u.a. Zustand neuwertig. Nähere Auskunft: Tel. 031 94 77 88

HEATH H8-Comp. 16K, RS232, 4 I/O-Ports Softw. (Kassette): BASIC, ASSEMBLER, EDITOR, DEBUG, kpl. Dok. Fr. 1400.— (erweiterbar bis 65K, Floppy mögl.). HEATH H9-Vid. Term. Fr. 700.—. Suche: HEATH-Floppy Z87 Tel. 052 22 91 79 ab 18 Uhr

Zu verkaufen: KIENZLE 6100/5 64 KB, Doppelkassettenstation, 3 Diskdrives zu 8,3 MB, Zeilendrucker 360 Z/Sek., 3 Bildschirme, 3 Hardcopydrucker 100 Z/Sek. Cendres & Metaux SA, 2500 Biel, Tel. 032 41 51 51

Schreibmaschinen-Interface (nach Funkschau), Bausatz Fr. 300.— Bruno Spoerri, Schneckenmannstrasse 27, 8044 Zürich

Zu verkaufen wegen Nichtgebrauch: HP 41C mit 2 Modulen, Magnetkartenleser, Akku + Netzteil, zusammen für Fr. 1000.— (NP Fr. 1400.—). Einwandfreier Zustand. Tel. 01 810 18 48, Hr. Buckmann verlangen.

Verk. Dis+Assembler, Editor für PET 2001/CEM (neue OS Vers.). Toolkit EPROM oder Kasset. Vers. Fr. 120.— Drucker Schnittstelle. J.B. Boichat, Stapfenstr. 71, 3018 Bern

Zu verk. 1 Microdata System mit 32K Basic + Interpreter-Compiler, div. Software, 2 Floppy Drive, 1 MByte Terminal + Printer, Interface+ 1ADM3 Terminal+ 1 Printer Teletype 43. Alles für Fr. 14'000.— VB. Tel. 031 91 25 25/44 33 03

Verkaufe: Motorola ADS-1 bestehend aus: Stand alone Computer, Keyboard, 5" Video-Monitor mit Interf., Handbuch, Gehäuse, Netzteil. Neupreis über Fr. 3000.—, jetzt nur noch Fr. 950.—! Tel. 01 242 50 30 intern 55

Für Bastler!

Zu verkaufen IBM-Konsolenschreibmaschinen (Kugelpopf) zu IBM/360 ohne power supply Fr. 350.— Tel. 041 30 11 66 (Herrn M. Weber verlangen)

Verkaufe HP41C + Leser + Drucker + 2 Memory Mod + 120 Magnetkarten, nur en bloc, Fr. 1900.— Tel. 061 76 95 41 ab 18.30

Zu verkaufen 6 Monate alt:  
32 K CEM inkl. Toolkit Fr. 2530.—  
CEM Dual Floppy 3040 Fr. 2530.—  
CEM Printer Fr. 1650.—  
CEM Recorder Fr. 180.—  
Tel. 061 P: 63 52 96 G: 35 01 30  
Hr. Winter verlangen

Verk. Techtran Floppy-Disc 950, kaum gebr., geprüft, V24-Anschluss, mit Mikroproz. gesteuert. (Neupreis Fr. 3200.—), inkl. 1 Diskette, V24-Kabel und Anleitung. Preis Fr. 2200.—. Anfragen ab 19.00 Tel. 01 40 76 23

Sofort: HP-41C mit Kartenleser, Thermodrucker, Handbuch, 2 Speicher-Moduln, Statistic Pac, 120 blank Magnetkarten. Zusammen Fr. 2000.— mit Garantie. Manfred Dechmann, Herrenbergstr. 1, 8006 Zürich, Tel. 01 361 54 88

WOLLEN SIE MEHR WISSEN  
WAS DIE KLEINCOMPUTER  
ALLES KOENNEN?

Dann bestellen Sie sich noch heute Ihr ganz persönliches Jahresabonnement MIKRO- UND KLEINCOMPUTER; das sind sechs aufeinanderfolgende Ausgaben, ohne Umweg direkt an Ihre Postanschrift. Oder noch besser - werden Sie Mitglied beim Schweizer Computer Club; das Abonnement ist dann als Clubleistung im Mitgliederbeitrag enthalten. Anmeldekarten finden Sie am Anfang und Schluss dieses Heftes.

VERLAG SCC AG

Zu verkaufen: EPROM Programmer für CEM-Computer, für 24 Pin/5 Volt EPROMS. Prg. Rechner HP97S mit Daten-Interface Tel. 055 31 72 30 (Poltera)

PET 2001 mit Ausbau 40K und Compu-Think Floppy 2x200K (Double Density) zu verkaufen Fr. 3000.— zusammen. A. Ritter, Florastr. 22, 3800 Interlaken, Tel. 036 22 86 22 P + G

SORCERER-Besitzer (Ostschweiz) wünscht Erfa-Austausch. Meine Interessen: Software (allg./Statistik/Spiele...), Interfacing usw. Bin Amateur mit Grundkenntnissen. Tel. G 071 41 53 13 (Hr. Bänziger)

Zu verk.: HP-41C mit 3 Speichermodulen, 6 Mt. alt, engl. Manual noch 6 Mt. Garantie Fr. 600.— (mit Card-Reader Fr. 1000.—) Tel. 061 95 22 35 abends (Max Thommen jun. verlangen)

ITT/MIKROPROZESSOR Lehrsystem inkl. 5 Lehrhefte, neuwertig Fr. 800.— (neu Fr. 1370.—) Tel. 063 22 89 74

Zu kaufen gesucht: Günstiger Heimcomputer, PET, CEM, APPLE o.a. ev. mit Drucker. Gleichzeitig zu verkaufen: 1 TI 59 mit Drucker. Tel. 041 89 21 69

Biete an: Informationen zum TI 59, die nicht im Handbuch stehen (+ Tips, Tricks, Interpretation des CROM etc.) 7.— DM auf Pschiko. 1438 69-752, Karlsruhe, BRD, Vermerk: INFO1, Marius Heyn, Kapplerbergstr. 62A, D-7753 Allensbach

Zu verkaufen: - preiswert - ADDO-Forster Schreibautomat, 2 Magnetbandstationen, 1 El. Schreibmaschine, 1 Textverarbeitungseinheit, div. Material Tel. 061 76 17 51

EINE NEUE LEISTUNG DES SCC  
FUER SEINE MITGLIEDER

Ab sofort kostet ein PRIVATES KLEININSERAT in der BOERSE von Mikro- und Kleincomputer für SCC-Mitglieder nur noch Fr. 20.-- (für Nichtmitglieder jetzt Fr. 40.--).

- Haben Sie etwas zu verkaufen?
- Suchen Sie eine günstige Occasion?
- Wollen Sie neue Kollegen finden?
- Möchten Sie etwas tauschen?

Das ist jetzt ganz einfach. Füllen Sie die beigeheftete Karte für Kleininserate aus (maximal sieben Zeilen zu 30 Zeichen) und senden Sie die Karte plus eine Zwanzig-Franken-Note (Nichtmitglieder zwei Zwanzig-Franken-Noten) an den Verlag SCC AG. Ihr Inserat erscheint in der nächsterreichbaren Ausgabe.

SCHWEIZER COMPUTER CLUB

# Lehrgänge



## Turtlegraphics

Dr. Bruno STANEK

Jedermann, der sich intensiv mit der Grafik-Implementierung in APPLE-PASCAL, den "TURTLEGRAPHICS", beschäftigt hat, bedauerte früher oder später die Schwierigkeit, schöne Grafiken vom Bildschirm auf das Papier zu bannen. Hier wird nun eine PASCAL-Software vorgestellt, mit der Programme leicht vom Display auf einen Printer umgeschrieben werden können, allerdings in Abweichung von echten Turtlegraphics in rechtwinkligen Koordinaten.

Die Programme sind auf einem Superbrain geschrieben, doch laufen sie nach einigen Änderungen auch auf einem Apple. Der wesentliche Hauptunterschied besteht darin, dass man den Output auf dem Superbrain direkt vom Programm aus zum Printer lenken kann:

```
VAR PRINTER: TEXT;  
REWRITE('LST:',PRINTER);  
WRITE(PRINTER,CHAR1,CHAR2,...);
```

Dabei bedeuten CHAR1, CHAR2... irgendwelche Zeichen zur Steuerung des Printers. Auf dem APPLE muss dagegen zuerst ein ASCII-File erzeugt werden, das dann vom PASCAL-Betriebssystem (Filer) auf das PRINTER-Standardfile transferiert werden kann.

Der inkrementelle Printer NEC (siehe auch m+k computer 80-5) kann Punkte im horizontalen Abstand von 1/120 Zoll und im vertikalen Abstand von 1/48 Zoll drucken. Auf einem hoch eingespannten A3-Papier können somit gut 1200 Punkte in der Breite und wegen der geringeren Auflösung in Richtung des Papiertransports "nur" 720 in der Länge dargestellt werden. Spannt man das Papier dagegen quer ein, hat man 1800 Punkte in der Länge und 480 in der Breite. Um hier nicht jedesmal die Skalierung selber vornehmen zu müssen, werden die Punktezahlen gerade von der Initialisierungs-Prozedur Grafik berechnet:

```
GRAFIK(BLATT: PAPIER;  
EX,EY: INTEGER;  
VAR SX,SY: INTEGER);
```

Dabei gilt die Typendeklaration

```
TYPE PAPIER=(A3HOCH,A3BREIT,  
A4HOCH,A4BREIT,  
A5HOCH,A5BREIT);
```

EX und EY bedeuten die Längeneinheiten, gemessen im minimalen Inkrement des Druckers. Oft wird EX=5 und EY=2 gewählt, weil dies den feinsten quadratischen Raster ergibt: 5/120 Zoll = 2/48 Zoll = 1 mm. Die Ausgabeparameter SX und SY sind die resultierende Auflösung in X und Y.

Sind die Koordinaten einmal gewählt, stehen eine Reihe von elementaren Standardprozeduren zur Verfügung:

```
PROCEDURE PUNKT(X,Y: INTEGER;  
ZEICHEN: CHAR);  
PROCEDURE HORIZONTALA  
(X1,X2,Y: INTEGER);  
PROCEDURE VERTIKALE  
(X,Y1,Y2: INTEGER);  
PROCEDURE GERADE  
(X1,Y1,X2,Y2: INTEGER);  
PROCEDURE RAHMEN  
(OLX,OLY,URX,URY);
```

Bei allen diesen Koordinaten handelt es sich um absolute, die auf den Nullpunkt (auf dem Blatt oben links) bezogen sind. Dort wird auch die Anfangsposition initialisiert.

In der Prozedur PUNKT kann ein beliebiges Zeichen positioniert

werden, alle Linien dagegen werden immer aus Punkten (CHR(46)) aufgebaut. Ein rechteckiger Rahmen wird durch die Koordinaten der oberen linken und der unteren rechten Ecke festgelegt.

Etwas interessanter ist das Aequivalent zur Prozedur DRAWBLOCK in APPLE-PASCAL. Mit dieser kann ein durch die "Bitmap" definierter Raster an beliebiger Stelle des Bildschirms plaziert werden.

```
TYPE BITMAP=ARRAY[1..50,1..50];
```

definiert eine logische Matrix, deren Elemente angeben, ob an dem entsprechenden Rasterelement eines 50x50-Quadrates ein Punkt steht (TRUE) oder nicht (FALSE).

```
PROCEDURE BLOCK(BILD: BITMAP;  
X,Y,IX,IY: INTEGER);
```

sorgt dafür, dass der durch die Variable BILD definierte Raster mit der Ecke 1,1 bei den Koordinaten (X,Y) gezeichnet wird. Mit IX, IY (1<=IX<=X, 1<=IY<=Y) kann auf Wunsch nur ein Teil davon dargestellt werden. Da BILD in einer ganz fremden Prozedur definiert sein kann, lassen sich grafische Programme mit minimalem Aufwand für den NEC-Printer nutzbar machen. So war es z.B. mit der

```
PROCEDURE KRATER(GROESSE: INTEGER;  
VAR BILD: BITMAP);
```

die in m+k computer 80-3, S.42, damals noch in BASIC, beschrieben war. In kürzester Zeit entstand so ein anschauliches Testbeispiel (vgl. die beigefügte Abbildung).



Oft will man aber eine Figur explizit definieren, z.B. ein neues Symbol, das auf Printern nicht zu finden ist, wie etwa das astronomische Zeichen von Jupiter. Eine Hilfsprozedur

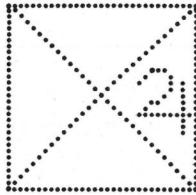
```
PROCEDURE GEN(VAR BILD: BITMAP;
              Z: INTEGER; S: ZEILE);
wobei
TYPE ZEILE=STRING 50;
```

generiert die Zeile Nr. Z der BITMAP aus einem STRING, welcher die Bildinformation über diese Zeile des Rasters enthält. Sie gestattet eine auch für die eigene Dokumentation äusserst angenehme Definition des Zeichens als Prozedur, die vom Benutzer geschrieben wird. Hier zum Beispiel:

```
PROCEDURE JUPITER
(VAR SYMBOL: BITMAP);
BEGIN
GEN(SYMBOL, 1, ' XXX  ');
GEN(SYMBOL, 2, ' X  X  ');
GEN(SYMBOL, 3, 'X   X X ');
GEN(SYMBOL, 4, '  X X ');
GEN(SYMBOL, 5, '   X X ');
GEN(SYMBOL, 6, '  XX  X ');
GEN(SYMBOL, 7, '  X   X ');
GEN(SYMBOL, 8, 'XXXXXXXXX');
GEN(SYMBOL, 9, '   X  ');
GEN(SYMBOL,10, '    X ');
GEN(SYMBOL,11, '     X ');
END;
```

Als Zusammenfassung des bisher gesagten mag die folgende Reihe von Anweisungen dienen, die das Zeichen von Jupiter in einem quadratischen Rahmen samt Diagonalen zeigt. Man beachte die Genauigkeit, die trotz zeitweise rückwärtstransportiertem Papier erreicht wurde.

```
JUPITER(SYMBOL);
BLOCK(SYMBOL,52,51,9,11);
RAHMEN(40,40,70,70);
GERADE(40,40,70,70);
GERADE(70,40,70,40);
```



Man stellt hier bereits die Grenzen fest: Es ist zweifellos nicht sinnvoll, mit solchen punktierten Geraden einen Plotter konkurrenzieren zu wollen, doch gibt es viele andere Fälle von komplexeren Rastern, die sich mit dem Printer in höchster Qualität druckfertig erzeugen lassen.

Zum Abschluss sei noch eine weitere Konstruktion der NEC-Grafik-Software genannt. Hier geht es um die solide "ausgemalte" Darstellung von Figuren, die nur durch ihre Umrisse gegeben sind. Die gesamte Information über die Umrisse lässt sich in PASCAL als RECORD darstellen: obere Grenze, untere Grenze und dazwischen je ein ARRAY für die linke bzw. rechte Berandung:

```
TYPE FORM=RECORD
TOP,BOT: INTEGER;
LM, RM: ARRAY[-200,200]
OF INTEGER;
END;
```

Da man gelegentlich nur Teile der Figur zeichnen will, definiert man noch z.B.

```
TYPE SYMMETRIE=(KEINE,OBEN,UNTEN,
LINKS,RECHTS,AXIAL);
```

Neben den vier wohl selbstverständlichen Hälften bedeutet AXIAL, dass eine doppelt berandete Figur (LM jetzt innerer und RM äusserer Rand) in axialer Symmetrie gemeint ist.

Sei nun

```
VAR F: FORM;
```

eine vom Benutzer definierte Datenstruktur der Umrandung. Zur Darstellung an der Stelle (X,Y) dient ihm die

```
PROCEDURE FIGUR(F: FORM;
S: SYMMETRIE;
X,Y: INTEGER);
```

Will man etwa eine Kugel haben, die in der Projektion als Kreisscheibe erscheint, definiert man selber:

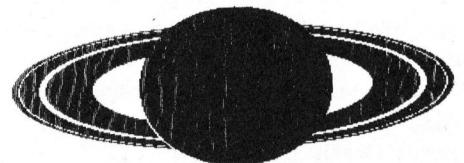
```
PROCEDURE KUGEL(R: INTEGER;
VAR RUND: FORM);
VAR Y: INTEGER;
Q: REAL;
BEGIN
WITH RUND DO
BEGIN
TOP:=R; BOT:=-R; Q:=SQR(R);
FOR Y:=BOT TO TOP DO
BEGIN
RM[Y]:=ROUND(SQRT(Q-SQR(Y)));
LM[Y]:=-RM[Y];
END;
END;
END;
```

Wir wollen dem Leser das Vergnügen nicht vorenthalten, sich eine eigene

```
PROCEDURE RING(R,R1,R2: INTEGER;
NEIGUNG: REAL;
VAR SICHTBAR: FORM);
```

zur Darstellung von Planetenringen in allen möglichen Perspektiven zu schreiben. Das Programm für die beigefügte Abbildung von Saturn lässt sich damit bequem formulieren:

```
GRAFIK(A4HOCH,3,1,SX,SY);
DX:=SX DIV 2; DY:=SY DIV 2;
KUGEL(60,SATURN);
FIGUR(SATURN,KEINE,DX,DY);
RING(60,88,117,20.0,ARING);
FIGUR(ARING,AXIAL,DX,DY);
RING(60,122,133,20.0,BRING);
FIGUR(BRING,AXIAL,DX,DY);
RING(60,135,138,20.0,FRING);
FIGUR(FRING,AXIAL,DX,DY);
```



## Plotroutinen für Typenradrunder

Leopold ASBÖCK

In m+k computer 80-2 und 80-5 brachten wir Beschreibungen zum Typenradrunder Spinwriter von NEC. Dieser Beitrag soll nun all jenen Hilfe und Anregung bieten, die mit diesem universellen Drucker mit hochauflösender Grafik arbeiten oder plotten wollen. Die wichtigsten Druckerfunktionen werden in BASIC-Unterprogrammen und PASCAL-Prozeduren "gebrauchsfertig" zur Verfügung gestellt.

Um die Kommunikation Mensch/Computer/Peripherie in Fluss zu bringen, bedarf es einer genormten Grundlage für den Informationsaustausch. Häufig treffen wir den ASCII-Code an (American Standard Code for Information Interchange), der eine geeignete Basis darstellt.

Unsere Kommunikationselemente (Buchstaben, Ziffern, Sonderzeichen) müssen dem Computer in binärer Form übermittelt werden. Umgekehrt muss der Computer seine binären Resultate in Symbole umsetzen, die für uns verständlich sind.

		b6-	0	0	0	0	1	1	1	1
		b5-	0	0	1	0	1	0	0	1
		b4-	0	0	1	0	1	0	1	0
b3	b2	b1	b0							
0	0	0	0	0	0	NUL	DLE	SP	@	P
0	0	0	1	1	1	SOH	DC1	!	A	Q
0	0	1	0	2	2	STX	DC2	"	B	R
0	0	1	1	3	3	ETX	DC3	#	C	S
0	1	0	0	4	4	EOT	DC4	\$	D	T
0	1	0	1	5	5	ENQ	NAK	%	E	U
0	1	1	0	6	6	ACK	SYN	&	F	V
0	1	1	1	7	7	BEL	ETB	'	G	W
1	0	0	0	8	8	BS	CAN	(	H	X
1	0	0	1	9	9	HT	EM	)	I	Y
1	0	1	0	10	A	LF	SUB	*	J	Z
1	0	1	1	11	B	VT	ESC	+	K	[
1	1	0	0	12	C	FF	FS	,	L	\
1	1	0	1	13	D	CR	GS	-	M	]
1	1	1	0	14	E	SD	RS	>	N	^
1	1	1	1	15	F	SI	US	/	?	o

Abb.1 ASCII-Tabelle

Jedem Buchstaben und jeder Ziffer lässt sich eine Folge von Binärziffern (0 oder 1) zuordnen. Acht Binärziffern oder Bits lassen 256 verschiedene Folgen zu. Für die 96 gebräuchlichsten Zeichen reichen bereits 7 Bits, weiters bleiben 32 Kombinationen für spezielle Befehlscodes.

### ASCII-Code

Aus Abb.1 und Abb.2 sind die Zuordnungen in hexadezimaler und dezimaler Form ersichtlich. Die ersten 32 Codes (00 bis 1F) sind nichtdruckbare Zeichen und dienen der Befehlsübermittlung. Ihre Bedeutung ist aus Abb.3 ersichtlich.

So wird zum Beispiel ØD (hex) vom Drucker oder vom Terminal als CR (Carriage Return, Wagenrücklauf) interpretiert, ØA (hex) als LF (Line Feed, Zeilenvorschub).

Dieser Befehlsvorrat reicht aber unter Umständen bei intelligenten Peripheriegeräten, die über viele Funktionsabläufe verfügen, nicht aus. Es ist dann notwendig, nicht nur ein einzelnes ASCII-Zeichen, sondern eine wohldefinierte Folge von Zeichen zu übermitteln, um die gewünschte Funktion auszulösen.

So reagiert der NEC-Spinwriter, ein bewährter Typenradrunder japanischer Provenienz, auf über 350 Befehle, wobei der grösste Anteil aus Tabulatorbefehlen besteht. Ueber diesen Befehlssatz ist es beispielsweise möglich, hochauflösende Grafik zu drucken, von rechts nach links zu drucken oder die Farbe am Farbband zu wechseln.

Das achte Bit (Bit b7) zieht man häufig als Paritätsbit heran, um bei störempfindlichen Datenübertragungen eine einfache Methode zum Erkennen von fehlerhaften Bits zu erhalten. Bei "even parity" wird dieses Bit so ergänzt, dass die Quersumme (= Summe aller acht Bits) gerade, bei "odd parity" ungerade ist.

Eine Kontrolle erfolgt am einfachsten per Software, es gibt aber auch Schaltkreise, die als "parity generator" bzw. "parity check" hardwaremässig die notwendigen logischen Verknüpfungen ausführen.

Natürlich müssen bei Datenübertragung "Sender" und "Empfänger" auf gleiche Parität gesetzt werden (parity even, parity odd, mark), andernfalls kommt es zu Fehlermeldungen oder Systemblockierungen.

	0	1	2	3	4	5	6	7	8	9
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT
10	LF	VT	FF	CR	SO	SI	DLE	DC1	DC2	DC3
20	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS
30	RS	US	SP	!	"	#	\$	%	&	'
40	(	)	*	+	,	-	.	/	0	1
50	2	3	4	5	6	7	8	9	:	;
60	<	=	>	?	@	A	B	C	D	E
70	F	G	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W	X	Y
90	Z	[	\	]	^	_	`	a	b	c
100	d	e	f	g	h	i	j	k	l	m
110	n	o	p	q	r	s	t	u	v	w
120	x	y	z	{		}	~	DEL		

Abb.2 ASCII-Tabelle (dezimal)

Control-Taste	hex	ASCII-Befehlscode
CTRL @	00	NUL Null
CTRL A CTRL a	01	SOH Start of Heading
CTRL B CTRL b	02	STX Start of Text
CTRL C CTRL c	03	ETX End of Text
CTRL D CTRL d	04	EOT End of Transmission
CTRL E CTRL e	05	ENQ Enquiry
CTRL F CTRL f	06	ACK Acknowledge
CTRL G CTRL g	07	BEL Bell
CTRL H CTRL h	08	BS Backspace
CTRL I CTRL i	09	HT Horizontal Tab
CTRL J CTRL j	0A	LF Line Feed
CTRL K CTRL k	0B	VT Vertikal Tab
CTRL L CTRL l	0C	FF Form Feed
CTRL M CTRL m	0D	CR Carriage Return
CTRL N CTRL n	0E	SO Shift Out
CTRL O CTRL o	0F	SI Shift In
CTRL P CTRL p	10	DLE Data Link Escape
CTRL Q CTRL q	11	DC1 Device Control 1
CTRL R CTRL r	12	DC2 Device Control 2
CTRL S CTRL s	13	DC3 Device Control 3
CTRL T CTRL t	14	DC4 Device Control 4
CTRL U CTRL u	15	NAK Negative Acknowledge
CTRL V CTRL v	16	SYN Synchronous Idle
CTRL W CTRL w	17	ETB End of Transmission Block
CTRL X CTRL x	18	CAN Cancel
CTRL Y CTRL y	19	EM End of Medium
CTRL Z CTRL z	1A	SUB Substitute
CTRL [ CTRL {	1B	ESC Escape
CTRL \ CTRL	1C	FS File Separator
CTRL ] CTRL }	1D	GS Group Separator
CTRL ^ CTRL ~	1E	RS Record Separator
CTRL _	1F	US Unit Separator

Abb.3 ASCII - Befehlscode

Es ist ein mühsames Unterfangen, nach den Codetabellen die notwendigen Zeichen in einem Programm an den Drucker zu übermitteln, deshalb lohnt sich der Aufwand, einige Unterprogramme zu erstellen, die einen einfacheren Zugriff ermöglichen.

Besonders elegant lässt sich diese Aufgabe in PASCAL lösen; in BASIC ist der Aufwand grösser und das Programm wesentlich unübersichtlicher.

Für praktische Anwendungen werden in Listing 1 und Listing 2 elementare und kompliziertere Unterprogramme in PASCAL (PASCAL/M von Sorcim) und BASIC (MBASIC von Microsoft) wiedergegeben. Zum besseren Verständnis sind diese Unterprogramme mit Kommentaren versehen und in ein Demo-Programm verpackt, des-

sen Resultat für beide Programmiersprachen gleich ist.

Eine Erklärung dürfte aber zusätzlich notwendig sein: Da der Drucker mit einer Genauigkeit von 1/120 Zoll (0.21 mm) horizontal bzw. 1/48 Zoll (0.53 mm) vertikal plottet, ist eine Rundung von Zahlenwerten auf entsprechende Integer-Einheiten notwendig. Für absolute Positionswerte ist das kein Problem, für relative Positionierungen (z.B. bei Vektoradditionen etc.) summieren sich jedoch Rundungsfehler zu untragbarer Ungenauigkeit auf.

Am einfachsten ist deshalb die genaue Berechnung der jeweiligen Punktkoordinaten und eine Rundung vor der Positionierung. Das wird vor allem dann notwendig sein, wenn Winkel oder metrische Längen auf-

treten. Benötigt man nur ganzzahlige äquidistante Teilungen, dann lässt sich als Einheit 1/24 Zoll wählen (im weiteren kurz als 1 me bezeichnet, 1 me = 1/24 Zoll = 1.05833..mm), was ungefähr einem Millimeter entspricht und das kleinste ganzzahlige Vielfache von 1/120 und 1/48 Zoll ist.

## ERKLÄRUNG DER UNTERPROGRAMME

Bei der Eingabe können im BASIC-Programm alle REM-Zeilen weggelassen werden. Im PASCAL-Programm wurden nicht die üblichen Zeichen (\* und \*) für Kommentare verwendet, sondern ein Doppelslash --, wie dies auch in ADA üblich ist.

## HCHRITT(N)

Aus Abb.4 ersehen Sie den Befehlscode, der die Programmierung der horizontalen Schrittweite von 0/120 bis 15/120 Zoll ermöglicht. So bewirkt zum Beispiel die ASCII-Folge ESC,J,L, dass die Schrittweite auf 12/120 = 1/10 Zoll gesetzt wird.

Im PASCAL-Programm sorgt der Prozeduraufruf HCHRITT(N) mit einem ganzzahligen Variablenwert N zwischen 0 und 15 für die Ausgabe der richtigen ASCII-Werte.

		Horizontal- abstand	Vertikal- abstand
1.	2.	3. / Zoll	3. / Zoll
ESC	J	@ 0	P 1/48
	A	1/120	Q 2/48
	B	2/120	R 3/48
	C	3/120	S 4/48
	D	4/120	T 5/48
	E	5/120	U 6/48 (1/8)
	F	6/120	V 7/48
	G	7/120	W 8/48 (1/6)
	H	8/120	X 9/48
	I	9/120	Y 10/48
	J	10/120 (1/12)	Z 11/48
	K	11/120	[ 12/48
	L	12/120 (1/10)	\ 13/48
	M	13/120	] 14/48
	N	14/120	^ 15/48
	O	15/120	- 16/48

Abb.4 Horizontal-/Vertikalabstand

Im BASIC-Programm ist zuerst der Variablenwert ZZ zwischen 0 und 15 zu setzen und dann das zugehörige Unterprogramm mit GOSUB 50010 aufzurufen.

## VSCHRITT(N)

Ebenfalls aus Abb.4 ist zu ersehen, dass 16 verschiedene Vertikalschrittweiten programmierbar sind, von 1/48 bis 16/48 Zoll. Der Prozeduraufruf VSCHRITT(N) bzw. der Unterprogrammaufruf GOSUB 50020 bewirkt den Transfer der drei notwendigen ASCII-Zeichen.

## HTAB(N)

Besonders umfangreich ist die in Abb.5 dargestellte Codetabelle für den absoluten Horizontaltabulator.

Die gesamte Wagenbreite des Druckers ist in 163 1/12-Zoll-Schritte unterteilt. Dementsprechend können 163 verschiedene Befehle, bestehend aus ESC, gefolgt von zwei weiteren ASCII-Zeichen, an den Drucker gesendet werden, um den Druckkopf unverzüglich an einer dieser 163 Stellen zu positionieren.

Für die Position 59 ist die Zeichenfolge ESC,Q,Z notwendig. Im Unterprogramm wird die Positionszahl in die entsprechende ASCII-Folge umgesetzt und an den Drucker ausgegeben.

## VTAB(N)

Abb.6 zeigt die Codefolgen, die notwendig sind, um den Papiertransport in verschiedenen Schrittzahlen (0 bis 63) zu programmieren.

Zusätzlich lässt sich die Richtung festlegen, in die das Papier transportiert werden soll. Im Unterprogramm VTAB wird durch einen positiven, ganzzahligen Variablenwert (1 bis 63) ein Papiervorschub, durch einen negativen, ganzzahligen Variablenwert (-1 bis -63) ein Papierrücktransport erzielt. Die Schrittweite entspricht der in VSCHRITT programmierten Einheit.

## EINSCHRITT(N)

bewirkt die Ausführung eines Einzelschrittes nach rechts (Variablenwert 0), nach "oben" (1), nach links (2) oder nach "unten" (3).

## PUNKT

druckt in der momentanen Position einen Punkt. Um das Weiterrücken des Druckkopfes zu kompensieren wird zusätzlich ein Backspace ausgegeben.

## DRUCKVOR

initialisiert die Druckrichtung links-rechts.

## DRUCKRUECK

initialisiert die Druckrichtung rechts-links.

## ROT

schaltet das Farbband auf rot.

## SCHWARZ

schaltet das Farbband auf Schwarz.

## VEKTOR (DX,DY)

führt eine Vektorbewegung um DX (der in HSCHRITT programmierten) Horizontaleinheiten (nach rechts oder links) und DY Vertikaleinheiten (nach oben oder unten) aus.

1st	ESC																					
2nd	P		Q		R		S		T		U											
3rd	@	1	P	17	@	33	P	49	@	65	P	81	@	97	P	113	@	129	P	145	@	161
A	2	Q	18	A	34	Q	50	A	66	Q	82	A	98	Q	114	A	130	Q	146	A	162	
B	3	R	19	B	35	R	51	B	67	R	83	B	99	R	115	B	131	R	147	B	163	
C	4	S	20	C	36	S	52	C	68	S	84	C	100	S	116	C	132	S	148			
D	5	T	21	D	37	T	53	D	69	T	85	D	101	T	117	D	113	T	149			
E	6	U	22	E	38	U	54	E	70	U	86	E	102	U	118	E	134	U	150			
F	7	V	23	F	39	V	55	F	71	V	87	F	103	V	119	F	135	V	151			
G	8	W	24	G	40	W	56	G	72	W	88	G	104	W	120	G	136	W	152			
H	9	X	25	H	41	X	57	H	73	X	89	H	105	X	121	H	137	X	153			
I	10	Y	26	I	42	Y	58	I	74	Y	90	I	106	Y	122	I	138	Y	154			
J	11	Z	27	J	43	Z	59	J	75	Z	91	J	107	Z	123	J	139	Z	155			
K	12	[	28	K	44	[	60	K	76	[	92	K	108	[	124	K	140	[	156			
L	13	\	29	L	45	\	61	L	77	\	93	L	109	\	125	L	141	\	157			
M	14	]	30	M	46	]	62	M	78	]	94	M	110	]	126	M	142	]	158			
N	15	^	31	N	47	^	63	N	79	^	95	N	111	^	127	N	143	^	159			
O	16	-	32	O	48	-	64	O	80	-	96	O	112	-	128	O	144	-	160			

Abb.5 Absoluter Horizontaltabulator

	Papierrücktransport				Papiervorschub			
1st	ESC				ESC			
2nd	X		Y		Z		[	
3rd	e	P	@	P	@	P	e	P
0	16	32	48	0	16	32	48	
A	Q	A	Q	A	Q	A	Q	
1	17	33	49	1	17	33	49	
B	R	B	R	B	R	B	R	
2	18	34	50	2	18	34	50	
C	S	C	S	C	S	C	S	
3	19	35	51	3	19	35	51	
D	T	D	T	D	T	D	T	
4	20	36	52	4	20	36	52	
E	U	E	U	E	U	E	U	
5	21	37	53	5	21	37	53	
F	V	F	V	F	V	F	V	
6	22	38	54	6	22	38	54	
G	W	G	W	G	W	G	W	
7	23	39	55	7	23	39	55	
H	X	H	X	H	X	H	X	
8	24	40	56	8	24	40	56	
I	Y	I	Y	I	Y	I	Y	
9	25	41	57	9	25	41	57	
J	Z	J	Z	J	Z	J	Z	
10	26	42	58	10	26	42	58	
K	[	K	[	K	[	K	[	
11	27	43	59	11	27	43	59	
L	\	L	\	L	\	L	\	
12	28	44	60	12	28	44	60	
M	]	M	]	M	]	M	]	
13	29	45	61	13	29	45	61	
N	^	N	^	N	^	N	^	
14	30	46	62	14	30	46	62	
O	-	O	-	O	-	O	-	
15	31	47	63	15	31	47	63	

Abb.6 Relativer Vertikaltabulator

## VEKTORPOL (LAENGE,WINKEL)

bewirkt eine Vektorbewegung, wobei LAENGE und WINKEL als vektorbestimmende Polarkoordinaten anzusehen sind. LAENGE ist in me, WINKEL in Grad anzugeben.

Beispiel: VEKTORPOL (42, 47.35)  
"schiebt" den Druckkopf 42me unter 47.35 Grad "höher".

## VEKTORKAR (XME,YME)

führt eine Vektorbewegung aus, wobei XME und YME als vektorbestimmende kartesische Koordinaten anzusehen sind. Ihre positiven oder negativen Werte sind in me anzugeben.

Beispiel: VEKTORKAR (68, -37)

## KREIS (RADIUS,PZAHL,MIPUNKT)

zeichnet einen Kreis aus PZAHL Punkten mit dem Radius RADIUS (in me!) um die momentane Druckkopfposition als Mittelpunkt. Ist MIPUNKT=1 gesetzt, so wird der Kreismittelpunkt markiert.

Die Liste dieser Unterprogramme können Sie, Ihren Bedürfnissen entsprechend, erweitern. Abschliessend noch einige Bemerkungen:

Im PASCAL-Programm (Listing 1) steht im Vereinbarungsteil

```
var F: text;
```

sowie nach Programmbeginn

```
rewrite (F, 'PRINTER:');
```

Dadurch wird die Ausgabe der ASCII-Werte über ein Printer-File vollzogen.

Im BASIC-Programm (Listing 2) sorgt LPRINT (LINEPRINT) für die Uebermittlung der ASCII-Zeichen an den Drucker. Da in MBASIC beim Ueberschreiten der vorprogrammierten Zeilenlänge automatisch ein CR ausgegeben wird, muss man dies unterdrücken, indem man mit WIDTH LPRINT 255 die Zeilenlänge auf "unendlich" stellt.

Vergleichen wir abschliessend die beiden Programme, die aufbau-mässig und funktionell identisch sind, so erkennt man die Ueberlegenheit des PASCAL-Programms bezüglich Struktur und Funktion. Während im BASIC-Programm die Variablen im Hauptteil belegt und namensgleich im Unterprogramm verarbeitet werden, der GOSUB-Aufruf äusserst verwirrend und nichtsagend ist, genügt in PASCAL der Prozeduraufruf mit signifikanten Namen. Variablenwerte werden als Parameter übergeben, die lokal definierten Variablennamen können konfliktfrei immer wieder verwendet werden und belegen nur temporär Speicherplatz.

Das PASCAL-Programm benötigt nach der Compilierung rund eine Minute zum Zeichnen der drei Demo-Kreise (p-Code-Interpreter). Das BASIC-Programm braucht genau dreimal so lang (BASIC-Interpreter).

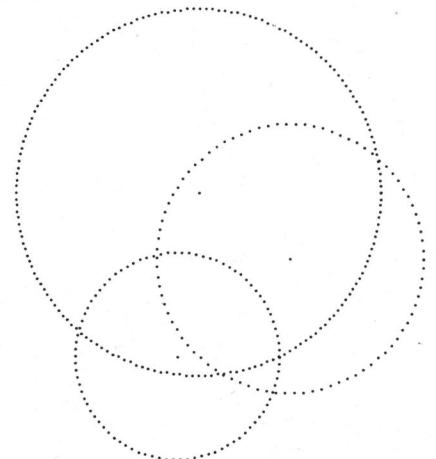


Abb.7 Demoprogramm PASCAL/BASIC

# Lehrgänge

```
-- NEC SPINWRITER 5510/5520 Grafik-/Plot-Prozeduren
-----
-- Leopold Asböck -190281-
-- Prozeduren in PASCAL/M zur Steuerung von
-- NEC-Spinwriter 5510/5520.
```

```
program SPINDEMO (input, output);
var F: text;          -- File-Variablen
----- Prozeduren! -----

procedure HSCHRITT (I: integer);
begin
  write (F, chr(27),chr(93),chr(64+I));
end;

procedure VSCHRITT (I: integer);
begin
  write (F, chr(27),chr(93),chr(79+I));
end;

procedure HTAB (I: integer);
begin
  var J,K: integer;
  J:= I div 32;
  K:= I mod 32;
  write (F, chr(27),chr(80+J),chr(65+K));
end;

procedure VTAB (I: integer);
begin
  var J,K: integer;
  J:= abs(I) div 32;
  K:= abs(I) mod 32;
  if I < 0 then
    write (F, chr(27),chr(88+J),chr(64+K))
  else
    write (F, chr(27),chr(90+J),chr(64+K));
end;

procedure EINSCHRITT (I: integer);
begin
  case I of
    1: write (F, chr(27),chr(88),chr(65));
    2: write (F, chr(8));
    3: write (F, chr(10));
  otherwise write (F, chr(32));
  end;
end;

procedure PUNKT;
begin
  write (F, '.',chr(8));
end;

procedure DRUCKVOR;
begin
  write (F, chr(27),chr(62));
end;

procedure DRUCKRUECK;
begin
  write (F, chr(27),chr(60));
end;

procedure ROT;
begin
  write (F, chr(27),chr(51));
end;

procedure SCHWARZ;
begin
  write (F, chr(27),chr(52));
end;

procedure VEKTOR (DX,DY: integer);
begin
  if DX <> 0 then
    for I:= 1 to abs(DX) do EINSCHRITT(1 - DX div abs(DX));
  if DY <> 0 then
    begin
      while abs(DY) > 63 do
        begin
```

```
          I:= -63*(DY div abs(DY));
          VTAB(I);
          DY:= DY+I;
        end;
      VTAB(-DY);
    end;
end;

procedure VEKTORPOL (LAENGE,WINKEL: real);
begin
  -- führt den Vektor (LAENGE,WINKEL) aus ... Polarkoordinaten!
  -- HSCHRITT und VSCHRITT werden auf kleinste Einheit gesetzt!
  -- LAENGE in me / WINKEL in Grad!

  var DX,DY: integer;
      X,Y: real;

  begin
    HSCHRITT(1); VSCHRITT(1);
    LAENGE:= LAENGE*120/24;
    WINKEL:= WINKEL*3.14159/180;
    X:= LAENGE*cos(WINKEL);
    Y:= LAENGE*sin(WINKEL);
    DX:= round(X);
    DY:= round(0.4*Y);
    VEKTOR (DX,DY);
  end;

procedure VEKTORKAR (XME,YME: real);
begin
  -- führt den Vektor (XME,YME) aus ... Kartesische Koordinaten!
  -- HSCHRITT und VSCHRITT werden auf kleinste Einheit gesetzt!
  -- XME und YME in me !

  var DX,DY: integer;

  begin
    XME:= XME*120/24;
    YME:= YME*0.4*120/24;
    DX:= round (XME);
    DY:= round (YME);
    HSCHRITT(1); VSCHRITT(1);
    VEKTOR (DX,DY);
  end;

procedure KREIS (RADIUS: real; PZAHL,MIPUNKT: integer);
begin
  -- zeichnet einen Kreis mit der momentanen Schreibkopfposition
  -- als Mittelpunkt, dem Radius RADIUS (in me !), aus PZAHL
  -- Punkten.
  -- HSCHRITT und VSCHRITT werden auf kleinste Einheit gesetzt!
  -- Für MIPUNKT=1 wird der Mittelpunkt markiert.

  var I,XU,YU,XNEU,YNEU,XALT,YALT,DX,DY: integer;
      PHI: real;

  begin
    HSCHRITT(1); VSCHRITT(1);
    XU:=0; YU:=0;
    XALT:=0; YALT:=0;
    RADIUS:= RADIUS*120/24;
    if MIPUNKT=1 then PUNKT;
    for I:= 0 to PZAHL do
      begin
        PHI:= I*2*3.14159/PZAHL;
        XNEU:= round (RADIUS*cos(PHI));
        YNEU:= round (RADIUS*0.4*sin(PHI));
        DX:= XNEU-XALT; YALT:= XNEU; XU:= XU+DX;
        DY:= YNEU-YALT; YALT:= YNEU; YU:= YU+DY;
        VEKTOR (DX,DY); PUNKT;
      end;
    VEKTOR (-XU,-YU);
  end;
end;

----- Ende der Prozeduren -----
----- Hauptprogramm -----

-- Dieses Demo-Programm plottet drei Kreise
-- mit 52me, 38me und 29me Radius aus 180
-- bzw. 90 Punkten.

begin
  rewrite (F, 'PRINTER:');

  HSCHRITT(12); VSCHRITT(8);
  HTAB(32); VTAB(23);
  HSCHRITT(1); VSCHRITT(1);
  KREIS (52, 180, 1);
  VEKTORKAR (26, -19);
  KREIS (38, 90, 1);
  VEKTORKAR (-32, -28);
  KREIS (29, 90, 1);
  writeln (F);
  HSCHRITT(12); VSCHRITT(8);
  VTAB(25);
end.
```

# Lehrgänge

```
1000 REM      SPINDEMO Programm für die Subroutinen für
1010 REM      NEC-Spinwriter 5510/5520 (ab Zeile 50000).
1020 REM      Dieses Programm plottet drei Kreise mit 52me,
1030 REM      38me und 29me Radius aus 180 bzw. 90 Punkten.
1040 REM
1045 WIDTH LPRINT 255: REM verhindert Ausgabe von Auto-CR
1050 ZZ=12: GOSUB 50010: REM HSCHRITT 12/120 Zoll
1060 ZZ= 8: GOSUB 50020: REM VSCHRITT 8/48 Zoll
1070 ZZ=32: GOSUB 50030: REM HTAB 32/12 Zoll
1080 ZZ=23: GOSUB 50040: REM VTAB 23/6 Zoll
1090 ZZ= 1: GOSUB 50010: GOSUB 50020
1100 RA= 52: PZ=180: MI=1: GOSUB 50180
1110 Z1= 26: Z2=-19: GOSUB 50160
1120 RA= 38: PZ= 90: MI=1: GOSUB 50180
1130 Z1=-32: Z2=-28: GOSUB 50160
1140 RA= 29: PZ= 90: MI=1: GOSUB 50180
1150 LPRINT
1160 ZZ=12: GOSUB 50010
1170 ZZ= 8: GOSUB 50020
1180 ZZ=25: GOSUB 50040
1190 END
1200 REM
50000 REM NEC SPINWRITER 5510/5520 Grafik-/Plot-Subroutinen
50001 REM
50002 REM
50003 REM      Leopold Asböck -190281-
50004 REM
50005 REM      Unterprogramme in Microsoft-BASIC zur Steuerung
50006 REM      von NEC-Spinwriter 5510/5520.
50007 REM
50008 REM -- HSCHRITT(ZZ) ... horizontale Schrittweite,
50009 REM      0 bis 15 (x 1/120 Zoll)
50010 LPRINT CHR$(27);CHR$(93);CHR$(64+ZZ):: RETURN
50017 REM
50018 REM -- VSCHRITT(ZZ) ... vertikale Schrittweite,
50019 REM      1 bis 16 (x 1/48 Zoll)
50020 LPRINT CHR$(27);CHR$(93);CHR$(79+ZZ):: RETURN
50027 REM
50028 REM -- HTAB(ZZ) ... absoluter Horizontaltabulator,
50029 REM      1 bis 163
50030 D1 = ZZ \ 32 : D2 = ZZ MOD 32 : REM backslash!
50032 LPRINT CHR$(27);CHR$(80+D1);CHR$(65+D2):: RETURN
50037 REM
50038 REM -- VTAB(ZZ) ... relativer Vertikaltabulator,
50039 REM      -63 bis 63
50040 D1 = ABS(ZZ) \ 32 : D2 = ABS(ZZ) MOD 32 : REM backslash!
50042 IF ZZ < 0 THEN LPRINT CHR$(27);CHR$(88+D1);CHR$(64+D2);
50044 IF ZZ >= 0 THEN LPRINT CHR$(27);CHR$(90+D1);CHR$(64+D2);
50046 RETURN
50047 REM
50048 REM -- EINSCHRITT(ZZ) ... 0=rechts/1=auf
50049 REM      2=links/3=ab
50050 IF ZZ = 0 THEN LPRINT CHR$(32);
50052 IF ZZ = 1 THEN LPRINT CHR$(27);CHR$(57);
50054 IF ZZ = 2 THEN LPRINT CHR$(27);CHR$(60);CHR$(32);
      CHR$(27);CHR$(62);
50056 IF ZZ = 3 THEN LPRINT CHR$(10);
50058 RETURN
50067 REM
50068 REM -- PUNKT ... druckt einen Punkt
50069 REM
50070 LPRINT ". ";CHR$(27);CHR$(60);CHR$(32);CHR$(27);
      CHR$(62);
50077 REM
50078 REM -- DRUCKVOR ... Druck von links nach rechts
50079 REM
50080 LPRINT CHR$(27);CHR$(62):: RETURN
50087 REM
50088 REM -- DRUCKRUECK ... Druck von rechts nach links
50089 REM
50090 LPRINT CHR$(27);CHR$(60):: RETURN
50097 REM
50098 REM -- ROT ... schaltet Farbband auf rot
50099 REM
50100 LPRINT CHR$(27);CHR$(51):: RETURN
50107 REM
50108 REM -- SCHWARZ ... schaltet Farbband auf schwarz
50109 REM
50110 LPRINT CHR$(27);CHR$(52):: RETURN
50117 REM
50118 REM -- VEKTOR (DX,DY)
50119 REM
50120 IF DX > 0 THEN ZZ=0: FOR DD= 1 TO DX :
      GOSUB 50050: NEXT DD
50122 IF DX < 0 THEN ZZ=2: FOR DD= 1 TO -DX :
      GOSUB 50050: NEXT DD
50124 IF DY > 0 THEN DD=-63: GOSUB 50130
50126 IF DY < 0 THEN DD= 63: GOSUB 50130
50128 RETURN
50130 IF ABS(DY) > 63 THEN ZZ=DD: GOSUB 50040:
      DY=DY+DD: GOTO 50130
50132 ZZ=-DY: GOSUB 50040: RETURN
50135 REM
50136 REM -- VEKTORPOL(LA,WI) führt Vektor (LA,WI) aus.
50137 REM      Länge LA in me !
50138 REM      Winkel WI in Grad !
50139 REM
50140 ZZ=1: GOSUB 50010: GOSUB 50020
50142 LA= LA*120/24: WI= WI*3.14159/180
50144 D1= LA*COS(WI): D2=LA*SIN(WI)
50146 DX= CINT(D1): DY= CINT(.4*D2)
50148 GOSUB 50120: RETURN
50155 REM
50156 REM -- VEKTORKAR(Z1,Z2) führt Vektor (Z1,Z2) aus.
50157 REM      x-Wert Z1 in me !
50158 REM      y-Wert Z2 in me !
50159 REM
50160 Z1= Z1*120/24: Z2= Z2*.4*120/24
50162 DX= CINT(Z1): DY= CINT(Z2)
50164 ZZ=1: GOSUB 50010: GOSUB 50020
50166 GOSUB 50120: RETURN
50175 REM
50176 REM -- KREIS(RA,PZ,MI) zeichnet Kreis mit Radius
50177 REM      RA aus PZ Punkten. Wenn
50178 REM      MI=1 mit Mittelpunkt.
50179 REM
50180 ZZ=1: GOSUB 50010: GOSUB 50020
50182 XU=0: YU=0: XA=0: YA=0
50184 RA= RA*120/24
50186 IF MI=1 THEN GOSUB 50070
50188 FOR II= 0 TO PZ
50190 WW= II*2*3.14159/PZ
50192 XN= CINT(RA*COS(WW))
50194 YN= CINT(RA*SIN(WW))
50196 DX= XN-XA: XA= XN: XU= XU+DX
50198 DY= YN-YA: YA= YN: YU= YU+DY
50200 GOSUB 50120: GOSUB 50070
50202 NEXT II
50204 DX= -XU: DY= -YU: GOSUB 50120: RETURN
```

Listing 2 Plot-Subroutine (BASIC)

# PPC - Die Programmierbaren



## Textverarbeitung mit PPC?

Gerfried TATZL, dipl. Ing. WIV

Man wird auf die Frage, ob eine Textverarbeitung mit programmierbaren Taschenrechnern überhaupt möglich ist, nur müde lächelnd darauf hinweisen, dass ein Schuster bei seinem Leisten bleiben solle und dass ein Taschenrechner eben ein Taschenrechner sei, Textverarbeitung mit Textverarbeitung zu tun habe und dass zwischen diesen beiden Dingen so wenig Verwandtschaft vorliege, als zwischen Aepfel und Birnen.

Wenn wir also doch von Textverarbeitung in Verbindung mit programmierbaren Taschenrechnern sprechen wollen, dann müssen wir erst einmal abgrenzen, was wir in diesem Fall unter Textverarbeitung verstehen.

Mit Sicherheit werden wir keine Briefe schreiben können. Nicht nur aufgrund der mangelnden Speicherfähigkeit, sondern auch wegen der Unmöglichkeit, einen Taschenrechner an einen Drucker anschliessen zu können, der mit Papier im Format DIN A4 beschickt wird. Wenn wir also keine Briefe schreiben können, so liegt doch der Gedanke nahe, dem Rechner wenigstens das Schreiben von Anschriften, also Adresstiketten, zuzumuten.

Voraussetzen müssen wir aber einen Rechner mit der Fähigkeit, alphanumerische Zeichenketten verarbeiten zu können. Es versteht sich von selbst, dass diese Modelle mit einer Druckeinrichtung ausgestattet sein müssen oder an eine solche anschliessbar sind. Dafür sind z.B. die Rechnermodelle HP 41C und TI-59 mit dem Drucker PC-100C geeignet.

Vom System her ergibt sich dabei eine Schwierigkeit: Aufgrund des Thermopapiers kann der Druckstreifen nicht mit einem lösungsmittelhaltigen Leim auf einen Umschlag geklebt werden. Man wird also versuchen, von den ausgedruckten Anschriften Trockenkopien anzuferti-

gen und diese Kopien dann auf die Umschläge zu kleben.

Besser wäre, nach einer Möglichkeit zu suchen, diese Trockenkopien gleich als Etiketten herzustellen. Tatsächlich gibt es heute Kopieretiketten, die unseren Zwecken nützlich sind. Für das in nachstehendem Programm verwendete Druckbild wurden Etiketten der Grösse von etwa 70/36 mm (3 mal 8 Stück/Blatt) verwendet; u.a. wird ein solches Papier von Rank Xerox geliefert. Nun müssen wir den Druckstreifen nur noch auf einer Unterlage mit Klarsichtklebestreifen befestigen und können dann die Adresstiketten in beliebiger Anzahl kopieren.

Für den vorliegenden Zweck wurde für den Rechner TI 59 samt Drucker PC-100C ein solches Programm verfasst. Das vorliegende Programm soll als Anregung dienen, sich mit einem ähnlichen Problem auseinanderzusetzen. Sie können das Programm aber auch in nur einigen wenigen Punkten abändern und damit Ihren Bedürfnissen anpassen.

- Für die ANREDE wurde hier der Text "FRAU" gewählt, da in unserem Beispiel nur die Anschriften von Frauen erstellt werden sollen. Dieser Text wird vom Programm aus erzeugt und gilt nicht als personenspezifisch. Das gilt für das Unterstreichen der Ortsbezeichnung sinngemäss.

- Der TITEL einer Person wird mindestens einen 5-stelligen alphanumerischen Druckspeicher beanspruchen. Man könnte für längere Titel auch zwei Speicher wählen; es sollte aber mit solchen Textteilen etwas hausgehalten werden, um so viele Anschriften in einem verarbeiten zu können, als dies möglich ist.
- VORNAME und NAME bringen wir in eine Druckzeile mit vier Druckspeichern. Allenfalls wäre es möglich, hier auch Reste eines Titels unterzubringen.
- Ebenfalls eine ganze Zeile reservieren wir für Postleitzahl und ORT.
- Haben wir Anschriften sowohl im In- als auch im Ausland zu bedienen, sollten wir zumindest einen Druckspeicher für die Bezeichnung LAND freihalten, der für Inlandsanschriften allerdings leer bleiben kann.

Im Programm selbst werden wir folgende voneinander unabhängige Programmzweige benötigen:

- Abruf der Adressliste, worauf alle im Augenblick gespeicherten Anschriften ausgedruckt werden; neue Anschriften werden nach Einlesen weiterer Datenkarten ausgedruckt.
- Wenn möglich, sollte auf dem gleichen Programmstreifen auch das Laden der Informationen für den Adressenausdruck untergebracht werden.

# PPC - Die Programmierbaren

- Ebenso sollte mit dem gleichen Programm das Aendern einzelner Detailinformationen möglich sein. Dazu ist es erforderlich, einzelne Datenspeicher direkt ansteuern zu können.
- Korrekturen von Fehleingaben sollen als eigenständiger Programmzweig in das Gesamtprogramm eingebaut werden.
- Für den Fall nachträglicher Aenderungen empfiehlt sich die eigenständige Anwahl der Datenübernahme. Es muss ja nicht immer die letztmögliche Information sein, die geändert wird und nach der ohnehin die Datenübernahme automatisch erfolgen sollte. Eine solche Massnahme ist nämlich ein ausgezeichnetes bedienereführendes Element in der Verarbeitung.

Bei Betrachtung der Speicherkapazität des TI 59 werden wir die Speicherverteilung erst einmal so zu wählen versuchen, dass das Programm in Block 1 und die Anschriften in Form der Zifferncodes in den Blöcken 2, 3 und 4 gespeichert werden. Das ergibt somit eine Kapazität von 240 Programmschritten und von 90 Datenspeichern.

Für eine 20 Druckpositionen umfassende Zeile werden vier Druckspeicher (OP 01 bis OP 04) benötigt. Da wir zur Steuerung der Verarbeitung den einen oder anderen Hilfsspeicher brauchen werden, müssen wir deren Anzahl von 90 abziehen, um so zur Anzahl der für die Speicherung von Anschriften zur Verfügung stehenden Datenspeicher zu gelangen. Für eine Anschrift benötigen wir im Mindestfall drei Druckzeilen für Name, Strasse und Ort; das allein macht schon 12 Datenspeicher aus. Wenn wir noch je einen Datenspeicher für den Titel und für die Landesbezeichnung bereitstellen, sollten insgesamt 14 Datenspeichern reichen.

Wohl ergibt die Operation 90 geteilt durch 15 einen ganzzahligen Quotienten; 15 Speicher für eine Anschrift lassen aber keine Hilfsspeicher zu. Bei 12 Datenspeichern

je Anschrift könnten neben dem Programm 7 Anschriften untergebracht werden; ab 13 Speicher sind es nur mehr 6 Anschriften. Wir bleiben also bei 6 Anschriften und wählen dazu die bereits erwähnten 14 Datenspeicher je Anschrift. Für unsere Speichernutzung treffen wir also folgende Festlegungen:

- Den Speicher 00 reservieren wir als Indexspeicher zur Steuerung der Verarbeitung, sowohl für den Fall der Einspeicherung der Adressinformationen als auch der Listung der Anschriften.
- Zur Aufnahme einer Anschrift wählen wir im einzelnen folgende Verteilung:

## TITEL

- 1 Speicher (5 Stellen); VORNAME, NAME
- 4 Speicher (20 Stellen); STRASSE, HAUSNUMMER
- 4 Speicher (20 Stellen); PLZ, ORT
- 4 Speicher (20 Stellen); LAND
- 1 Speicher (5 Stellen).

Pro Anschrift werden im einzelnen folgende Datenspeicher in Anspruch genommen:

1. Anschrift - Speicher 01 bis 14;
2. Anschrift - Speicher 15 bis 28;
3. Anschrift - Speicher 29 bis 42;
4. Anschrift - Speicher 43 bis 56;
5. Anschrift - Speicher 57 bis 70;
6. Anschrift - Speicher 71 bis 84.

- Die Speicher 85 bis 89 stehen im Bedarfsfall als Hilfsspeicher anderen Zwecken zur Verfügung.

Diese hier getroffenen Festlegungen erfordern die Speicherverteilung 239.89, die mit der Tastenfolge

9 (2nd) (OP) 17

eingestellt werden kann.

Als praktisches Beispiel wählen wir folgende willkürlich angenommene Musteranschrift:

FRAU DR.  
DORIS MUELLER-GRAF  
BAHNHOFSTRASSE 21  
6002 LUZERN  
CH

Die für die Erzeugung der Textteile nötigen Zahlen sehen im TI-Code wie folgt aus:

Textteil	Zifferncode	Speicherposition
FRAU	2135134100	Programmspeicher
DR.	1635400000	n + 01
DORIS	1632352436	n + 02
MUEL	30411727	n + 03
LER-G	2717352022	n + 04
RAF	3513210000	n + 05
BAHNH	1413233123	n + 06
OFSTR	3221363735	n + 07
ASSE	1336361700	n + 08
21	302000000	n + 09
6002	701010300	n + 10
L U	27004100	n + 11
Z E R	4600170035	n + 12
N	31000000	n + 13
C H	1500230000	n + 14

Der Doppelstrich zur Markierung von PLZ und ORT befindet sich ebenfalls wie der Text "FRAU" im Programm. Die allgemeine in der Spalte "Speicherposition" angegebene Zahl n bedeutet die Position einer Anschrift, zu der die jeweils angegebene Zahl zu addieren ist, um auf die Speicheradresse zu kommen. Im Fall unserer Anschrift ist n natürlich 0. Damit ist bestimmt, dass der erste Speicher der hier angeführten Anschrift jener mit der Adresse 01 ist; der Speicher mit der letzten Information ist der mit der Adresse 14.

## EINGABEN

Die Eingaben der Zifferncodes erfolgen nach Anwahl der Taste E mit Hilfe der R/S-Taste. Nach Abschluss

# PPC - Die Programmierbaren

einer Eingabe wird die Speichernummer angezeigt, für die die nächste Eingabe vorzunehmen ist.

Fehleingaben werden mit Hilfe der Tasten 2nd E rückgängig gemacht, indem die Positionsnummer der ungültigen Eingabe einfach wiederholt wird. Durch Neueingabe wird die Fehleingabe überschrieben. Änderungen werden nach Anwahl der jeweiligen Eingabeposition über die Taste E mit Hilfe der R/S-Taste vorgenommen.

## AUSGABEN

Nach Betätigung der Taste A werden sämtliche augenblicklich gespeicherten Anschriften ausgedruckt. Ist die Speicherkapazität

des Rechners nicht ausgenutzt, wird die Ausgabe der Anschriften mit der als letzte gespeicherten beendet.

```

FRAU DR.
DORIS MUELLER-GRAF

RAHNHUFSTRASSE 21
6002 L U Z E R N
=====
C H
    
```

Je nach vorhandenen Etiketten muss das Druckbild einer Anschrift entsprechend gestaltet werden. Dabei ist eine Eigenheit des gewählten Rechners zu berücksichtigen: Ein mit der ADV-Taste ausgelöster Papiervorschub wird mit einem anderen Zeilenabstand vorgenommen, als ein solcher im Falle ausgeführter Druckanweisungen gegeben ist.

Will man nun eine Leerzeile im gleichen Abstand mit den Druckzeilen einplanen, ist folgenden Anweisungen nötig:

```

CLR
OP
00
OP
05
    
```

Dieser Umstand kann ausgezeichnete Dienste leisten, wenn es gilt, unterschiedlichen Abmessungen von Etiketten gerecht zu werden. Vor endgültiger Festlegung sollte das Druckbild eingehend studiert werden. Unser Druckbild wurde der Etikettenbreite von ca. 36 mm annähernd angepasst und ist für acht in einer Spalte unterzubringende Anschriften ausreichend gewählt.

000	76	LBL	040	76	LBL	080	02	2	120	69	DP	160	01	1	200	03	03
001	69	DP	041	42	STD	081	03	3	121	02	02	161	03	3	201	69	DP
002	69	DP	042	91	R/S	082	03	3	122	69	DP	162	04	4	202	04	04
003	20	20	043	72	ST*	083	00	0	123	05	05	163	01	1	203	69	DP
004	73	RC+	044	00	00	084	01	1	124	04	4	164	00	0	204	05	05
005	00	00	045	69	DP	085	07	7	125	96	WRT	165	00	0	205	25	CLR
006	92	RTN	046	20	20	086	69	DP	126	91	R/S	166	69	DP	206	69	DP
007	76	LBL	047	43	RCL	087	03	03	127	76	LBL	167	01	01	207	00	00
008	52	EE	048	00	00	088	69	DP	128	12	B	168	71	SBR	208	29	CP
009	71	SBR	049	22	INV	089	05	05	129	42	STD	169	69	DP	209	71	SBR
010	69	DP	050	77	GE	090	25	CLR	130	00	00	170	69	DP	210	69	DP
011	69	DP	051	42	STD	091	69	DP	131	32	XIT	171	02	02	211	69	DP
012	01	01	052	01	1	092	00	00	132	08	8	172	69	DP	212	01	01
013	71	SBR	053	06	6	093	01	1	133	05	5	173	05	05	213	69	DP
014	69	DP	054	01	1	094	04	4	134	32	XIT	174	71	SBR	214	05	05
015	69	DP	055	03	3	095	02	2	135	61	GTD	175	52	EE	215	02	2
016	02	02	056	03	3	096	07	7	136	42	STD	176	25	CLR	216	44	SUM
017	71	SBR	057	07	7	097	03	3	137	76	LBL	177	69	DP	217	00	00
018	69	DP	058	01	1	098	02	2	138	10	E'	178	00	00	218	73	RC+
019	69	DP	059	07	7	099	01	1	139	69	DP	179	69	DP	219	00	00
020	03	03	060	03	3	100	05	5	140	30	30	180	05	05	220	67	EQ
021	71	SBR	061	01	1	101	02	2	141	43	RCL	181	71	SBR	221	91	R/S
022	69	DP	062	69	DP	102	06	6	142	00	00	182	52	EE	222	02	2
023	69	DP	063	01	01	103	69	DP	143	61	GTD	183	71	SBR	223	22	INV
024	04	04	064	04	4	104	01	01	144	42	STD	184	52	EE	224	44	SUM
025	69	DP	065	01	1	105	03	3	145	76	LBL	185	06	6	225	00	00
026	05	05	066	01	1	106	69	DP	146	11	A	186	04	4	226	43	RCL
027	92	RTN	067	07	7	107	02	02	147	00	0	187	06	6	227	00	00
028	76	LBL	068	01	1	108	69	DP	148	42	STD	188	04	4	228	32	XIT
029	15	E	069	04	4	109	05	05	149	00	00	189	06	6	229	07	7
030	47	CMS	070	01	1	110	02	2	150	76	LBL	190	04	4	230	00	0
031	08	8	071	07	7	111	96	WRT	151	23	LNK	191	06	6	231	77	GE
032	05	5	072	03	3	112	04	4	152	98	ADV	192	04	4	232	23	LNK
033	32	XIT	073	05	5	113	69	DP	153	25	CLR	193	06	6	233	76	LBL
034	25	CLR	074	69	DP	114	02	02	154	69	DP	194	04	4	234	91	R/S
035	69	DP	075	02	02	115	69	DP	155	00	00	195	69	DP	235	91	R/S
036	00	00	076	03	3	116	05	05	156	02	2	196	01	01	236	00	0
037	01	1	077	01	1	117	03	3	157	01	1	197	69	DP	237	00	0
038	42	STD	078	01	1	118	96	WRT	158	03	3	198	02	02	238	00	0
039	00	00	079	03	3	119	05	5	159	05	5	199	69	DP	239	00	0

# PPC - Die Programmierbaren

16.	00	001	69	OP
1635400000.	01	008	52	EE
1632352436.	02	029	15	E
30411727.	03	041	42	STD
2717052022.	04	128	12	B
3513210000.	05	138	10	E'
1413233123.	06	146	11	A
3221363735.	07	151	23	LNK
1336361700.	08	234	91	R/S
302000000.	09			
701010300.	10			
27004100.	11			
4600170035.	12			
31000000.	13			
1500230000.	14			
0.	15			
0.	16			
0.	17			
0.	18			
0.	19			
0.	20			
0.	21			
0.	22			
0.	23			
0.	24			
0.	25			
0.	26			
0.	27			
0.	28			

Das vorliegend Programm kann auch etwas anders gestaltet werden, indem anstelle der sechs nur vier Anschriften gespeichert werden, wozu die vom Rechner aus automatisch gewählte Speicherverteilung mit 479.59 genügt. Die zusätzliche Programmspeicherkapazität wird zur Aufnahme weiterer Funktionen verwendet, zumal auch noch eine Reihe freier Funktionstasten vorhanden sind.

Ein zusätzlicher Programmzweig wäre beispielsweise für das Aus-

drucken von nur einer Anschrift einzurichten. Eine weitere Zusatzeinrichtung könnte dazu dienen, diese eine Anschrift mehr als einmal hintereinander auszugeben, wobei in diesem Fall die Funktion mit der Anzahl der gewünschten Ausdrücke ausgelöst wird.

Nun soll jemand kommen und sagen, Textverarbeitung wäre nichts für programmierbare Taschenrechner. Auch wenn es nur ein begrenztes Problem war, das wir hier gelöst haben, so war dennoch erkennbar, wozu solch kleine Rechner fähig sind. Unsere Aufgabe, die wir uns in diesem Beitrag gestellt haben, war aufzuzeigen, dass mit einem programmierbaren Taschenrechner nicht nur gerechnet, sondern auch geschrieben werden kann.

Wer findet ähnliche Beispiele?

## Regent 20/25

Die raffiniert Einfachen



Bildschirmterminals von ADDS (Applied Digital Data Systems Inc.) für Übertragung im Zeichenmodus. Verfüg im Basismodell über:

- V24/RS232-Schnittstelle (20 mA Current Loop Option)
- Druckeranschluss mit Transparent-Schaltung
- Gross-/Kleinschrift
- 7 umschaltbare Charaktersätze
- Monitor-Mode mit Anzeige der ASCII-Kontrollcodes
- adressierbarer Cursor

**KONTRON AG**  
**DATASYSTEMS**

8048 Zürich, Bernerstrasse-Süd 169  
Telefon 01-62 82 82, Telex 57439  
1066 Epalinges, 10, ch. des Croisettes  
Téléphone 021-33 15 35, Télex 26 398

**(Für Lösungen mit Flair)**

## Mini-Disketten 5 1/4"

- 40 + 77 Spur, einzeln getestet
- speziell abriebfest = lange Lebensdauer
- mit Verstärkungsring

Günstige Preise!



## 8"-Disketten

- 100% fehlerfrei

Plastikboxen und weitere Ablegesysteme für Disketten

**KONTVA AG**

Gotthardstrasse 40, 8800 Thalwil, Telefon 01 / 720 10 26

# PPC - Die Programmierbaren

## Neue Befehle für HP 41

Jörg WARMUTH

Der Rechner HP 41C stellt den Grundbaustein zu einem überaus leistungsfähigen Taschenrechner-System dar. Seine alphanumerische LCD-Anzeige erlaubt in Verbindung mit der auf 2240 Bytes ausbaubaren Speicherkapazität, für einen Taschenrechner ungewöhnlich komfortable Programme mit Dialogmöglichkeit. Einige nicht aus dem Handbuch ersichtliche Betriebssystemmängel und die damit verbundenen Manipulationsmöglichkeiten zeigen wir nachfolgend auf.

Die ersten ausgelieferten Geräte wiesen Fehler auf, die dann nach und nach von HP "ausgebügelt" wurden. So gibt es Rechner, bei denen indirekt alle Flags - also auch die Systemflags 30-35 - gesetzt, gelöscht und abgefragt werden können.

Die späteren Modelle hatten diesen Fehler nicht mehr, aber ein weiterer Fehler war hier auch noch vorhanden. Diese Geräte sind leicht zu identifizieren, denn die statistische Funktion Summe + ( $\Sigma$  +) schreibt den letzten eingegebenen Wert nicht in das Last-X-Register. Rechner, die diese Eigenheit aufweisen, lassen den indirekten Aufruf der internen Statusregister zu. Bei neueren Geräten sind diese Möglichkeiten nicht mehr vorhanden. Die im folgenden beschriebenen Programme lassen sich somit nur auf einem Rechner ausführen, der diese Fehler besitzt.

Die dabei entstehenden Ergebnisse in Form von neuen Befehlen lassen sich jedoch auf Magnetkarten speichern, und bei neueren Geräten einlesen und anwenden.

Die möglichen indirekt aufrufbaren Register haben bei der werksmässig vorgegebenen Einstellung von 17 Daten- und 46-Programmregistern die Adressen 785-800. Es ist möglich, z.B. die Zahl 785 im Register 00 zu speichern und im Anschluss daran das erste Statusregister mit der Tastenfolge RCL IND 00 aufzurufen. Mit Stack und Alpharegister sind insgesamt 16 Register enthalten.

Der Wert dieser Adressen verschiebt sich bei Aenderung der SIZE-Einstellung. Die Adresse des 1. Registers kann mit folgender Formel ermittelt werden:

$$768 + \text{SIZE} - 64m = T$$

SIZE stellt die Anzahl der augenblicklich vorhandenen Datenregister - gemäss Einstellung - dar, m steht für die Anzahl der eingesetzten zusätzlichen Memory-Module und T ist die gesuchte Adresse. Die ersten 5 Register des Statusblockes stellen den Stack dar (Register T, Z, Y, X und Last-X), so dass die ermittelte Adresse indirekt das T-Register im Stack anspricht.

Zum Beispiel soll Register L indirekt aufgerufen werden. Der Rechner ist mit 3 Memory-Moduln bestückt, der Datenspeicherbereich ist auf 100 Register eingestellt. Die Adresse für L lautet: Adresse von T + 4. Gemäss der Formel wird nun der Standort der Statusregister ermittelt:

Formel:

$$768 + \text{SIZE} - 64m = T$$

eingesetzt:

$$768 + 100 - 64 * 3 = 676$$

$$676 = \text{Adresse von T,}$$

$$\text{somit } 676 + 4 = \text{Adresse von L.}$$

Die Tastenfolge von 680; RCL IND X ruft nunmehr indirekt das Last-X-Register auf. Die 16 Statusregister tragen folgende Bezeichnungen:

T, Z, Y, X, L, M, N, O, P, Q, R, a, b, c, d, e.

Eine Erhöhung der Adresse um jeweils 1, ausgehend von T, lässt den Aufruf des nächsten in der oben genannten Reihenfolge-Registers zu. Die Register T bis L stellen den Stack dar, M bis P das Alpharegister, b und e werden für die Tastenzuordnungen von HP 41C Funktionen benötigt, d enthält alle Flags. Die restlichen Register enthalten unter anderen die SIZE-Einstellung und die Adresse des Statistik-Registerblockes.

Diese Register lassen sich alle auch direkt adressieren. STO T oder STO L zum Beispiel kann vom Tastenfeld aus erzeugt werden. Der Bereich T bis L gehört ja zu den Registern, die notwendigerweise zum Rechnen benötigt werden. Der Befehl STO M oder STO d ist aber so nicht möglich.

Diese Befehle müssen auf eine besondere Weise erzeugt werden. Das ist möglich durch einen weiteren Fehler im Rechner: Der Programmspeicherbereich kann ebenfalls in-

### Digital Matrix-Printer DEC LA 34



**Neu: Fr. 1990.-**

DEC-Terminals ab Lager lieferbar von unseren Computer-Shops.

**pfeiffer**<sup>®</sup>

J.F. Pfeiffer AG, Löwenstrasse 61, 8001 Zürich  
Telefon 01/45 93 33  
weitere Filialen in Bern, Chur und Basel

# PPC - Die Programmierbaren

nerhalb eines bestimmten Bereiches indirekt adressiert werden. Es können Daten in Form von Befehls-codes in diesem Bereich gespeichert werden. Die Möglichkeit, Daten in den HP-Befehlscode zu "übersetzen", bildet die Grundlage für die Erzeugung der neuen Befehle.

Für den herzustellenden Befehl muss der zugehörige Binär-code erzeugt und in den Programmspeicher gebracht werden. Dieser Binär-code lässt sich einfach durch Löschen bzw. Setzen von Flags im Register d erzeugen. Im Register d wird durch die Flags ein Bitmuster gespeichert. Der Inhalt des d-Registers wird nach Herstellung des Bitmusters durch das Setzen bzw. Löschen von bestimmten Flags in den Programmbereich gespeichert. Die Erzeugung dieses Bitmusters wird durch Programm gemäss dem vorher eingegebenen Hex-Code aus der Tabelle der HP 41C Befehle ausgeführt.

Um diese programmgesteuerte Erzeugung von neuen Befehlen vornehmen zu können, ist der Abruf des Flagregisters (d) nötig. Der indirekte Aufruf verändert aber den Inhalt des Registers, so dass die Binär-code nicht mehr dem gewünschten zu erzeugenden Befehl entspricht. Es ist notwendig, auf eine andere Weise die Befehle STO d und RCL d zu erzeugen.

Das dazu notwendige Vorgehen wird im folgenden stichwortartig beschrieben. Wie schon erwähnt, ist dies nur bei den Rechnern aus den ersten Serien möglich.

Der Rechner wird vollständig gelöscht durch Einschalten bei gedrückter CLX-Taste. Nach Loslassen dieser Taste erscheint im Display "MEMORY LOST". Folgende Programmschritte werden jetzt eingegeben:

```
01 LBL X
02 +
. +
```

```
. +
. +
169 +
170 AVIEW
. "
. "
. "
176 "
177 +
178 +
. +
. +
183 +
184 AVIEW
. "
. "
190 "
- dann GTO...
```

Nun wird der Programm-Modus ausgeschaltet, es wird 997 ; ENTER ; EEX 90 ; STO IND . Y ; 999 ; ENTER ; EEX 91 ; STO IND . Y ausgeführt. Jetzt wird der Rechner erneut in den PRGM-Modus geschaltet und folgende Befehlsfolge ausgeführt: GOT . Alpha X Alpha ; BST ; BST die Programmzeilen 178 bis einschliesslich 175 werden gelöscht; BST die Zeilen 173 bis einschliesslich 167 werden ebenfalls gelöscht, dann GTO . 001 ; XEQ DEL 165;

Damit stehen jetzt die Befehle STO d und RCL d unmittelbar hintereinander. Das nun folgende Programm wird um diese beiden Befehle "herumgebaut":

```
01 LBL NB
02 I E 6
03 /
04 INT
05 LAST X
06 FRC
07 3
08 X - Z
09 LBL 00
10 16
11 *
12 X - Y
13 I E 2
14 *
15 FRC
16 LAST X
17 INT
18 ST+ Z
```

```
19 RDN
20 X - Y
21 DSE Z
22 GTO 00
23 0
```

jetzt SST - somit

```
24 STO d
25 RDN
26 LBL 01
27 STO Y
28 LOG
29 2
30 LOG
31 /
32 INT
33 15
34 -
35 SF IND X
36 LAST X
37 +
38 2
39 X - Y
40 Yx
41 -
42 X ≠ 0 ?
43 GTO 01
44 999
```

jetzt SST - somit

```
45 RCL d
46 STO IND Y
47 GTO 02
48 +
49 +
50 +
. +
. +
142 +
143 LBL 02
144 STOP
145 +
. +
. +
151 +
GTO ..
```

Es ist nun möglich, programmgesteuert neue, vom Tastenfeld nicht erzeugbare Befehle zu produzieren.

Dazu ist der erforderliche Hexadezimale-Code aus der Tabelle zu entnehmen. Der Code für den Befehl RCL lautet beispielsweise 90. Das Postfix, z.B. a, hat den Code 7B,

# PPC - Die Programmierbaren

zusammengesetzt ergibt sich der Hex-Code 907B. Im vorliegenden Programm wurde auf die Umwandlung der Hex-Zeichen verzichtet, der Code muss daher für jedes Hex-Zeichen als zweistellige Dezimalzahl eingegeben werden:

```
9 0 7 B
09 00 07 11
```

Es wird also 09000711 eingegeben und das Programm mit XEQ NB gestartet. Nach ca. 13 Sec hält das Programm an; Umschalten in den PRGM-Modus zeigt den erzeugten Befehl: RCL a.

Um weitere Befehle erzeugen zu können, ohne den zuletzt erzeugten zu vernichten, wird dieser nach "unten" verschoben, d.h. es werden insgesamt 7 Ein-Byte Befehle (z.B. +) nach dem Befehl STOP eingesetzt. Somit steht der Befehl RCL a in Zeile 152, die Zeilen 145-151 enthalten +; nur diese Bereich wird bei der Erzeugung der synthetischen Befehle benötigt.

Alle erzeugten Befehle lassen sich auf Magnetkarte speichern, auf neueren Rechnern (die nicht den beschriebenen Fehler besitzen) einlesen und auch verwenden.

Die Erzeugung der Befehle RCL bzw. STO M - e ermöglichen somit einen direkten Zugriff zu den Statusregistern.

Als zweites Beispiel die Erzeugung des Befehles TONE e:

```
TONE gemäss Tabelle 9F
Hex-Code 9 F 7 F
e gemäss Tabelle 7F
Hex-Code 09 15 07 15
```

Einzugeben ist 09150715; Start mit XEQ NB. Nach Anhalten des Programmes umschalten in den PRGM-Modus - das Display zeigt TONE e.

Da das Flag-Register benutzt wird, um diese Befehle zu erzeugen, ist Flag 26 (für das akustische Signal) z.Zt. gelöscht; zurück-

schalten in den Normal-Modus (PRGM aus); SF 26; SST -es ertönt TONE e.

Da es insgesamt 128 mögliche Postfix gibt, lassen sich neben den schon bekannten Tönen (TONE 0 - 9) noch weitere 118 erzeugen. Der Befehlscode im Display zeigt zwar nur ein einstelliges numerisches Postfix (z.B. bei 9F59 - 09150509 - TONE 9) im Bereich von 00-99, aber es sind unterschiedliche Frequenzen und Impulslängen zu hören.

Die möglichen direkten Speicheradressen für den Datenspeicherbereich 00-99 (RCL-STO- VIEW- usw. 00-99) werden nach "oben" hindurch 00; 01; A bis J ergänzt, diese stellen den Bereich 100-111 dar, der damit auch direkt adressierbar ist. 110; RCL IND X entspricht: RCL I.

Der Bereich der direkt aufrufbaren Register erhöht sich damit auf insgesamt 112 Register.

Die Aufstellung zeigt als Auszug aus der gesamten Befehlstabelle, nur die Befehle, die einen Postfix benötigen. Tabelle 1 zeigt den zu den Befehlen gehörigen HEX-CODE, Tabelle 2 gibt den HEXCODE für den notwendigen Postfix an.

Damit steht einer Produktion von nahezu unendlichen Kombinationsmöglichkeiten nichts mehr im Wege. Nicht alle Befehle aber haben die erwartete Wirkung: FIX e schaltet die Flags 42 und 43 wechselweise,  $\Sigma$ REG T setzt den Anfang des Statistikregisterblockes in die Stack-Register und X - c löscht?

Vorsicht! - hoffentlich war das Programm schon auf einer Magnetkarte aufgezeichnet!

TABELLE 1

Befehl	HEX
RCL	90
STO	91
STO+	92
STO-	93
STO*	94

STO/	95
ISG	96
DSE	97
VIEW	98
$\Sigma$ REG	99
ASTO	9A
ARCL	9B
FIX	9C
SCI	9D
ENG	9E
TONE	9F
X -	CE

TABELLE 2

Postfix	HEX	Postfix	HEX
00	00	T	70
.	.	Z	71
.	.	Y	72
99	63	X	73
00	64	L	74
01	65	M	75
A	66	N	76
B	67	O	77
C	68	P	78
D	69	Q	79
E	6A	F	7A
F	6B	a	7B
G	6C	b	7C
H	6D	c	7D
I	6E	d	7E
J	6F	e	7F

## Digital Matrix-Printer DEC LA 34



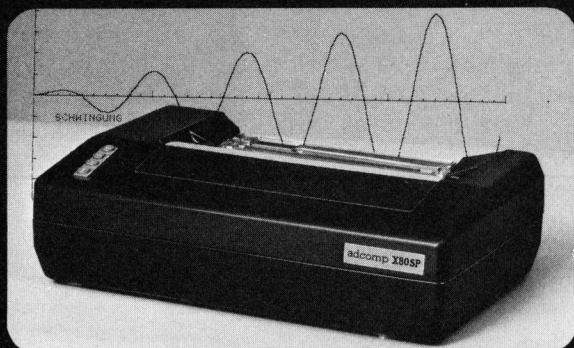
**Neu: Fr. 1990.-**

DEC-Terminals ab Lager lieferbar von unseren Computer-Shops.

**pfeiffer®**

J.F. Pfeiffer AG, Löwenstrasse 61, 8001 Zürich  
Telefon 01/45 93 33  
weitere Filialen in Bern, Chur und Basel

# Minis werden Riesen mit adcomp...



## ...X80SP

### Intelligenter Plotter und Drucker in einem Gerät:

Generatoren für Vektor, Ellipse, Rechteck, Gitter, Achse, Symbol; einfache Befehle. Positionierung: absolut, relativ, Einzelpunkt, Image, Grafik. 3 Zeichengeneratoren je 128 Zeichen, Format 8 x 8, einer frei programmierbar. Negativdruck, barcodefähig. Programmierbar: Linienart, Schreibrichtung, Schriftgröße, Zeilenabstand, Tabulatoren, Formularlänge. Formatierter Druck, Print using. Höchster Datendurchsatz, Druckwegoptimierung, Kurzwegtechnik. Bidirektionaler Papiertransport und Druck, Stachelwalze. Hardcopy mit Puffer. Digitizer, Positionierung über Tastatur. Z80 mit 10k Betriebssystem. Anschließbar an alle Rechner. Schnittstellen CBM, RS 232 C, 20 mA, V 24 oder Centronics parallel. DIN-A4-Format.

## AF 104

### Zuverlässige Floppy für CBM-Rechner:

Schnelles DOS mit automatischer Dateiverwaltung. Random-Pointer. Search und Copy. Paßwortschutz. Für kaufmännischen und technisch-wissenschaftlichen Einsatz. 8-Zoll-Laufwerke. Hardsektoriert 1 Mio byte, softsektoriert, IBM 3741-kompatibel 0,5 Mio byte.

## AP11

### Winchester-Massenspeicher für CBM-Rechner:

11 Mio byte Festplatte. Kurze Zugriffszeiten, automatische Dateiverwaltung.

## MU 80

### Multi-User-Option für Floppy und Platte

Echte verteilte Intelligenz. 8 Benutzer, Paßwortschutz. Lieferbar in AF 104 und AP 11.



DIALOG COMPUTER  
TREUHAND AG  
Seeburgstrasse 18  
6002 Luzern  
☎ 041-3145 45

## Elektronische Datenverarbeitung (EDV) ist heute nicht mehr wegzudenken.

EDV-Probleme gibt es viele –  
sie zu lösen ist unsere Stärke.

16 Jahre EDV-Erfahrung in Organisation, Analyse  
und Programmierung.

Wir beraten Sie neutral und kompetent in allen  
Organisations- und EDV-Fragen.

- EDV-Konzepte
- Pflichtenheft

unsere COBOL/BASIC-Software ist

- seriös
- zuverlässig
- fachlich nach den neusten Methoden
- preisgerecht

Bevorzugte HARDWARE

- Honeywell Bull QUESTAR/M
- SUPERBRAIN
- SESAM III

Rufen Sie uns an, bevor Sie sich in Sachen EDV  
entscheiden.

## DATABRAIN AG

EIGENES RECHENZENTRUM

8623 Wetzikon  
Bahnhofstrasse 261

Telefon 01 930 03 06  
01 930 03 07

Zu verkaufen

## Kienzle 6000 Magnetkontencomputer

mit sämtlichen Finanz-/Debitoren- und Kreditorenprogrammen,  
die sich schon einige Jahre in der Treuhandpraxis bewährt  
haben.

Der Computer ist in einem äusserst gepflegten und guten Zu-  
stand mit Service-Abonnement.

F. Brunner Treuhand, Basel  
Tel. 061 / 57 11 56



## PET/CBM-Besitzer Kennen Sie schon das neue SYNTAX- Kassetten-Magazin?

Die Programme in MICROSOFT-BASIC können im  
Abonnement bezogen werden. Monatlich erscheint  
eine Kassette mit 5 verschiedenen Programmen in  
deutscher Sprache zum günstigen Programm-  
Preis von nur ..... 3,- DM

Fordern Sie gleich heute noch kostenlose  
Informationen von

**SYNTAX**  
Soft- und Hardware GmbH  
7550 Rastatt, Postfach 16 09

## PPC-Plotter

J. BERGER, M. SPERBER, G. SCHLÜTER

Das nachfolgend vorgestellte Programm ermöglicht das Plotten beliebiger Funktionen auf dem TI 58/59/PC 100 mit einer Auflösung von 60 Punkten über die volle Papierbreite. Das Programm das uns von einer Autorengemeinschaft zugestellt wurde, verwendet viele Spezialtricks, mit denen der Drucker überlistet wird. Unseres Wissens wurden ähnliche Trickprogramme bisher noch in keiner Fachzeitschrift veröffentlicht.

Die folgenden Programme ermöglichen das Plotten beliebig vieler Funktionen auf dem PC100 mit einer Auflösung von 60 "Punkten" über die Papierbreite. Eine derartige Steuerung ist in der Bedienungsanleitung des Druckers PC 100 nicht vorgesehen, so dass eventuelle Beschädigungen des Druckers nicht ausgeschlossen sind.

Nach Auskunft der Autoren laufen diese Programme auf deren Rechnern jedoch bereits seit längerer Zeit ohne das irgendwelche Störungen auftraten. Trotzdem möchten wir unsere Leser darauf hinweisen, dass die Benutzung dieser Programme auf eigenes Risiko geschieht und wir jede Haftung ablehnen.

### WIE FUNKTIONIEREN DIESE PROGRAMME?

Es ist bekannt, dass es Tastenfolgen gibt, die ein recht merkwürdiges Verhalten sowohl des Rechners als auch des Druckers verursachen können.

Beim hier beschriebenen Verfahren werden von jedem Zeichen nur die ersten 3 oder 4 Dot-Reihen ausgedruckt. Das Programm setzt jeden eingegebenen Wert in eines der Zeichen "L", "I" oder "J" um. Da jeweils nur die ersten Dots gedruckt werden, ergeben sich 3 etwa gleichartige Zeichen, die zum Plotten verwendet werden können. Die Zeichen haben zwar eine unterschiedliche Gestalt, werden aber beim Plotten als gleichwertig angesehen.

Der Vorschubmotor arbeitet nicht immer exakt, weshalb die Graphen zeitweise etwas gestreckt oder zusammengedrängt erscheinen.

Weshalb nun gleich 3 Programme? Mit PLOT 60 kann man viele Kurven drucken und so komplexe Graphen erzeugen, PLOT 60 (1) ist sehr schnell und PLOT 60 (2) ist schnell und meist leistungsfähig genug.

Beachten Sie bei allen 3 Programmen folgende Punkte:

1. Beim Aufruf einer Funktion befindet sich X in der Anzeige und in R05, nach berechneter Funktion muss sich der Funktionswert  $f(x)$  im Anzeigeregister befinden. Verwenden Sie auf keinen Fall Labels in Ihrer Funktion. Alle nicht benötigten Speicher, alle HIR-Register sowie das t-Register stehen Ihnen zur Verfügung.
2. Nachdem Sie die Programme initialisiert haben, dürfen keine Labels (mit Ausnahme von A' und E') aufgerufen werden.
3. Tasten Sie die Programme in den Rechner ein und zeichnen Sie sie mit 6 OP 17 auf eine Magnetkartenseite auf.

### PLOT 60 (1)

Mit diesem Programm können Sie eine Funktion mit einer Auflösung von 60 Punkten auf einem Papierstreifen plotten.

### INITIALISIERUNG

(Programm einlesen oder eintasten); GTO 024; 10 OP 17 (4 OP 17 für TI 58); CLR; Pgm 19; Sbr 045; P-R; LRN; INS; LRN; RST; CLR; 6 OP 17 (3 OP 17 für TI 58);

Wenn Sie das Programm einmal so initialisiert haben, können Sie Ihre Funktion definieren und den Plotvorgang starten:

1. Eingabe der Funktion  
A' BST  
 $f(x)$  RTN
2. Eingabe der Plotparameter:
  - a. Anzahl der zu plottenden Punkte:  
n E
  - b. Untere Grenze des Plotbereichs:  
Ymin R/S
  - c. Obere Grenze des Plotbereichs:  
Ymax R/S
  - d. Anfangswert für X:  
Xo R/S
  - e. Schrittweite für X:  
delta X R/S
3. Für eine weitere Funktion: weiter bei 1., jedoch nicht A' BST sondern GTO 145 LRN drücken.
4. Fortsetzung des Plotvorgangs: Wenn der Rechner nach n geplotteten Punkten anhält, Sie die Kurve aber fortsetzen möchten, führen Sie nochmals Schritt 2a durch und drücken dann SBR 045

### SPEICHERBELEGUNG:

- |    |                                |
|----|--------------------------------|
| 00 | n                              |
| 01 | Ymin                           |
| 02 | Ymax                           |
| 03 | X0                             |
| 04 | delta X                        |
| 05 | X                              |
| 06 | Indirekt-Speicher (für OP 1-4) |
| 07 | Stelle im OP-Bereich           |

# PPC - Die Programmierbaren

Speicherverteilung änderbar zwischen 1 OP 17 und 10 OP 17

## LABELS:

C belegt  
E Eingabe der Parameter  
A' Funktion

## BEISPIELE ZU PLOT 60 (1)

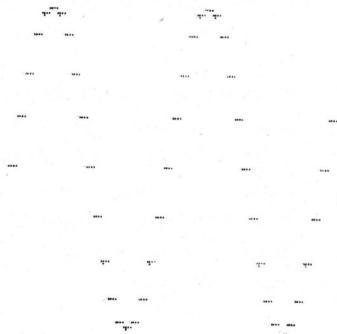
Als Beispiel soll ein Sinus geplottet werden. Die Eingaben für die Plotparameter sind natürlich nicht verbindlich.

Y liegt zwischen den Grenzen +1 und -1. Wir lassen 42 Punkte drucken, beginnen bei  $X = 0$  und wählen als Schrittweite  $\Delta X = 18$ . Sicherheitshalber drücken Sie vor den Eingaben noch DEG.

1. Eingabe der Funktion:  
A'; BST; sin; INV SBR;

2. Eingabe der Plotparameter:  
a, 42 E (n)  
b, 1 +/- R/S (Ymin)  
c, 1 R/S (Ymax)  
d, 0 R/S (X0)  
e, 18 R/S (delta X)

Nach einigen Minuten Wartezeit erhalten Sie den untenstehenden Ausdruck.



## PLOT 60 (2)

Mit diesem Programm können Sie zwei voneinander unabhängige Funktionen mit einer Auflösung von 60 Punkten auf einem Papierstreifen plotten.

Wenn Sie das Programm initialisiert haben, können Sie Ihre Funktionen eintasten und den Plotvorgang starten:

1. Eingabe der ersten Funktion  
B' BST  
f (x) RTN

2. Notieren Sie sich den Programmschritt auf dem der Rechner jetzt steht!

3. Eingabe der zweiten Funktion:  
f (x) RTN

4. Eingabe der Anfangsadresse der zweiten Funktion:  
GTO A'  
LRN GTO

Geben Sie nun den Programmschritt ein, den Sie sich bei 2. gemerkt haben:  
xxx LRN

5. Eingabe der Plotparameter:

a. Anzahl der zu plottenden Punkte  
n E  
b. Untere Grenze des Plotbereichs  
Ymin R/S  
c. Obere Grenze des Plotbereichs  
Ymax R/S  
d. Anfangswert für X  
X0 R/S  
e. Schrittweite für X  
delta X R/S

6. Für neue Funktionen: Weiter bei 1, jedoch nicht B' BST sondern  
GTO 218 LRN

7. Zur Fortsetzung eines beendeten Plotvorganges: Wiederholen Sie 5a und drücken Sie danach  
SBR 051

## SPEICHERBELEGUNG:

00 n  
01 Ymin  
02 Ymax  
03 X0  
04 delta X  
05 X  
06 Indirekt für OP 1-4 (Funktion 2)  
07 Stelle im OP-Bereich (Funktion 2)  
08 Druckzeichen (Funktion 2)  
09 Indirekt für OP 1-4 (Funktion 1)

10 Stelle im OP-Bereich (Funktion 1)  
11 Druckzeichen (Funktion 1)

Speicherverteilung zwischen 2 OP 17 und 9 OP 17

## LABELS:

C belegt  
E Eingabe  
A' Funktion 2  
B' Funktion 1

Es wird das Mastermodul benötigt.

## BEISPIELE ZU PLOT 60 (2)

Als Beispiel soll ein Sinus-, Cosinusplot gezeichnet werden.

Wir wählen als Untergrenze für den Plotbereich -1 (Ymin), als Obergrenze +1 (Ymax). Der Ausdruck soll bei  $X_0 = 0$  beginnen und in Schritten von  $\Delta X = 4.5$  Grad erfolgen. Es sollen 81 Punkte gedruckt werden. Falls die Kurven sich schneiden, soll der Sinus durchgehend gedruckt werden, deshalb ist der Sinus unsere Funktion !!

Eingabe der ersten Funktion:  
B' BST  
ins INV SBR  
Merken Sie sich, auf welchem Schritt der Rechner steht! (Programmstep 220)

Eingabe der zweiten Funktion:  
cos INV SBR LRN

Eingabe der Anfangsadresse der Funktion 2:

GTO A' LTN GTO

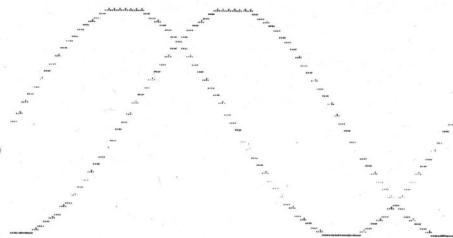
Programmstep eintasten:  
220 LRN  
(Programmstep 220)

Eingabe der Plotparameter:

(n) 81 E  
(Ymin) 1 +/- R/S  
(Ymax) 1 R/S  
(X0) 0 R/S  
(delta X) 4.5 DEG R/S

# PPC - Die Programmierbaren

Warten Sie einige Minuten und Sie sehen den unten abgebildeten Ausdruck.



PLOT 60

Mit diesem Programm können Sie beliebig viele Funktionen mit einer Auflösung von 60 Punkten auf einem Papierstreifen plotten.

Wenn Sie das Programm initialisiert haben, können Sie Ihre Funktionen definieren und den Plotvorgang starten: Instruktionen für TI 59:

1. Eingabe der Funktionen:  
GTO 305  
LRN

Eingabe der ersten Funktion:  
f1 (x) A

Eingabe der zweiten Funktion:  
f2 (x) A

Eingabe der dritten Funktion:  
f3 (x) A

Wiederholen Sie dies so oft wie nötig, nach jeder Funktion programmieren Sie "A", nach der letzten Funktion aber:  
fm (x) D GTO  
305  
LRN

2. Eingabe der Plotparameter:

a Anzahl der zu plottenden Punkte  
n E

b Untere Grenze für den Definitionsbereich  
Ymin R/S

c Obere Grenze des Definitionsbereiches  
Ymax R/S

- d Anfangswert für X  
X0 R/S
  - e Schrittweite für X  
delta X R/S
- Damit Sie falsche Eingaben, die Sie bemerkt haben, noch korrigieren können, hält der Rechner nochmal an. Zur Korrektur weiter bei 2a, sonst:  
R/S

3. Für neue Funktionen weiter bei 1.
4. Zur Fortsetzung des Plotvorgangs wiederholen Sie 2a und drücken kann weiter bei 2e  
SBR 295

## SPEICHERBELEGUNG:

- 00 n
- 01 Ymin
- 02 Ymax
- 03 X0
- 04 delta X
- 05 X
- 06 Zähler (für Querlinie)
- 07 Zeiger für f (x)
- 08 Zeiger für f (x)
- 09 OP 01
- 10 OP 02
- 11 OP 03
- 12 OP 04
- 13 belegt bis Speicher "(Anzahl der Funktionen + 12)"

## LABELS:

- A Eingabe f(x)
- C belegt
- D Eingabe fm (x) und Druck
- E Eingabe

Es wird das Mastermodul benötigt.

## BEISPIEL ZU PLOT 60

Plotten Sie einen Graphen von Sinus, Cosinus, Sinus - Cosinus, Sinus \* Cosinus und Nulllinie!

Wir wählen als Untergrenze für den Plotbereich -1.42 und als Obergrenze +1.42. Der Ausdruck soll im Abstand von 4.5 Grad erfolgen und insgesamt wollen wir 160 Punkte geplottet haben. Als Ausgangspunkt wählen wir X0 = 0 Grad.

Eingabe der Funktionen:  
GTO 305 LRN

Erste Funktion:  
sin + RCL 05 cos = A

Zweite Funktion:  
CLR A

Dritte Funktion:  
sin x RCL 05 cos = A

Vierte Funktion:  
sin A

Fünfte und letzte Funktion:  
cos D GTO 305 LRN

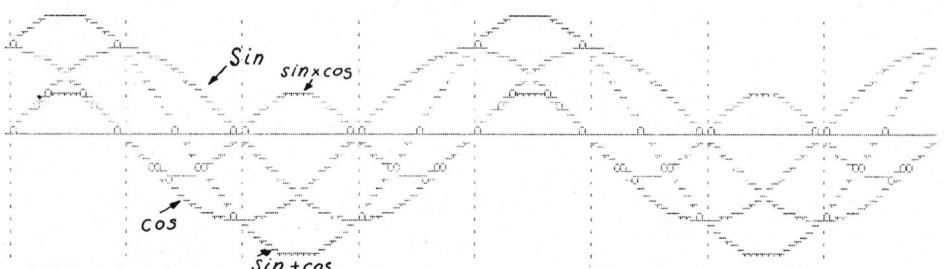
## Eingabe der Plotparameter:

n 160 E  
Ymin 1.42 +/- R/S  
Ymax 1.42 R/S  
X0 0 R/S  
delta X 4.5 DEG R/S

Start des Plottens R/S

## INSTRUKTIONEN FUER TI 58:

1. Beachten Sie, dass für den TI 58 ein ab Schritt 278 verändertes Listing gilt! Geben Sie dies mit 3 OP 17 ein.



# PPC - Die Programmierbaren

## 2. Eingabe der Funktionen:

GTO 310 LRN

Eingabe der ersten Funktion

f1 (x) SBR 295

Eingabe der zweiten Funktion

f2 (x) SBR 295

etc. bis

Eingabe der letzten Funktion

fm (x) SBR 302

GTO 310

## 3. Eingabe der Plot-Parameter:

a Anzahl der zu plottenden Punkte

n E

b Untere Grenze des Plotbereichs

Ymin STO 01

c Obere Grenze des Plotbereichs

Ymax STO 02

d Anfangswert für X

X0 STO 03 STO

05

## e Schrittweite für X

delta X STO 04

Starten des Plotvorgangs

R/S

## 4. Neue Funktionen weiter bei 2

## 5. Zur Fortsetzung des beendeten

Plotvorganges geben Sie n ein

n STO 00

und starten Sie den Plotvor-

gang

RCL 05

SBR 310

PLOT 60 (1)				PLOT 60 (2)										
000	92	RTN	046	05	05	096	69	DP	046	43	RCL	096	28	28
001	76	LBL	047	16	A'	097	26	26	047	03	03	097	85	+
002	16	A'	048	75	-	098	95	=	048	42	STD	098	43	RCL
003	61	GTD	049	43	RCL	099	52	EE	049	05	05	099	10	10
004	01	01	050	01	01	100	32	XIT	050	25	CLR	100	42	STD
005	45	45	051	95	=	101	08	8	051	43	RCL	101	07	07
006	76	LBL	052	55	+	102	75	-	052	05	05	102	43	RCL
007	15	E	053	53	(	103	02	2	053	17	B'	103	11	11
008	48	EXC	054	43	RCL	104	65	x	054	71	SBR	104	71	SBR
009	00	00	055	02	02	105	43	RCL	055	01	01	105	01	01
010	92	RTH	056	75	-	106	07	07	056	57	57	106	30	30
011	48	EXC	057	43	RCL	107	95	=	057	42	STD	107	84	DP*
012	01	01	058	01	01	108	28	INV	058	08	08	108	09	09
013	92	RTN	059	54	)	109	28	LDG	059	43	RCL	109	13	C
014	61	GTD	060	65	x	110	52	EE	060	09	09	110	43	RCL
015	00	00	061	05	5	111	22	INV	061	42	STD	111	04	04
016	33	33	062	09	9	112	52	EE	062	06	06	112	44	SUM
017	68	NOP	063	85	+	113	65	x	063	43	RCL	113	05	05
018	76	LBL	064	01	1	114	00	0	064	10	10	114	97	DSZ
019	13	C	065	95	=	115	67	EQ	065	42	STD	115	00	00
020	25	CLR	066	59	INT	116	01	01	066	07	07	116	00	00
021	69	DP	067	32	XIT	117	28	28	067	43	RCL	117	51	51
022	05	05	068	05	5	118	01	1	068	05	05	118	91	R/S
023	68	NOP	069	09	9	119	67	EQ	069	16	A'	119	71	SBR
024	74	SM*	070	77	GE	120	01	01	070	71	SBR	120	01	01
025	80	80	071	00	00	121	26	26	071	01	01	121	28	28
026	02	2	072	75	75	122	02	2	072	57	57	122	84	DP*
027	68	NOP	073	32	XIT	123	05	5	073	42	STD	123	06	06
028	68	NOP	074	00	0	124	93	.	074	11	11	124	61	GTD
029	92	RTH	075	77	GE	125	02	2	075	43	RCL	125	00	00
030	68	NOP	076	00	00	126	93	.	076	09	09	126	99	99
031	68	NOP	077	85	85	127	02	2	077	32	XIT	127	43	RCL
032	48	EXC	078	32	XIT	128	07	7	078	68	NOP	128	08	08
033	02	02	079	55	+	129	69	DP	079	00	00	129	32	XIT
034	92	RTN	080	01	1	130	00	00	080	43	RCL	130	00	0
035	48	EXC	081	05	5	131	95	=	081	06	06	131	67	EQ
036	03	03	082	75	-	132	84	DP*	082	22	INV	132	01	01
037	92	RTN	083	59	INT	133	06	06	083	67	EQ	133	44	44
038	48	EXC	084	42	STD	134	13	C	084	01	01	134	01	1
039	04	04	085	06	06	135	43	RCL	085	20	20	135	67	EQ
040	43	RCL	086	95	=	136	04	04	086	43	RCL	136	01	01
041	03	03	087	65	x	137	44	SUM	087	10	10	137	42	42
042	42	STD	088	05	5	138	05	05	088	32	XIT	138	02	2
043	05	05	089	75	-	139	97	DSZ	089	43	RCL	139	05	5
044	25	CLR	090	59	INT	140	00	00	090	07	07	140	93	.
045	43	RCL	091	42	STD	141	00	00	091	67	EQ	141	02	2
			092	07	07	142	45	45	092	00	00	142	93	.
			093	95	=	143	91	R/S	093	99	99	143	02	2
			094	65	x	144	31	LRN	094	71	SBR	144	07	7
			095	03	3	145	00	0	095	01	01	145	59	INT



# NEUHEIT für COMMODORE-COMPUTER

3000er Serie  
4000er Serie  
8000er Serie

Soft:  
**BASIC-COMPILER**

DM 1200.- + 13% Mwst.

Andere Produkte  
CBM-COMPUTER  
Floppy, Drucker  
Olivetti-Interface  
Computhink-Floppy  
A/D-Wandler  
Hochauflösende  
Graphik mit  
64000 Bildpunkten  
u. v. a. m.



Hard:  
**UPGRADE-SYSTEM**

3064, 4064 und 8064

Erweitern Sie Ihren CBM  
um 64 K-Byte  
(oder 128 K-Byte) auf  
96 K-Byte  
(für Programme und Daten!)  
ab DM 2300.- + 13% Mwst.

Händleranfragen  
erwünscht



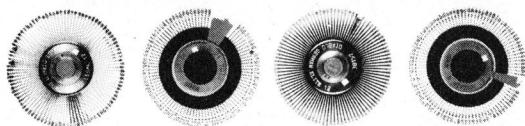
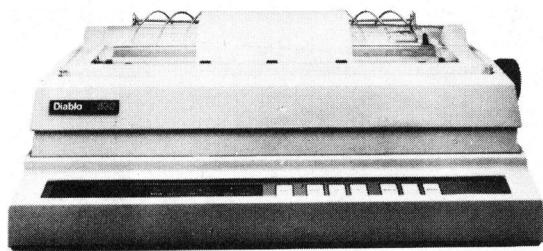
Info durch

# SPIMA · COMPUTER · GMBH

Turbinenstrasse 4 \* D-6800 Mannheim-31 \* Tel. ☎ 0621/721515 \* Telex 0463708 spimad

## Diablo 630

Der Drucker, auf dem erstmals alle erhältlichen  
Typenrad-Versionen laufen. Metall und Plastik.



Der 630 hat noch weniger mechanisch bewegte Teile als  
je ein Typenradrunder zuvor. Und er druckt vorwärts  
wie rückwärts.

Ob mit 88, 92 oder 96 Zeichen. Auch das ist einmalig.

Der neue Typenradrunder 630:  
Das Beste, was Ihnen passieren kann,  
seit Diablo das Typenrad erfunden hat.

Generalvertretung für die Schweiz und Liechtenstein:

### STUDER ELECTRONIC AG

Computer-Peripherie • Datentechnik • OEM-Produkte

Kappelenring 69 CH-3032 Hinterkappelen Tel. 031 362236 / Telex 33633



### ZEV ELECTRONIC AG COMPUTER DIVISION

Tramstrasse 11, 8050 Zürich, ☎ 01 312 22 67

#### SOFTWARE

<b>CP/M</b>		<b>APPLE</b>		<b>TRS 80/V.G.</b>	
CP/M 1.4	280.-	KRAM 2.0	380.-	ED/ASSEM.	98.-
CP/M 2.2	380.-	APPLE POST	280.-	BASEX COMP.	85.-
MBASIC 52	740.-	PROGRAM AID	125.-	MON. DEBUG. DIS.	90.-
BAS 80	740.-	INFOTREE	255.-	DEBUGGER	45.-
LINK 80	310.-	M-CAT	125.-	Z-80 DISASS.	90.-
CBASIC 2	275.-	STAT. ANAL.	380.-	MIOSYS	50.-
FORTRAN 80	885.-	APPLE LOCK	78.-	BRIDGE 2.0	59.-
COBOL 80	1420.-	COMPU READ	125.-	AIRRAID	39.-
PASCAL/M	610.-	GRAPHICS	155.-	BARRICADE	39.-
USCD PASCAL	1400.-	MICRO MEMO	125.-	TIME TREK	45.-
M/T PASCAL	560.-	MASTER CAT.	125.-	INVADERS	45.-
BUCHHALTUNG	2700.-	GRAP. FIT	78.-	SARGON II	99.-
ADRESSEN VW	820.-	PLANETARIUM	98.-	MISSION	49.-
WORDSTAR	930.-	MYSTERY H.	78.-	MYSTERY TUM.	49.-
MQSORT	240.-	AMBUSH	180.-	KEY EDIT	49.-
u. v. a.		u. v. a.		u. v. a.	
all on disk 5/8"		all on disk		all cass. 16K/LII	



Zur Unterstützung unserer  
grossen Apple-Kundschaft suchen  
wir für den **SCC-Computer-Shop**  
in Luzern einen engagierten

## Berater



**Schweizer Computer Club**  
Seeburgstrasse 18  
6006 Luzern  
Telefon 041 - 31 45 45

## OSI-Board-Erweiterung

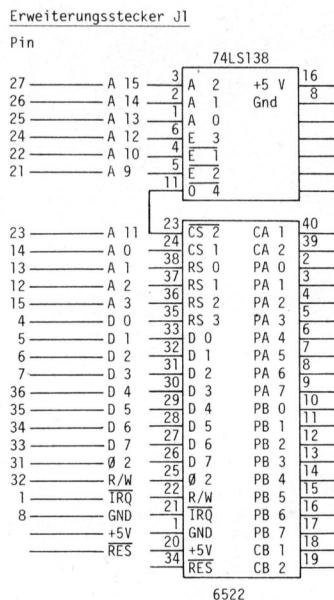
Hansjörg NIPP

Ein grosser Nachteil des OSI-Superboard ist das Fehlen von Ein-/Ausgabeports. Mit der unten beschriebenen Schaltung wird dieser Nachteil aufgehoben. Es wurde dabei Wert auf einen möglichst geringen Bauteileaufwand gelegt.

Der Interfacebaustein 6522 erweitert das Superboard um zwei Ports mit je 8 I/O-Leitungen sowie 4 Steuerleitungen, also insgesamt 20 Leitungen. Die volle Leistungsfähigkeit dieses Bausteins soll hier nicht beschrieben werden. Dazu sei auf die entsprechende Literatur verwiesen.

Zu erwähnen ist, dass der VIA 6522 nicht nur I/O-Leitungen, sondern auch 2 Timer und ein Schieberegister zur seriellen Datenübertragung enthält.

### SCHALTUNGSBESCHREIBUNG



Die gesamte Schaltung zeigt Bild 1. Um den Hardwareaufwand gering zu halten, wird die Adressdekodierung nur unvollständig mit dem 8 aus 1-Dekoder 74LS138 vorgenommen

(Adressbereich des VIA: 9800-9FFF). Diese Vereinfachung wird erst hinderlich, wenn das System stärker erweitert werden soll. Das ist aber ohnehin nur mit einem grösseren Aufwand möglich, da dann sämtliche Signale gepuffert werden müssten.

Die Verbindung der Schaltung mit dem Superboard erfolgt über ein 25-adriges Kabel. Davon werden 23 Adern direkt dem Erweiterungsstecker J1 (40-poliger Sockel) zugeführt (Bild 2). Die positive Versorgungsspannung von +5 V entnimmt man dem Netzteil des Superboard. Die RES-Leitung muss separat verlötet werden (Bild 2). Die letzten zwei Leitungen sind auf dem Erweiterungsstecker nicht enthalten.

Im Schaltplan des Superboard sind zur Datenpufferung zwei bidirektionale Datenpuffer (U6, U7) eingezeichnet. Diese zwei Bauteile sind aber nicht in ihre Sockel eingesteckt (sollten sie doch vorhanden sein, sind sie zu entfernen). Um den Datentransport zu ermöglichen, müssen die in Bild 3 gezeich-

neten Verbindungen zwischen den Pins gemacht werden (im einfachsten Falle mit ein paar Drahtstücken).

### DIE PORTS ALS EIN-/AUSGABE-LEITUNGEN

Wie die Ports als Ein- und Ausgänge verwendet werden, soll an einem Beispiel gezeigt werden. Ein Port muss keineswegs nur als Eingang oder Ausgang betrieben werden, sondern kann auch gemischt programmiert werden. Dazu schreibt man zuerst ein Bitmuster in das Datenrichtungsregister DDRA (Port A) bzw. DDRB (Port B). Eine "1" schaltet den zugehörigen Pin auf Ausgang, eine "0" auf Eingang. Das Ausgangssignal wird erst durch das Einschreiben in das Ausgangsregister ORA (ORB) erzeugt. Das Einlesen erfolgt im einfachsten Falle durch lesen des Registers ORA (ORB).

In unserem Schaltungsbeispiel sind sämtliche Adressierungen der internen Register aus der folgenden Tabelle zu entnehmen:

Adresse hexadezimal	Register
9800	ORB Ausgaberegister Port B

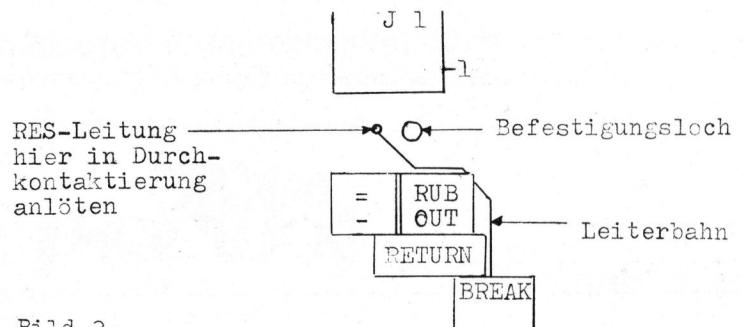


Bild 2

Adresse	Register				
		9809	T2C-H Zähler HIGH lesen und schreiben	LDA = \$DF	Ausgangssignale ins
9801	ORA Ausgaberegister Port A	980A	SR Schieberegister	STA \$9800	Ausgangsregister geben
9802	DDRB Datenrichtungsregister Port B	980B	ACR Hilfsregister	LDA \$9800	Port B einlesen
9803	DDRA Datenrichtungsregister Port A	980C	PCR Steuerregister		
	Timer 1:	980D	IFR Interrupt-Anzeigeregister		
9804	TIC-L Zähler lesen LOW	980E	IER Interrupt-Freigaberegister		
	Vorspeicher Low	980F	ORA Ausgaberegister für Port A ohne Handshake, im Gegensatz zu 9801		
9805	TIC-H Zähler HIGH lesen und schreiben				
9806	TIL-L Vorspeicher LOW lesen oder schreiben				
9807	TIL-H Vorspeicher HIGH lesen oder schreiben				
	Timer 2:				
9808	T2C-L Zähler LOW lesen				
	Vorspeicher LOW schreiben				

Als Beispiel sollen PB 1-4 als Eingänge und PB 5-7 als Ausgänge programmiert werden. Anschliessend werden die Ausgänge PB 5=low, PB 6 und PB 7=high gesetzt und die Eingänge eingelesen:

LDA = \$70 Ein-/Ausgänge festlegen  
 STA \$9802 und in Datenrichtungsregister schreiben

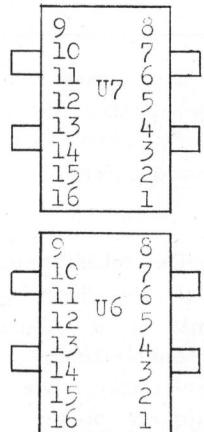
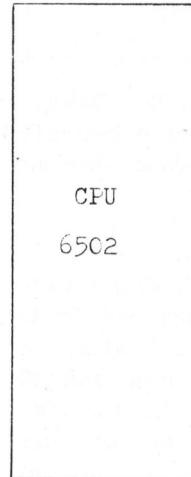


Bild 3

## COMPUTERSCHULE SCHAFFHAUSEN

Kurse: BASIC I  
 BASIC II  
 Strukturierte Programmierung  
 Informationskurse «Microcomputer»  
 Alle Kurse mit max. 10 Teilnehmern

Fr. 380.-  
 Fr. 450.-  
 Fr. 780.-  
 Fr. 90.-

Kursunterlagen CSS Tel. 053 - 45 45 0

## COMPUTERSCHULE SCHAFFHAUSEN

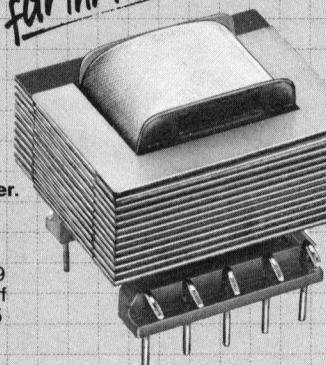
## 5 Kleintransformatoren für Ihr Power-Pack

Offene Netztrafos (Primär 220V) für direkte Printmontage in 5 Grössen von 1,2 bis 11,8 VA und für Sekundärspannungen von 2 x 4,5V bis 2 x 15V. DIN 41302.

Wir liefern innert 24 h ab Lager. Verlangen Sie den Katalog!



D. Leitgeb AG  
 Überlandstr. 199  
 8600 Dübendorf  
 Tel. 01-8201545  
 Telex 55547



LEITGEB DISTRIBUTION

Ihr **Commodore PET-CBM** ist ein besserer **POKER-Spieler**, als Sie denken!

Unser Programm beweist es!

Kassette: **SFr. 39.50** (einschliesslich Verpackung und Versand)

in Englisch, Deutsch oder Französisch (Anleitung im Programm enthalten). Läuft auf allen Modellen mit min. 8 K.

Senden Sie Ihre Bestellung an:

### MICRO SOLUTIONS

28, chemin des Longs-Prés  
 1232 Confignon (GE)  
 Telefon (022) 57 48 34

Ich bestelle:  Kassette(n)

Name: \_\_\_\_\_

Adresse: \_\_\_\_\_

Engl.  Deutsch  Franz.

## 2m Contest

Peter DAMBROWSKI

Dieses Programm ist in erster Linie für Funkamateure gedacht, die an 2m Funkwettbewerben teilnehmen. Es kann aber selbstverständlich auch von jedem anderen Funkamateurer benützt werden, der sich die Logbuchführung beim abendlichen QSO-fahren etwas erleichtern will.

Das Programm läuft auf einem AIM 65 mit 12K RAM und TV-Display mit direktem Speicherzugriff. Da es in BASIC geschrieben ist, lässt es sich auch - nach kleinen Änderungen wie z.B. PRINT! - an jeden anderen Basic-Computer anpassen. Sollte jemand mit dem AIM 65 ohne Bildschirm arbeiten, müssen die zu erfragenden Texte im Programm sinnvoll auf eine Zeile mit max. 20 Buchstaben zusammengedrückt werden.

Einige Besonderheiten (Bild 1) sollen noch erwähnt werden. Der Befehl in Zeile 40 bedeutet: Schirm löschen, Cursor home. Der Befehl in den Zeilen 595 und 625 bedeutet: Umschalten von normal auf reverse und umgekehrt.

Der Befehl PRINT ! ist eine Besonderheit des AIM 65 und gibt alles Nachfolgende in dieser Zeile auf dem Schirm und mit dem Drucker aus. Der Drucker sollte vor dem Programmstart ausgeschaltet werden.

Bei der Eingabe auf eine Frage des AIM dürfen keine Kommas verwendet werden, da der Computer alles nach dem Komma ignorieren würde.

In Zeile 50 in den Klammern ist die Anzahl der möglichen Rufzeichen eingetragen. Sie muss verkleinert werden, wenn der AIM "OM ERROR" ausgibt.

### PROGRAMMBESCHREIBUNG

Nach Eingabe und Überprüfung starten Sie das Programm mit RUN. Der Computer fragt: NUR CALL-VERGLEICH (1) ODER CALL-VER-

GLEICH MIT LOGAUFEICHNUNG (2). Wenn Sie eine 1 eintippen, nimmt der AIM nur Rufzeichen an. Er sucht dann im Speicher, ob das Rufzeichen schon gearbeitet wurde. Wenn ja, teilt er Ihnen dies mit. Wenn nein, fordert er Sie zur Eingabe des 2. Calls auf. Tippen Sie eine 2 ein, wird zusätzlich noch nach Frequenz, Rapport, Zeit und QRA-Kenner gefragt. Beenden Sie jede Antwort mit Return.

Danach fragt der AIM "DATENSICHERUNG J ODER N ?". Beantworten Sie die Frage mit J (= Ja), werden nach jedem 5. Call die letzten 5 Rufzeichen mit den jeweiligen Daten ausgedruckt.

#### DATEN-SICHERUNG QSO-NR. 1 - 5

-----  
Nr. 1 DC4SG  
144.350MHZ  
RST:59001 12.45UHR  
QRA:EI17F

-----  
Nr. 2 DK3SI  
144.180MHZ  
RST:56009 13.15UHR  
QRA:EJ22G

-----  
Nr. 3 DC5SS  
144.205MHZ  
RST:41012 13.25UHR  
QRA:EJ34A

-----  
Nr. 4 DB3UH  
144.2454MHZ  
RST:58003 13.30UHR  
QRA:EI45G

-----  
Nr. 5 DK4SE  
144.280MHZ  
RST:51021 13.40UHR  
QRA:EI23D  
-----

Jetzt gibt der AIM folgendes aus:

1.QSO

CALL? Tippen Sie das Call ein. Beenden Sie wie immer mit RETURN. AIM antwortet:

CALL OK!!! QSO (J ODER N)?

Sollte das Call schon gearbeitet sein, erscheint statt obiger Antwort der Satz: AUF 2M SCHON GEARBEITET. WEITERMACHEN!!!

Die obige Frage QSO (J ODER N) bezieht sich darauf, ob es zu einer Verbindung mit dieser Station gekommen ist oder nicht.

Wurde die Frage mit J beantwortet, fährt der Computer in Abhängigkeit der Beantwortung, ob nur Call-Vergleich oder Log-Aufzeichnung gewünscht wurde, mit der Abarbeitung des Programmes weiter.

Bei Call-Vergleich fragt der Computer nach dem nächsten Call. Bei Call-Vergleich mit Log-Aufzeichnung werden noch Frequenz, Rapport, Zeit und QRA-Kenner erfragt, bevor zum nächsten Call weitergegangen wird.

Wenn Sie zwischendurch alle bisher gearbeiteten Rufzeichen ausdrucken möchten, beantworten Sie die Frage: X.QSO

CALL?

mit dem Wort DRUCK. Beenden Sie mit RETURN und der es erfolgt die Auslistung der bisher bearbeiteten Rufzeichen.

DC4SG  
DK3SI  
DC5SS  
DB3UH  
DK4SE

Haben Sie einmal das Call schon eingegeben und das QSO mit J bestätigt, wollen es aber wieder aus Ihrem Log entfernen, beantworten Sie die Frage:

X.QSO  
CALL?

mit dem Wort ZURUECK. Beenden Sie

```

40 PRINT CHR$(24)
50 DIM B%(150).R$(15)
55 PRINT "*****"
60 PRINT "* C O N T E S T / L O G - P R O G R A M M *"
65 PRINT "*****"
70 PRINT " "
80 PRINT " "
90 PRINT " "
100 PRINT " "
110 PRINT "NUR CALL-VERGLEICH (1) ODER CALL-VERGLEICH MIT"
115 INPUT "LOG-AUFZEICHNUNG (2) ";X$
116 PRINT " "
120 INPUT "DATENSICHERUNG J ODER N ";Y$
125 PRINT " "
130 B%=2
150 N=1:K=0
160 PRINT N;" QSO"
170 INPUT "CALL";C$
180 IF LEFT$(C$.7)="ZURUECK" THEN 410
200 IF LEFT$(C$.5)="DRUCK" THEN GOSUB 640
220 FOR I=1 TO N-1
230 IF R$(I)=C$ THEN 370
240 NEXT
250 PRINT "CALL OK !!!"
260 INPUT "QSO (J ODER N)";A$
270 IF A$="N" THEN A$=" ";GOTO 160
280 IF A$<>"J" THEN GOTO 260
282 IF X$="1" THEN GOTO 290
285 INPUT "FREQUENZ IN MHZ:";F$
286 INPUT "RAPPORT:";O$
287 INPUT "ZEIT (MEZ)";Z$
288 INPUT "QRA-KENNER:";Q$
290 R$(N)=C$:B%(N)=B%:A$=" "
295 F$(N)=F$
296 O$(N)=O$
297 Z$(N)=Z$
298 Q$(N)=Q$
300 N=N+1:K=K+1:F=F+1
310 IF K<5 THEN 160
312 PRINT!"DATEN-SICHERUNG"
313 PRINT!"QSO-NR. ";N-5;"-";N-1
314 PRINT!"===== "
316 PRINT!"NR. ";N-5.R$(N-5).F$(N-5);"MHZ".RST:";O$(N-5).
317 PRINT!"Z$(N-5);"UHR".QRA:";Q$(N-5)
318 PRINT!"----- "
319 PRINT!"NR. ";N-4.R$(N-4).F$(N-4);"MHZ".RST:";O$(N-4).
320 PRINT!"Z$(N-4);"UHR".QRA:";Q$(N-4)
321 PRINT!"----- "
322 PRINT!"NR. ";N-3.R$(N-3).F$(N-3);"MHZ".RST:";O$(N-3).
323 PRINT!"Z$(N-3);"UHR".QRA:";Q$(N-3)
324 PRINT!"----- "
325 PRINT!"NR. ";N-2.R$(N-2).F$(N-2);"MHZ".RST:";O$(N-2).
326 PRINT!"Z$(N-2);"UHR".QRA:";Q$(N-2)
327 PRINT!"----- "
328 PRINT!"NR. ";N-1.R$(N-1).F$(N-1);"MHZ".RST:";O$(N-1).
329 PRINT!"Z$(N-1);"UHR".QRA:";Q$(N-1)
330 PRINT!"----- "
350 K=0:GOTO 160
360 STOP
370 IF B%(I)=B% THEN GOTO 590
380 PRINT " ";
390 PRINT R$(I);" QSO NR ";I;" AUF";B%(I);"M"
400 GOTO 240
410 N=N-1
420 IF K>0 THEN K=K-1:GOTO 160
430 K=4:GOTO 160
470 GOTO 160
480 STOP
490 IF K=0 THEN 810
500 N=N+5-K:X=99
580 GOTO 540
590 PRINT " "
595 PRINT CHR$(23)
600 PRINT "*****"
610 PRINT " AUF";B%:"M SCHON GEARBEITET WEITERMACHEN!!!"
620 PRINT "*****"
625 PRINT CHR$(23)
630 GOTO 160
640 FOR T=N TO 0 STOP -1
650 PRINT! R$(N-T)
660 NEXT T
670 RETURN

```

mit RETURN und das zuletzt eingegebene Call wird mitsamt den gespeicherten Daten gelöscht. Sie können dann mit der vorherigen QSO-Nummer weitermachen.

Da der AIM alle Daten erfragt, ist der Umgang mit diesem Programm leichter, als es auf den ersten Blick erscheint.

Nun noch ein paar Worte zur Programmaufzeichnung. Es dürfte bekannt sein, dass beim AIM 65 der Audio-Gap in A409 hex 08 ist. Wenn man aber BASIC-Programme mit diesem Audio-Gap aufzeichnet, kommt es beim Laden des Programmes zu einer Fehlermeldung. Um das zu umgehen, empfiehlt es sich diesen Wert auf hex 80 zu vergrößern.

Und nun, lieber Computerfreund, fangen Sie an. 73 es 55.

**Sollen andere Personen für Sie entscheiden über etwas, das Sie in einigen Minuten Zeitinvestition pro Monat ebensogut selbst überblicken können?**

Lesen Sie das neue

**COMPUTER *Journal***  
Informationen für kommerzielle Small business computer-Anwender

Endlich eine Zeitschrift, die konzentriert **die** Informationen über Tischcomputer und deren Anwendung bringt, welche leitende Personen in Wirtschaft und Schule benötigen sowie «Insider-Informationen» für die Benutzer.

**Hier finden Sie die Grundlage für Ihre Entscheidungen von heute und morgen.**

Verlangen Sie unverbindlich ein Probeexemplar beim

**Verlag SCC AG**  
**Seeburgstrasse 12**  
**CH-6006 Luzern**

## SORCERER-Tips

In der Schweiz besteht eine sehr aktive SCC-Sorcerer-Gruppe aus deren Publikationen wir die nachstehenden Sorcerer-Tips entnommen haben. Die Kontaktadresse ist: SCC-Sorcerer-Gruppe, Herrn Werner Gribi, Industriestrasse 3, CH-3294 Büren a. A.

### INVERSE DARSTELLUNG VON BUCHSTABEN

Der Sorcerer verfügt über je 128 feste und frei definierbare Zeichen. Für die optische Hervorhebung einer Textstelle oder eines Resultates fehlt eine einfache Funktion zur Umkehrung des schwarz-weiss-Wertes eines Characters. Eine solche Funktion lässt sich jedoch leicht in BASIC programmieren, indem man den 128 freien Characters die negative Darstellung der 128 festen Characters zuordnet.

Die Subroutine 1000, die nur ein einziges Mal am Anfang des Programms aufgerufen werden muss, übernimmt diese Funktion:

```
1000 FOR I=-2048 TO -1025
1001 POKE I+1024,(255-PEEK(I))
1002 NEXT I
1003 RETURN
```

Subroutine 2000 führt für alphanumerische Strings die Umkehrung aus. Sie bearbeitet einen String T\$, dem der hervorzuhebende Text vorher zugeordnet werden muss.

Für einen String namens S\$ würde der Aufruf lauten

```
T$ = S$:GOSUB 2000
```

für eine numerische Variable X müsste man T\$ mittels der STR\$-Funktion definieren

```
T$ = STR$(X):GOSUB 2000
```

Die Subroutine 2000 lautet

```
2000 L = LEN(T$)
2001 FOR I=1 TO L
2002 T$=LEFT$(T$,I-1)+CHR$(ASC
(MID$(T$,I)+128)+RIGHT$(T$,
L-I)
2003 NEXT I
2004 RETURN
```

Zur Demonstration ein Beispiel: Zwei Zahlen, die addiert werden sollen, erscheinen im Display. Nach einer Wartezeit erscheint das optisch hervorgehobene Ergebnis.

```
10 GOSUB 1000: REM Definition der
umgekehrten Characters
20 R = 0
30 R = R-1
40 A = INT(50*RND(1)): REM erste
Zufallszahl
50 B = INT(50*RND(2)): REM zweite
Zufallszahl
60 T$ = STR$(A+B): REM Def. des
Ergebnis-Strings
70 GOSUB 2000: Umkehrung des
s-w-Wertes
80 FOR I=1 TO 30:PRINT:NEXT
90 PRINT A; "+"; B
100 FOR I=1 TO 1000: NEXT I: REM
Warteschleife
110 PRINT "="; T$: REM hervorge-
hobenes Resultat
120 FOR I=1 TO 1000: NEXT I:
Warteschleife
130 GO TO 30
```

### MONITOR-FEHLER

H. HUENI

berichtet in den Newsletters Ausgabe Oktober 1980 der Z-80 Microfans (USA):

Das Monitorprogramm des Sorcerer weist einige Unebenheiten auf. Nach Angaben der australischen Gruppe von Sorcerer-Besitzern soll ein neues Monitorprogramm in Vorbereitung sein, das diese Fehler vermeidet, im übrigen aber dieselben Routinen verwendet. Das neue Monitorprogramm soll ohne weiteres mit der

übrigen bisherigen Software (ROM Pacs usw.) zusammen arbeiten. Folgende Fehler werden genannt:

- 1) Die RS232-Schaltung des Sorcerer, die auf den seriellen Ausgang wirkt, arbeitet nicht richtig, es entstehen beim Anschluss eines Printers bei jeder Baudrate Fehler. Die Ursache liegt darin, dass die "Keyboard Scan Routine" den seriellen Ausgang vor jedem neuen Zeichen zurückgesetzt. In Sorcerer I kann dieser Fehler durch eine Schaltungsänderung behoben werden.
- 2) Auch mit der obgenannten Schaltungsänderung gehen bei 1200 Baud gelegentlich Zeichen verloren. Programme mit eigener Keyboardroutine arbeiten einwandfrei.
- 3) Es gibt kein Monitor-Kommando zur Aktivierung der RS232-Schaltung und des seriellen Ausgangs. Das neue Monitorprogramm wird zwei Befehle dafür erhalten.
- 4) Das Kommando SET 0=S (seriell) wird nur mit 300 Baud ausgeführt.
- 5) Ein mit GO aktiviertes Benutzerprogramm bleibt hängen, wenn das IY-Register modifiziert ist.
- 6) Sind in der Mitte eines Files auf dem Band eine Reihe von "0" vorhanden, so erfolgt irrtümlicherweise eine CRC-ERROR-Meldung, statt dass auf den nächsten FILE-Anfang gewartet wird.
- 7) Für die Monitorkommandos dürfen nur Grossbuchstaben verwendet werden, was bei der Textverarbeitung etwas umständlich ist. Mit einer neuen "FOLD"-Subroutine wird die entsprechende Umwandlung vorgenommen.

## EXIDY EXTENDED BASIC

In Portland, Oregon, USA gibt es eine Gruppe von SORCERER-Benutzern "Z-80 MICROFANS", die in der Oktober-Nummer ihrer Newsletters die Mitteilung einer Australian Sorcerer Users Group zitiert, wonach EXIDY nun doch noch eine Kasette mit Extended Basic herausbringt.

Vic Tolomei, Chef der Software Entwicklungs-Abteilung von Exidy, hat darüber einige Informationen gegeben. Das neue BASIC wird 19K lang sein und neben den bisherigen, eine Reihe neuer Möglichkeiten bieten. Es soll dann etwa dem Microsoft Extended Disc Basic entsprechen, aber ohne Disc-Befehle.

Wichtigste Neuerungen sind die auf 17 Digit vergrösserte numerische Genauigkeit, Linien mit bis zu 255 Zeichen und praktisch alle Möglichkeiten des "TRS 80 Level II Basic", insbesondere auch "Screen Editor" und "Print Using".

ABS	AND	ASC
ATN	AUTO	BAUD
BYE	CALL	CDBL
CHR\$	CALL	CLEAR
CLOAD	CINT	COS
CSAVE	CONT	CURSOR
DATA	CSNG	DEFDBL
DEFINT	DEF	DEFSTR
DELETE	DEFSNG	EDIT
ELSE	DIM	EQV
ERASE	END	ERR
ERROR	ERL	FIX
FN	EXP	FRE
GOTO	FOR	GO TO
HEX\$	GOSUB	IMP
INKEY\$	IF	INPUT
INT	INP	LEN
LET	LEFT\$	LIST
LLIST	LINE	LPOS
LPRINT	LOCK	MOD
NEW	MID\$	NOT
NULL	NEXT	ON
OPTION	OCT\$	OUT
PEEK	OR	POS
PRINT	POKE	RANDOMIZE
REM	RANDOMIZE	READ
RESUME	RENUM	RESTORE
RND	RETURN	RIGHT\$
SGN	RUN	SERIAL
SPC(	SIN	SPACE\$
STOP	SQR	STEP
SWAP	STR\$	STRING\$
THEN	TAP(	TAN
TRON	TO	TROFF
VAL	USING	USR
WEND	VARPTR	WAIT
XOR	WHILE	WIDTH
	UND DIVERSE ZEICHEN	
	WIE + - * /	

## TIPS ZUR BESCHLEUNIGUNG DES MICROSOFT-BASIC

Helmut VORKAUF

NEXT ohne den Index benutzen. Der Interpreter hat seine eigene Buchführung über begonnene FOR-Schlaufen, deshalb ist der Index unnötig.

```
FOR I=1 TO 10000
```

```
NEXT I
```

19 Sek

```
FOR I=1 TO 10000
```

```
NEXT
```

16 Sek, Einsparung 16 %

Eine Zahl zu sich selbst addieren statt mit 2 zu multiplizieren.

```
FOR I=1 TO 10000
```

```
Y = 2*I
```

```
NEXT I
```

60 Sek

```
FOR I=1 TO 10000
```

```
Y = I+I
```

```
NEXT I
```

47 Sek, Einsparung 22 %

Multiplizieren statt dividieren:

```
PI = 3.14
```

```
FOR I=1 TO 3000
```

```
Y = I/PI
```

```
NEXT I
```

26 Sek

```
QPI = 1/3.14
```

```
FOR I=1 TO 3000
```

```
Y = I*QPI
```

```
NEXT I
```

20 Sek, Einsparung 23 %

Aber Achtung: Die Ersparnis gilt nicht, wenn die Division durch Integer, die Multiplikation dagegen mit Fließkomma geschieht.

```
FOR I=1 TO 10000
```

```
Y = I/20
```

```
NEXT I
```

95 Sek

```
FOR I=1 TO 10000
```

```
Y = I*.05
```

```
NEXT I
```

150 Sek, 58 % mehr Rechenzeit!

Multiplikation mit sich selbst statt Quadrat.

```
FOR I=1 TO 5000
```

```
Y = I^2
```

```
NEXT I
```

236 Sek

```
FOR I=1 TO 5000
```

```
Y = I*I
```

```
NEXT I
```

30 Sek, Einsparung 87 % !!!

Variablen sind schneller als Konstanten, also Ziffern aus Schleifen herausnehmen und vor der Schleife als Variable einführen:

```
FOR I=1 TO 5000
```

```
Y = I/3.14
```

```
NEXT I
```

58 Sek

```
PI = 3.14
```

```
FOR I=1 TO 5000
```

```
Y = I/PI
```

```
NEXT I
```

26 Sek, Einsparung 55 % !!!

Wie man sieht, lohnt es sich, diese einfachen Regeln zu beachten, damit der Sorcerer schneller wird. Schliesslich ist er im Vergleich mit CBM/PET nicht der Schnellste.

Wenn man all diese Einsparungen kombiniert, ergeben sich unter Umständen ganz beachtlich schnellere Programme. Man vergleiche einmal diese beiden Programme:

```
FOR R=0 TO 7 STEP 0.1
```

```
FLACH = 3.1416 * R.2
```

```
PRINT "Radius=";R;" Fläche="
```

```
    ";FLACH
```

```
NEXT R
```

```
PI = 3.1416
```

```
FOR R=0 TO 7 STEP 0.1
```

```
FLACH = PI*R*R
```

```
PRINT "Radius=";R;" Fläche="
```

```
    ";Flach
```

```
NEXT
```

Wer kennt noch andere Tricks?

## Spezialzeichen auf Matrixdrucker

Eduard J. SIMON

Zu den meisten Kleincomputer sind auch eine ganze Reihe Printer, d.h. Drucker für den raschen Ausdruck von Texten oder Listings, erhältlich. Diese Drucker haben allerdings in der Regel nur den ASCII-Charaktersatz zur Verfügung, der aufgrund seiner amerikanischen Herkunft die für deutsche Texte notwendigen Umlaute sowie das scharfe ß nicht berücksichtigt.

Zwar kann man sich mit der Umschreibung der Umlaute als ae, oe, ue zufriedengeben. Für das scharfe ß gibt es aber bereits Schwierigkeiten, wenn beispielsweise zwischen Massen und Maßen unterschieden werden soll. Die übliche Umschreibung des scharfen ß mit ss ergibt die gleiche Schreibweise, bei doch recht unterschiedlicher Bedeutung.

Die gleichen Schwierigkeiten treten auch bei anderen europäischen Sprachen auf, wenn Akzente, z.B. in den skandinavischen Sprachen werden sollen.

Die japanischen Hersteller, die eine ganze Reihe vorzüglicher, meist recht preiswerter Drucker anbieten, haben dieses Problem, das auch für sie selbst im eigenen Land besteht, elegant gelöst, um die lateinischen Zeichen der ASCII-Norm neben den für die japanische Schrift notwendigen kana-Zeichen drucken zu können, ist bei vielen Modellen als Charaktergenerator für die Schrift ein ROM mit zwei Charaktersatzten vorgesehen.

Durch einen Schalter oder durch Umstecken einer Kurzschlussbrücke kann der eine oder andere Zeichensatz für den Druck ausgewählt werden.

In den europäischen Bedienungsanleitungen wird auf diese Möglichkeit im Detail vielfach nicht hingewiesen, da für diesen Markt nur der lateinische Zeichensatz interessant zu sein scheint.

Bei einigen Druckern, wie zum Beispiel bei dem "Microline 80" von OKI Electric, kann das ROM für den doppelten Zeichensatz, das im Gegensatz zu den anderen integrierten Schaltungen für die Druckersteuerung in einem Sockel sitzt, gegen ein EPROM 2716 (2K x 8) ausgetauscht werden.

Damit erhält man die Möglichkeit, einen persönlichen Zeichensatz zu programmieren.

Bei einer eigenen Programmierung des Charaktergenerators kann anstelle des zweiten Zeichensatzes

mit den kana-Symbolen ein vollständiger Zeichensatz kyrillischer Buchstaben, programmiert werden. Auf ähnliche Weise können auch andere Zeichen oder ganze Zeichensätze erstellt werden.

Buchstabe	Code	Buchstabe	Code
A	41	a	61
B	42	b	62
C	43	c	63
D	44	d	64
E	45	e	65
F	46	f	66
G	47	g	67
H	48	h	68
I	49	i	69
J	4A	j	6A
K	4B	k	6B
L	4C	l	6C
M	4D	m	6D
N	4E	n	6E
O	4F	o	6F
P	50	p	70
Q	51	q	71
R	52	r	72
S	53	s	73
T	54	t	74
U	55	u	75
V	56	v	76
W	57	w	77
X	58	x	78
Y	59	y	79
Z	5A	z	7A
↑	5B	{	7B
↓	5C		7C
←	5D	}	7D
→	5E	~	7E
-	5F		7F

Bild 2: ASCII-Code für Gross- und Kleinbuchstaben.

Es soll hier aber nur die Einfügung der deutschen Umlaute sowie des scharfen ß beschrieben werden.

Bild 2 zeigt die Zuordnung der einzelnen Buchstaben und Symbole des ASCII-Zeichensatzes, bei 7 bit, zu 2-stelligen Hex-Codes.

Grossbuchstaben A - Z entsprechen Hex 41 - 5A, Kleinbuchstaben a - z entsprechen Hex 61 - 7A. (Alle weiteren Angaben ebenfalls in Hex).

Von 5B bis 5F sowie von 7B bis 7F, laufen eine Reihe von Sonder-

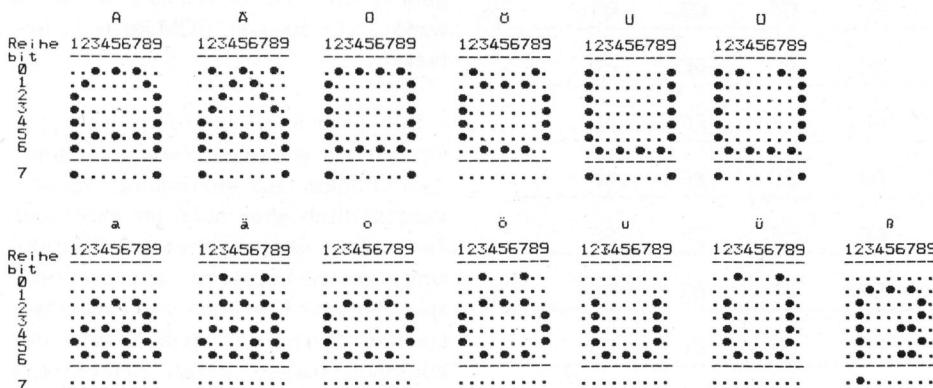


Bild 1: • = Aussans Charaktergenerator "Low" (0),  
• = Aussans Charaktergenerator "High" (1)

zeichen (siehe Bild 2), die für die Textverarbeitung nicht unbedingt notwendig sind. Statt dieser Zeichen können die Umlaute (5B - 5D und 7B - 7D) sowie das scharfe ß (7E) eingefügt werden.

Auf dem Bildschirm erscheinen zwar, wenn man bei dem hierfür zuständigen Charaktergenerator des Bildschirm-Interface keine gleichartige Aenderung vornimmt, die bisherigen Zeichen, an deren jetzige

Zuordnung zu den Umlauten man sich aber schnell gewöhnt.

Um die Programmierung vorzunehmen, wurde zuerst das im Drucker befindliche ROM kopiert und in Form eines Hexdump aufgelistet, also 16 Zeilen pro Zeile.

Wo waren aber die Signale für die Betätigung des Druckkopfes - der beim OKI Microline in einer 7 x 9 Matrix druckt - angeordnet?

Die 7 Daten-Ausgänge D0 bis D6 des Charaktergenerators steuern die 7 Drucknadeln des Druckkopfes. Dabei bedeuten ein "Low"-Ausgang Druck und ein "High"-Ausgang kein Druck. Ausgang D7 betrifft die möglichen grafischen Symbole des Druckers und bleibt hier ausser Betracht.

Damit kann in einer einzelnen Speicherzelle, mit den verfügbaren 8 bit, das Druckprogramm für eine einzelne Punktreihe festgelegt werden. Nachdem die Matrix für das einzelne Zeichen aber aus insgesamt 9 Reihen besteht, sind für jeden Buchstaben nacheinander auch 9 Speicherzellen im ROM oder EPROM anzusprechen.

Die Anordnung der einzelnen Zeilen war für jeden Buchstaben der Reihen 1 bis 8 leicht zu finden, da hierfür der Abstand jeweils 80 Hex betrug, die Codierung für die 9. Reihe wurde durch Vergleich der Sequenz der noch fehlenden Zeichen schliesslich ebenfalls gefunden.

In Bild 3 sind die Adressen und Inhalte der Zeilen für den Buchstaben A als Beispiel aufgeführt. Das sich für den Druck ergebende Bitmuster zeigt Bild 1.

Für die Programmierung der Umlaute wurden aus dem Zeichensatz des ursprünglichen ROM die Grundlaute herangezogen und, wie Bild 1 zeigt, die entsprechenden Aenderungen vorgenommen. Der Rest des Programms wurde, wie aus dem ROM kopiert, belassen.

Das ursprüngliche ROM steht jederzeit zur weiteren Verwendung mit dem Drucker zur Verfügung. Selbstverständlich aber auch um es erneut für andere Aenderungen des Charaktersatzes zu kopieren, wie zum Beispiel für das Einfügen von mathematischen Symbolen, dem Ersatz der Kleinbuchstaben durch griechische Buchstaben und vielem anderem mehr.

Bild 3 Adressen und Kodierung der Drucksignale (in Hex) für die einzelnen Grund-Laute aus dem ursprünglichen, im ROM festgelegten Programm, sowie die für die Umlaute erstellte neue Kodierung

Buchstabe Reihe	Adresse	A ..1	Ä ..B	O ..F	Ö ..C	U ..5	Ü ..D
1	04. 05.	03	0F	41	C3	40	40
2	0C. 0D.	FD	F6	BE	FE	BF	BF
3	14. 15.	EE	DB	FF	BD	FF	FE
4	1C. 1D.	FF	FD	BE	FF	BF	BF
5	24. 25.	EE	DE	FF	BD	FF	FF
6	2C. 2D.	FF	FD	BE	FF	BF	BF
7	34. 35.	EE	DB	FF	BD	FF	FE
8	3C. 3D.	FD	F6	BE	FE	BF	BF
9	21. 29.	03	0F	41	C3	40	40

Buchstabe Reihe	Adresse	a ..1	ä ..B	o ..F	ö ..C	u ..5	ü ..D	ß ..E
1	06. 07.	DF	DF	C7	C7	FF	FF	FF
2	0E. 0F.	AF	AF	FF	FF	C3	C3	03
3	16. 17.	FB	FA	BB	BA	BF	BE	FD
4	1E. 1F.	AF	AF	FF	FF	FF	FF	FF
5	26. 27.	FB	FB	BB	BB	BF	BF	FD
6	2E. 2F.	8F	8F	FF	FF	FF	FF	AF
7	36. 37.	FB	FA	BB	BA	BF	BE	AD
8	3E. 3F.	C7	C7	FF	FF	C3	C3	D3
9	31. 39.	BF	BF	C7	C7	FF	FF	FF

## Joystick-Programmierung

Stephan OEPEN

Durch den leicht zugänglichen User-Port und dessen einfache Abfragung ist der CBM/PET bestens zur Joystick-Programmierung geeignet. Einen Joystick kann man in den meisten Computershops kaufen oder wenn es sein muss auch selber bauen. Dieser Artikel befasst sich aber in erster Linie mit der Programmierung des Joysticks, soll jedoch auch einige Anregungen zum Selbstbau geben.

Es gibt zwei Arten von Joysticks, analoge und digitale. Die analogen arbeiten über zwei Potentiometer, die je nach Joystickbewegung in Richtung der x- oder y-Achse verschoben werden. Die entstehenden analogen Werte des Joysticks müssen dann aber erst durch einen A/D-Wandler umgeformt werden, bevor sie vom CBM/PET verarbeitet werden können. Ein A/D-Wandler dürfte für die meisten Hobbyisten wohl zu teuer sein.

Als Alternative bietet sich der digitale Joystick an. Er besteht aus 4, 8 oder mehr Richtungsschaltern, die je nach Joystickbewegung gegen Masse kurzgeschlossen werden.

Wenn man sich den User-Port einmal genauer betrachtet, wird man feststellen, dass er aus zwei Reihen mit je 12 Kontakten besteht. Für unsere Experimente sind nur die unteren 12 von Nutzen. Es ist auf jeden Fall zu vermeiden, dass einer der oberen Kontakte mit einem anderen oberen oder einem der unteren Kontakte kurzgeschlossen wird.

Für die weiteren Betrachtungen, sollten Sie sich nun einen preiswerten Joystick besorgen. Er braucht nicht fertig, mit Stecker, montiert zu sein, da wir das Anschliessen selber machen werden.

Ist ihr Joystick jedoch bereits komplett, können sie den nächsten Abschnitt überspringen und sich direkt der Programmierung zuwenden.

Wie oben bereits erwähnt, arbeitet der digitale Joystick über

mehrere Richtungsschalter. Um das mehradrige Kabel an den User-Port anzuschliessen, benötigt man einen passenden Stecker, wie den CINCH 251-12-90-160. An ihm ist die untere Steckerleiste mit A-N, ausgenommen C und I, gekennzeichnet.

A und N sind mit Masse digital (GND) verbunden. B und M sind sogenannte "Handshake-Leitungen" (CA1, CB2), die für den Quittungsbetrieb mit Peripheriegeräten benutzt werden. Sie sind für den Joystickbetrieb unerheblich.

Uebrig bleiben noch die Leitungen C, D, E, F, H, J, K und L. (PA0, PA1, ... PA7). Das sind die I/O Leitungen, die wir benutzen wollen.

Als erstes ist die Masseleitung (das Kabel, mit dem die anderen, bei Bewegung des Joysticks verbunden werden) zu finden. Dies geht am besten mit einem Ohm-Meter. Diesen Masseanschluss (er ist meistens schwarz) wird mit Pin A des Steckers (Masse) verbunden.

Als nächstes ist festzustellen (wieder mit dem Ohm-Meter), ob der Joystick bei Bewegung in eine der Hauptrichtungen (N, S, O und W) je einen oder zwei Schalter auf Masse legt.

Die gängigen Typen verwenden für die 4 Hauptrichtungen je einen, und für die Zwischenrichtungen zwei Kontakte. Aus diesem Grunde sind die unten aufgeführten Programmbeispiele auch für diese Typen zugeschnitten. Sie können jedoch ohne grossen Aufwand auf andere Typen umgestellt werden.

Die Zustände des User-Ports werden dezimal im Register (59471) gespeichert. Dabei entspricht Bit 0 des Registers dem Zustand an PA0 (Anschluss C des Steckers) Bit 1 PA1 usw.

Steht in einem Bit des Registers eine 1, so bedeutet das, dass der entsprechende Anschluss des User-Ports offen gegen Masse ist, also auf Output steht.

Enthält die entsprechende Speicherstelle dagegen eine 0, so ist der zugehörige Anschluss des User-Ports mit Masse kurzgeschlossen, steht also auf Input.

Wenn nun alle I/O Ports offen sind, müsste das Register (59471) 255 enthalten, denn binär 1 1 1 1 1 1 1 1 ist gleich dezimal 255. Probieren wir es doch einfach einmal aus. Ist nichts an den User-Port angeschlossen, müsste der Befehl Print PEEK (59471) 255 ergeben. Sie werden sehen, dass dies stimmt.

Würde nun PA0 auf Masse gelegt ist, stände in Bit 0 des Registers eine logische 0, so dass von den 255 -1 abgezogen würden. So kann man folgende Zahlenkombinationen erreichen:

255 - 1 = 254  
255 - 2 = 253  
255 - 4 = 251  
255 - 8 = 247  
255 - 16 = 239  
255 - 32 = 223  
255 - 64 = 191  
255 - 128 = 127

Die meisten Joysticks arbeiten mit diesen Werten für die Hauptrichtungen:

N -253 ; binär 1 1 1 1 1 1 0 1  
S -251 ; binär 1 1 1 1 1 0 1 1  
W -247 ; binär 1 1 1 1 0 1 1 1  
O -239 ; binär 1 1 1 0 1 1 1 1

Wenn nun aber zwei oder mehr Anschlüsse gleichzeitig auf Masse gelegt werden, so addieren sich die Subtrahenden und es entstehen Zwischenwerte. Wiederum die Durchschnittswerte:

N/W -245 ; binär 1 1 1 1 0 1 0 1  
 S/W -243 ; binär 1 1 1 1 0 0 1 1  
 N/O -237 ; binär 1 1 1 0 1 1 0 1  
 S/O -235 ; binär 1 1 1 0 1 0 1 1

Ist an dem Joystick auch noch ein Schussknopf vorhanden, so wird meistens, bei dessen Betätigung, 1 subtrahiert.

Bitte schliessen Sie Ihren Joystick jetzt nach folgendem Schema an!

N	MASSE DIGITAL	NICHTS
M	CB2 (Handshake)	NICHTS
L	PA7 I/O PORT	NICHTS
K	PA6 I/O PORT	NICHTS
J	PA5 I/O PORT	NICHTS
H	PA4 I/O PORT	Osten
F	PA3 I/O PORT	Westen
E	PA2 I/O PORT	Süden
D	PA1 I/O PORT	Norden
C	PA0 I/O PORT	Schuss
B	CA1 (Handshake)	NICHTS
A	MASSE DIGITAL	Masse
		Joystick

Welches Kabel zu welcher Richtung gehört, können Sie ja leicht mit einem Ohm-Messgerät feststellen.

**WICHTIG:** Bevor Sie etwas an den CBM/PET anstecken, sollten Sie das Gerät unbedingt ausschalten! Auch am anzuschliessenden Gerät sollten Sie sich vergewissern, dass kein Kurzschluss entstehen kann!

```
0 PRINT"Q":TAB(9)"USER PORT ZUSTAENDE"
1 REM FORA=1T02000:NEXTA
2 PRINT"DAS PROGRAMM ZEIGT DIE ZUSTAENDE"
3 PRINT " AM PORT"
4 REM DAS PROGRAMM ZEIGT DIE ZUSTAENDE
5 REM IER 8 BITS AM PORT 59471
6 REM
7 PRINT"000" :P=PEEK(59471):PRINTTAB(10);
8 PRINT"DEZIMAL " :P
9 PRINT "BINAER:";
10 PRINTTAB((PAND128)=128);
30 PRINTTAB((PAND64)=64);
40 PRINTTAB((PAND32)=32);
50 PRINTTAB((PAND16)=16);
60 PRINTTAB((PAND8)=8);
70 PRINTTAB((PAND4)=4);
80 PRINTTAB((PAND2)=2);
90 PRINTTAB((PAND1)=1)
100 GOTO5
READY.
```

Beobachten Sie während des Einschaltvorganges den Cursor genau! Sollte er unverhältnismässig schnell blinken, schalten Sie den CBM/ PET sofort wieder aus und versuchen Sie es nocheinmal! Wenn auch beim zweiten Einschalten der Cursor zu schnell ist, so ist dies ein untrügliches Zeichen dafür, dass der User-Port Ihres PET zerstört wurde!!!

Verläuft der Einschaltvorgang normal, so haben Sie Ihren Joystick korrekt angeschlossen und können jetzt mit ihm experimentieren.

Geben Sie als erstes einmal Demoprogramm Nr. 1 ein. Nach dem Einlesen werden auf dem Bildschirm die Zustände des User-Ports von PA0 bis PA7 ausgewiesen. Dabei entspricht die rechte Zahl PA0 (Anschluss C des Steckers), die erste daneben PA1 usw. Eine logische 0 entspricht einem auf Masse gelegten Zustand und eine logische 1 einem offenem Zustand. Mit Hilfe dieses kleinen Programmes können Sie die verschiedenen Wechselzustände am User-Port genau verfolgen.

Einiges zur Funktion des Programmes: Der dezimal ausgewiesene Wert entspricht dem dezimalen Inhalt von Register (59471).

Die Anzeige des binären Wertes wird durch einen kleinen Trick erreicht: Wie Sie sicher wissen, weist der Computer intern jeder Aussage einen Wert (richtig oder falsch) von (-1) und 0 zu. Dabei steht Null, wenn die Aussage falsch ist und (-1) wenn sie richtig ist.

Geben Sie beispielsweise einmal "Print 128=128" ein! Der CBM/PET wird Ihnen mit einer (-1) antworten, weil "128=128" richtig ist. Jetzt werden durch die "and" Anweisung der Inhalt von (50471) nacheinander mit 128, 64, 32, 16, 8, 4, 2 und 1 verknüpft und der CBM/PET aufgefordert den Wahrheitswert der Aussagen auszugeben. Durch die "ABS" Anweisung wird aus (-1) eine

1, und schon kann man die Zustände am User-Port verfolgen.

Wenn Sie sich die binäre Schreibweise einmal auf einem Blatt Papier veranschaulichen, werden Sie es schnell verstanden haben. Gleichzeitig können Sie mit dem Programm überprüfen, ob Ihr Joystick richtig angeschlossen wurde.

Geben Sie jetzt Demoprogramm Nr. 2 ein. Mit ihm können Sie Phantasielbilder, durch Joystickbewegungen zeichnen. Ist gleichzeitig auch noch ein Verstärker an CB2 angeschlossen, können Sie Ihre Bilder auch akustisch verfolgen, denn jeder Richtung ist ein anderer Ton zugeordnet.

Per Programm wird der dezimale Inhalt von (59471) abgefragt und je nach Grösse in Print-Anweisungen umgewandelt. Gleichzeitig werden per Unterprogramm die Töne erzeugt. Beim Drücken des Feuerknopfes wird der Bildschirm geleert und der Ton abgestellt.

```
1 PRINT"Q" MODERN ART SOUND"
3 FORA=1T05000:NEXTA
5 PRINT"Q":POKE59467,16
6 POKE59466,15:POKE59464,0
10 A=PEEK(59471)
20 IFA=254THENB=1
30 IFA=253THENB=2
40 IFA=251THENB=3
50 IFA=247THENB=4
60 IFA=239THENB=5
70 IFA=245THENB=6
80 IFA=255-18THENB=7
90 IFA=235THENB=8
100 IFA=255-12 THENB=9
110 ONBGOSUB 140,150,200,250,300
112 ONB-5GOSUB350,400,450,500
115 GOTO10
140 PRINT"Q":POKE59464,0
145 RETURN
150 PRINT"Q I":POKE59464,25
155 RETURN
160 PRINT"Q I":POKE59464,50
200 PRINT"Q I":POKE59464,50
205 RETURN
250 PRINT"Q I":POKE59464,75
255 RETURN
300 PRINT"Q I":POKE59464,100
305 RETURN
350 PRINT"Q I":POKE59464,125
355 RETURN
400 PRINT"Q I":POKE59464,150
405 RETURN
450 PRINT"Q I":POKE59464,175
455 RETURN
500 PRINT"Q I":POKE59464,200
505 RETURN
READY.
```

# GEWUSST WIE!

## Toolkit, 3. Teil

Mit den Befehlen TRACE, STEP und OFF schliessen wir die Reihe über das Toolkit ab. Die insgesamt zehn zusätzlichen Befehle dieser Programmierhilfe sind eine wesentliche Erleichterung für den CBM/PET-Programmierer, vor allem dann, wenn Fehler in einem Programm behoben werden sollen. Zum Schluss wollen wir noch Probleme besprechen, die mit dem Toolkit auftreten können sowie Hinweise geben, wie man damit fertig wird.

### TRACE

Der Befehl TRACE schaltet einen Modus ein, bei dem die Nummer der gerade ausgeführten Zeile angegeben wird. Die letzten sechs Zeilennummern werden in der oberen rechten Bildschirmecke in einem sogenannten Fenster (schwarz auf weissem Grund) angezeigt. Wenn die Zeilen ausgeführt werden, wandern die entsprechenden Nummern im Fenster hinauf, wobei die letzte Nummer zuunterst angezeigt wird.

Wenn SHIFT gedrückt wird, wird die Programmausführung auf etwa zwei Zeilen pro Sekunde verlangsamt. Sogar ohne die SHIFT-Taste wird die Programmausführung vom Befehl TRACE spürbar gebremst.

Geben Sie als Beispiel das folgende Programm in Ihren PET ein:

```
NEW
10 PRINT" (Bildschirm
   löschen)";
20 X=1
30 PRINT" (Cursor in die
   HOME-Position)";X
40 X=X+1
50 GOTO 30
```

Nun geben Sie den Befehl TRACE ein:

```
TRACE

READY.
RUN
```

Der Bildschirm wird gelöscht und das Fenster erscheint in Reverse-

Darstellung in der rechten oberen Bildschirmecke. Oben links steht eine Zahl die aufwärts zählt. Die Anzeige könnte zum Beispiel so aussehen:

```
50
30
40
50
30
40
```

Wenn Sie jetzt die SHIFT-Taste drücken, wird die Anzeige erheblich langsamer. Nun werden etwa zwei Zeilen pro Sekunde ausgeführt.

Um das Programm anzuhalten, drücken Sie die STOP-Taste. STOP funktioniert auch dann, wenn die SHIFT-Taste gedrückt ist.

Drücken Sie RETURN, wenn Sie das Programm angehalten haben: Die Fehlermeldung ?SYNTAX ERROR wird ausgegeben. Wenn nämlich die Taste RETURN gedrückt wird, schaut der CBM/PET die ganze Zeile an. Als dieses Programm gestoppt wurde, war der Cursor in der fünften Zeile auf dem Bildschirm. Rechts ist das Fenster immer noch da und der Rechner nahm das Kreuz und die Zeilennummer als Ihre Eingabe an.

Wenn Sie ein Programm im TRACE-Modus verlassen, und der Cursor befindet sich in den obersten sechs Zeilen auf dem Bildschirm, drücken Sie entweder die Taste die den Cursor nach unten bewegt oder löschen den Bildschirm um das Fenster zu entfernen.

Versuchen wir noch ein anderes Programm:

```
NEW
10 PRINT"(Cursor in die
HOME-Position)";X: X=X+1:
GOTO 10
RUN
```

Wenn dieses Programm im TRACE-Modus gestartet wird, erscheint nur die Zeile 10 rechts oben im Fenster in der untersten Linie. TRACE füllt das Fenster nicht auf, wenn das Programm sich in einer Schleife auf einer Zeile befindet. Dies ist absichtlich so gemacht, da in vielen Programmen GET-Schleifen enthalten sind.

Beispiel:

```
222 GET A$: IF A$=" " THEN 222
```

Dies ermöglicht Ihnen zu sehen, was vor der Schleife geschah, indem die Zeilennummer 222 nicht wiederholt wird.

### STEP

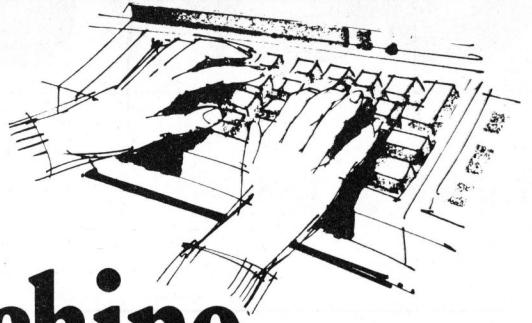
Der Befehl STEP aktiviert den Befehl TRACE, führt eine Zeile aus und hält an. (Die Zeilennummer wird im Fenster in der rechten oberen Bildschirmecke angezeigt.) Um die nächste Zeile ausführen zu lassen, drücken Sie nur die SHIFT-Taste.

Nachdem Sie RUN eingegeben haben, muss SHIFT gedrückt werden, damit die erste Zeile ausgeführt wird. Falls diese Programmzeile eine Schleife enthält, zum Beispiel

```
10 GOTO 10
```

wird sie unaufhörlich ausgeführt. Drücken Sie die STOP-Taste um das Programm zu verlassen.

# Ein Büro ohne Commodore Tisch-Computer ist wie ein Büro ohne Schreibmaschine



Niemand kann sich heute ein Büro ohne Schreibmaschine, Telefon oder Fotokopierer vorstellen. Schon lange sind sie für den Austausch von Informationen unentbehrlich. Doch beim täglichen Aufbereiten dieser Informationen war bisher zeitraubende, nervtötende Routinearbeit gang and gäbe. Künftig faßt der CBM Tisch-Computer sich wiederholende Arbeitsabläufe zu sinnvollen Informations-Schemen zusammen, die

Sie jederzeit auf Knopfdruck abrufen können – ohne spezielle Computer-Kenntnisse. Das gilt für die Buchhaltung ebenso wie für Kalkulationen, Karteien, Planungen, Textverarbeitung, jegliche Berechnungen und was Sie sonst noch fordern. So steigt die Produktivität und die Arbeitsfreude dazu.

All dies verwirklicht Commodore mit eigener Mikroelektronik. Diese bahnbrechende Technologie ermöglicht einen

kompletten Tisch-Computer mit großem 80-Zeichen Bildschirm zu einem Preis, der Sie überraschen wird.

**Niemand kann darauf verzichten.**



**commodore**  
COMPUTER

Commodore AG • Dufourstrasse 9 • 4010 Basel • Telefon 061 23 78 00

**Autorisierte Commodore-Wiederverkäufer  
mit technischem Kundendienst**

**Aarau**  
Dahms Computersysteme · Tel. (064) 2277 66

**Basel**  
BD-Electronic · Tel. (061) 35 36 37  
Geiger-Microcomputer · Tel. (061) 44 13 13  
Leobag Computer AG · Tel. (061) 35 31 14

**Bern**  
Computerland AG · Tel. (031) 24 25 54  
Radio TV Steiner AG · Tel. (031) 55 45 81

**Biel**  
EIM Computer AG · Tel. (032) 23 15 88

**Brugg**  
Megos AG · Tel. (056) 41 34 17

**Fontainemelon**  
Urs Meyer Electronic · Tel. (038) 53 43 43

**Fribourg**  
Sovitrel SA · Tel. (037) 22 78 37

**Genève**  
Centre informatique Gesmarco · Tel. (022) 21 11 75  
Egg-Telsa SA · Tel. (022) 20 06 00  
Gesmarco SA (Thônex) · Tel. (022) 49 88 44  
Ircó Electronic · Tel. (022) 20 33 06  
Radio TV Steiner AG · Tel. (022) 28 52 22

**Gossau**  
Pius Schäfler · Tel. (071) 85 13 87

**Huttwil**  
Rüfenacht AG Automation  
Vibrationstechnik · Tel. (063) 72 11 12

**Interlaken**  
DATATECHNIK · Tel. (036) 22 10 21

**Lausanne**  
Mafioly SA · Tel. (021) 22 00 44  
Schaer informatique · Tel. (021) 23 55 55

**Luzern**  
Dialog Computer Treuhand AG · Tel. (041) 31 45 45  
Helfenstein + Bucher AG · Tel. (041) 23 33 66  
Schweizer Computer Club · Tel. (041) 31 45 45

**Magliaso**  
Marah SA · Tel. (091) 71 14 28

**Mellingen**  
Instant-Soft AG · Tel. (056) 91 20 21

**Niederrohrdorf**  
Nöthiger Electronic · Tel. (056) 96 28 96

**Rüti/ZH**  
Logon AG · Tel. (055) 31 72 30

**Schaffhausen**  
Novotec-Systems · Tel. (053) 4 54 50  
Syntron Electronic · Tel. (053) 5 33 77

**Sion**  
Sphère Corporation · Tel. (027) 22 68 14

**St. Gallen**  
LASYS · Tel. (071) 28 39 05

**Thun**  
HMB electronic · Tel. (033) 22 66 88

**Wettingen**  
Elbatex AG · Tel. (056) 26 56 41

**Winterthur**  
Nowak AG · Tel. (052) 22 08 03

**Wohlen/AG**  
SYSAG System & Services AG · Tel. (057) 6 36 50

**Zürich**  
Fürrer Büro-Computer · Tel. (01) 201 56 10  
Hannes Keller AG · Tel. (01) 69 36 33  
Logon AG · Tel. (01) 62 59 22  
Microspot AG · Tel. (01) 241 20 30  
Erhard Wipf AG · Tel. (01) 221 21 00

# GEWUSST WIE!

Hier ist ein kleines Programm um den STEP-Befehl zu zeigen:

```
NEW
10 PRINT" Erster Satz"
20 PRINT" Zweiter Satz"
30 PRINT" Dritter Satz"
40 GOTO 10
```

STEP

```
READY.
RUN
```

Jetzt erscheint das Fenster in der oberen rechten Bildschirmecke mit der Nr.10 ganz unten. Beachten Sie, dass die Zeile Nr.10 noch nicht ausgeführt wurde.

Drücken Sie ganz kurz auf die SHIFT-Taste. ERSTER SATZ wird erscheinen und das Fenster enthält nun die Zahlen 10 und 20. Jedesmal wenn Sie die SHIFT-Taste drücken, wird eine BASIC-Zeile ausgeführt, und die Zeilennummer des nächsten Befehls der zur Ausführung bereitsteht wird angezeigt.

Wenn Sie die SHIFT-Taste gedrückt halten, werden die Zeilennummern im Fenster wandern, gleich wie beim Befehl TRACE, und auf dem Bildschirm wird folgendes erscheinen:

```
ERSTER SATZ          20
ZWEITER SATZ        30
DRITTER SATZ        40
ERSTER SATZ          10
                    20
```

(usw.)

Wenn Sie STOP drücken wird das Programm angehalten, auch wenn die SHIFT-Taste gedrückt ist. Falls sich der Cursor am unteren Bildschirmrand befindet, wenn die STOP-Taste gedrückt wird, werden nur die letzten drei Zeilen im Fenster angezeigt, da der CBM/PET den Bildschirm verschieben muss, um die Meldungen BREAK IN xxx und READY anzuzeigen.

Der Befehl STEP funktioniert gleich wie der Befehl TRACE, wenn

einer Schleife auf einer Zeile begegnet wird:

```
NEW
10 PRINT"(Cursor in die
HOME-Position)";X: X=X+1:
GOTO 10
RUN
```

Drücken Sie nun die SHIFT-Taste.

Im Fenster wird sich nichts ändern, nur die 10 wird angezeigt. Aber in der linken oberen Bildschirmecke wird eine zunehmende Zahl sichtbar, weil die zunehmende Zahl X angezeigt wird. Wenn eine Schleife auf einer Zeile vorkommt, wird sie ausgeführt, bis die STOP-Taste gedrückt wird. Die Anzeige entspricht der beim Befehl TRACE und füllt nicht das ganze Fenster mit der gleichen Zeilennummer auf.

O F F

Der Befehl OFF schaltet die Befehle TRACE und STEP aus. Nachdem Sie diesen Befehl eingegeben haben, wird ein laufendes Programm nicht mehr im Fenster angezeigt, oder warten, bis Sie die SHIFT-Taste drücken.

Nehmen Sie das Programmbeispiel für den Befehl STEP und starten Sie es mit dem Befehl RUN mit dem eingeschalteten STEP-Modus. Drücken Sie dann die STOP-Taste und geben Sie den Befehl OFF ein.

Nun starten Sie das Programm erneut. Sie werden sehen, dass das Fenster verschwunden ist, und das Programm ganz normal abläuft.

## VORKOMMENDE PROBLEME

Das Toolkit wurde entwickelt, um eine möglichst nützliche Erweiterung zum PET-BASIC zu einem vernünftigen Preis zur Verfügung zu haben. Dabei nützt dieses Maschinenprogramm die im CBM/PET vorhandenen Systemunterprogramme wo immer

möglich aus, und daher ist die Art wie die neuen Befehle ausgeführt werden ganz ähnlich, wie bei den alten BASIC-Befehlen.

In der Regel führt ein Computer nur das aus, was man ihm angibt, aber dies entspricht nicht immer dem, was man tatsächlich wollte. Auch beim Toolkit gibt es Situationen, wo die neuen Befehle eigenartige Resultate hervorbringen und die gewünschte Ausführung nicht so ist, wie man sie erwartet hat.

Die uns bekannten Probleme, die dabei auftreten können, beschreiben wir nachfolgend. Um unliebsame Ueberraschungen zu vermeiden, wenn Sie mitten im Programmieren sind, empfehlen wir Ihnen, diese Fälle einmal auszuprobieren.

## MONTAGE

Beim alten Modell des PET wird diese Schaltung am Speichererweiterungsanschluss auf der rechten Seite des PET angebracht. Die 5 Volt Stromversorgung wird über ein kurzes Kabel und über einen Stecker vom Anschluss des zweiten Kassettengerätes abgenommen. Achten Sie darauf, dass die beiden Verbindungen richtig eingesteckt sind (siehe auch m+k computer 80-5).

Beim neuen PET wird die integrierte Schaltung in einen Erweiterungssockel gesetzt. Achten Sie darauf, dass Sie den richtigen Sockel verwendet haben und dass der Stift 1 der integrierten Schaltung dem Kontakt 1 des Sockels entspricht.

Wenn Sie eine Speichererweiterung besitzen, die den Speichererweiterungsanschluss belegt, gibt es verschiedene Möglichkeiten, die neue elektronische Schaltung anzubringen. Benutzer vom EXPANDAMEM benötigen eine kleine Speicherkarte (bei Ihrem Händler erhältlich),

die in die Erweiterungssockel des EXPANDAMEM passt.

Fragen Sie Ihren Händler, falls Sie Probleme bei der Montage der Schaltung haben.

## SPEICHERBELEGUNG

Das Toolkit benötigt Speicherplatz sowohl um Daten zu speichern als auch für das Maschinenprogramm:

Daten  
\$ 03E0 - \$ 03FF für RAM  
Maschinenprogramm  
\$ D000 - \$ D7FF für ROM

Die Daten werden im oberen Teil des zweiten Kassettenbuffers abgespeichert und mit der Ausnahme des Befehls APPEND kann der Gebrauch des zweiten Kassettenbuffers im BASIC-Programm mit der Programmierhilfe unvorhersehbare Resultate bringen.

Die Startadresse des Maschinenprogramms ist höher als die des Bildschirmspeichers und wird die Arbeit der meisten Benutzer nicht beeinträchtigen, auch wenn sie eine Speichererweiterung besitzen. Wenn Sie jedoch einen COMPUTHINK-Floppy angeschlossen haben, gibt es einige Schwierigkeiten. Die Printplatte, die mit der Version für den alten PET mitgeliefert wird, hat einen speziellen Sockel, der für irgendeine B2716-kompatible Speicherschaltung (ROM oder PROM) benutzt werden kann. Die Startadresse auf diesem Sockel lautet \$9000.

## EINSCHALTEN DER PROGRAMMIERHILFE

Der Befehl SYS 45056 (oder SYS 11\*4096) ist nötig, um das Toolkit einzuschalten und die neuen Befehle zur Verfügung zu haben. Der Startbefehl ist immer dann wieder einzugeben, wenn der CBM/PET ausgeschaltet wurde oder ein System-Reset stattgefunden hatte.

## NUR ALS DIREKT-BEFEHL

Ihre Befehle der Programmierhilfe werden nur ausgeführt, wenn Sie direkt vom Tastenfeld eingegeben wurden. Wenn man einen dieser neuen Befehle in einem BASIC-Programm verwendet, ergibt dies einen ?SYNTAX ERROR.

## NUR EIN BEFEHL AUF EINMAL

Wenn Sie eine Zeile eingeben, die einen Befehl der Programmierhilfe enthält, darf dies der einzige Befehl in der Anweisung sein. Falls Sie versuchen, mehr als einen der neuen Befehle auf eine Zeile zu schreiben oder sie mit BASIC-Befehlen mischen, wird die Fehlermeldung ?SYNTAX ERROR ausgegeben.

Der CBM/PET wird entweder die zusätzlichen Zeichen in einer Zeile einfach übersehen oder in den meisten Fällen die Fehlermeldung ?SYNTAX ERROR anzeigen.

## BEFEHLS-ABKÜRZUNGEN FUNKTIONIEREN

Der CBM/PET akzeptiert Abkürzungen der BASIC-Befehle, wenn der zweite Buchstabe mit der gleichzeitig gedrückten SHIFT-Taste eingegeben wird. Zum Beispiel wird die Eingabe L (SHIFT-O) den LOAD-Befehl ausführen, V (SHIFT-E) wird den Befehl VERIFY durchführen, usw.

Die neuen Befehle können ebenfalls abgekürzt werden:

Au	AUTO
Re	RENUMBER
De	DELETE
Fi	FIND
Ap	APPEND
Du	DUMP
He	HELP
Tr	TRACE
St	STEP
Of	OFF

Selbstverständlich funktionieren andere Teilabkürzungen auch. Zum Beispiel: Re, REn und RENUm werden alle richtig verarbeitet.

## AUTO

Im AUTO-Modus kann es vorkommen, dass Sie mit dem Cursor vorher eingegebene Zeilen abändern möchten und dann zum Modus zurückkehren wollen. Falls die Nummer der Zeile, die abgeändert wurde kleiner ist als die Zeilennummer, die dem Modus gemäss an der Reihe wäre, wird letztere als nächste Zeilennummer angegeben.

Falls eine Zeile mit einer grösseren Zeilennummer abgeändert wurde, wird die Befehlsausführung von AUTO so modifiziert, dass dies berücksichtigt wird. Hier ist ein Beispiel, an dem Sie dies ausprobieren können. Es gibt viele Kombinationen. Versuchen Sie einige, um zu sehen, wie es funktioniert.

```
AUTO 100,10
100 REM ZEILE EINS
110 REM ZEILE ZWEI
120
```

Benutzen Sie jetzt den Cursor, um die Zeilen 100 auf 200 abzuändern und dann drücken Sie die Taste RETURN. Die neue Zeilennummer, die von AUTO ausgegeben wird, heisst 210, obwohl eigentlich 110 an der Reihe wäre.

Wenn Sie den AUTO-Modus verlassen wollen, nachdem Sie Zeilen mit dem Programm abgeändert haben, beachten Sie, dass wenn Sie die Taste RETURN nach einer Zeilennummer drücken, diese Zeile gelöscht wird! Benützen Sie die DEL-Taste (DELETE), um die Zeilennummer zu löschen und drücken Sie dann die Taste RETURN auf der leeren Zeile, wenn die Gefahr besteht, dass Sie eine Programmlinie verlieren würden.

Die Befehle AUTO 100,0 und AUTO 0,0 werden ausgeführt, obwohl sie eigentlich recht nutzlos sind.

Falls die nächste ausgegebene Zeilennummer grösser als 63999 ist, wird die Fehlermeldung ?OUT OF RANGE ERROR ausgegeben und die letzte Zeile, die Sie eingegeben haben, wird verloren gehen. Zum Beispiel:

```
AUTO 50000,10000
50000 REM EINS
60000 REM ZWEI
?OUT OF RANGE ERROR
READY.
LIST
50000 REM EINS
READY.
```

## RENUMBER

Mit dem Befehl RENUMBER werden alle Programmzeilen im Speicher neu nummeriert. Wenn eine Zeilennummer unbestimmt ist, d.h. wenn ein Befehl GOTO xxx vorhanden ist, und das Programm hat keine Zeile xxx wird diese Nummer zu 63999 geändert. Benützen Sie den Befehl FIND 63999 um solche Sprünge zu nicht vorhandenen Zeilen zu finden.

Falls die grösste Zeilennummer nach dem Numerieren grösser als 63999 werden würde, wird die Fehlermeldung ?OUT OF RANGE ERROR ausgegeben und das Programm wird nicht neu nummeriert.

Die Befehle RENUMBER 200,0 und RENUMBER 0,0 werden ausgeführt. Das Programm wird dabei unbrauchbar, da alle Zeilen mit der gleichen Nummer versehen werden!

Wenn kleine Zeilennummern zu grossen umgewandelt werden und umgekehrt (z.B. 25 zu 1000) muss das Programm im Speicher verschoben und gewisse Pointer geändert werden. Lange Programme, die vollständig mit neuen Zeilennummern versehen werden, benötigen dazu einige Zeit. Bei Programmen die sehr lang sind (d.h. weniger als 100 Byte freier Speicherplatz), ist es möglich, dass die Fehlermeldung ?OUT OF MEMORY ERROR ausgegeben wird und dass die Zeilen nicht richtig

neu nummeriert werden. Um dies zu vermeiden, versucht man den Befehl RENUMBER 1,1, um die Zeilennummern kurz zu halten. Trotzdem wird es wahrscheinlich nicht genug Platz für die Variablen haben, wenn Sie das Programm laufen lassen!

Adressen, die in REM-Anweisungen, bzw. in Anführungszeichen eingeschlossen sind, werden durch den Befehl RENUMBER nicht geändert. Beachten Sie dies, wenn Sie einen Befehl folgender Art in Ihrem Programm haben:

```
REM GOTO 500 - VERZWEIGUNG
FUER 16K PET
```

Programme, die sich selber abändern, indem Programmzeilen auf den Bildschirm ausgedruckt und der Tastenbuffer mit RETURN-Codes gefüllt wird, werden möglicherweise nicht mehr richtig funktionieren, nachdem RENUMBER durchgeführt wurde.

Wenn Sie ein Programm mit aus der Reihe tanzenden Zeilennummern mit dem Befehl APPEND angehängt haben, kann der Befehl RENUMBER benützt werden, um den neuen Programmteil arbeitsfähig zu machen. Beachten Sie, dass irgendwelche Verzweigungen (GOTO, IF-THEN, usw.) nun zu falschen Zeilennummern zeigen!

## DELETE

Der Befehl DELETE ohne eine Zeilennummer oder DELETE -, ergibt die Fehlermeldung ?SYNTAX ERROR. Dies ist beabsichtigt um den Verlust Ihres Programms zu vermeiden, wenn Sie irrtümlich DELETE eingegeben haben. (Benutzen Sie den Befehl NEW um ein Programm zu löschen.)

Falls der Befehl DELETE in einem Bereich von einer grösseren Zeilennummer zu einer kleineren benutzt wird, ergibt dies die Fehlermeldung ?SYNTAX ERROR, falls in diesem

Bereich eine Zeile existiert. Ist keine Zeile da, wird nichts verändert.

## FIND

Der Zeilennummer-Bereich für den Befehl FIND ist gleich wie beim Befehl LIST. FIND ohne Zeilennummer untersucht das ganze Programm. Falls zuerst eine grössere und dann eine kleinere Zeilennummer angegeben wird, reagiert der Befehl FIND wie der Befehl DELETE.

Wenn der Suchbegriff nicht in Anführungszeichen steht, wird er teilweise in Schlüsselzeichen umgesetzt und die Suche erstreckt sich nur auf die Teile im BASIC-Programm, die nicht in Anführungszeichen stehen. Steht der Suchbegriff in Anführungszeichen, werden nur die Teile im BASIC-Programm abgesehen, die in Anführungszeichen stehen.

Beachten Sie, dass der Befehl FIND ohne einen Suchbegriff in Anführungszeichen nach passendem Text sucht, der teilweise in Schlüsselzeichen umgewandelt wurde. Da ja der ganze BASIC-Programmtext in Schlüsselzeichen umgewandelt worden ist, wird der Befehl FIND PRINT alle Printanweisungen finden. Zu beachten ist, dass Texte in REM-Anweisungen nicht in Schlüsselzeichen umgewandelt werden. Der Befehl PRINT innerhalb einer REM-Anweisung wird als Folge von fünf Buchstaben gespeichert, anstatt eines einzelnen Schlüsselzeichens. Die Folge davon ist, dass der Befehl FIND PRINT keine Anweisung wie REM PRINT finden wird, weil er nur nach dem Schlüsselzeichen für PRINT Ausschau hält. Ein Ausweg aus dieser Situation ist nur für Teil-Schlüsselworte in REM-Anweisungen zu suchen. Zum Beispiel: FIND RINT.

Beim Befehl FIND PRINT "BAZ" wird der Ausdruck PRINT in ein Schlüsselzeichen umgewandelt und der Ausdruck "BAZ" wird überse-

hen. Aehnlich wird mit dem Befehl FIND "PRINT" BAZ verfahren, indem die Suche nur mit "PRINT" durchgeführt wird. Bei FIND wird nur der Teil der Zeile hinter der Zeilennummer abgesucht, und die Zeilennummer selbst wird nicht gefunden.

## APPEND

Der Befehl APPEND arbeitet nur mit den Kassettenstationen 1 und 2. Geräte am IEEE-Bus werden nicht erkannt. Auch für Programme auf Disketten funktioniert dieser Befehl nicht.

Falls der Programmteil, der angehängt werden soll, den zu Verfügung stehenden Speicherbereich überschreitet, wird die Befehlsausführung abgebrochen. Die Grösse des angehängten Programms ist bekanntlich auf dem Bandvorspann gespeichert, sodass teilweise angehängte Programme nicht sichtbar sind, und sich auch sonst keine Änderungen ergeben.

Die Aufnahmen werden gleicherweise nach Programmnamen abgesucht, wie bei LOAD.

## DUMP

Der Befehl DUMP wird die Variablen immer dann anzeigen, wenn sie dem BASIC noch zur Verfügung stehen. Beachten Sie, dass das Abändern irgendeiner Zeile im Programm alle Variablen löscht.

## HELP

HELP ist sofort einzugeben, nachdem das Programm angehalten wurde, da die zur Befehlsausführung nötigen Daten sonst gelöscht werden. Die Marke befindet sich oft ein Zeichen vor dem fehlerhaften Teil.

Wenn ein Programm mit der Stopptaste angehalten wird, bzw. wenn es

selbst anhält, wird mit dem Befehl HELP die letzte ausgeführte Zeile angezeigt.

Die Marke wird den ganzen fehlerhaften Befehl schwarz auf weiss darstellen (reversed).

In einigen Fällen steht die Marke nicht dort, wo die Fehlerquelle ist. Die Marke wird jeweils dorthin gesetzt, wo der CBM/PET gerade arbeitet, wenn er den Fehler bemerkt. Dies kommt oft in arithmetischen Ausdrücken vor, oder mit READ, INPUT und GET.

## TRACE UND STEP

Wenn sich der Cursor in einer der obersten Bildschirmzeilen befindet, nachdem TRACE oder STEP eingegeben wurde, enthalten alle Direktbefehle die Zeichen im "Fenster", wenn Sie RETURN drücken, was nicht sehr gut funktionieren wird. Entweder löschen Sie den Bildschirm, oder bewegen den Cursor entsprechend nach unten, bevor Sie Befehle eingeben. Mit Shift-RETURN können Sie den Cursor bequem eine Zeile weiterschieben.

## OFF

Beim Befehl OFF sind bisher keine Probleme aufgetaucht.

## SCHLUSSBETRACHTUNG

Das Toolkit bietet vor allem jenen CBM/PET-Benutzern, die viel programmieren, eine echte Hilfe auf die sie, einmal mit dieser Programmierhilfe gearbeitet, kaum mehr verzichten wollen. Dabei ist zum richtigen Funktionieren entscheidend, dass beispielsweise das BASIC-Programm intakt ist, die Variablenpointer an den richtigen Ort zeigen, die Zeilennummern stimmen, usw. Unter Beachtung dieser an sich normalen Voraussetzungen werden Sie mit dem Toolkit Ihre helle Freude haben.

## Schnelle Grafik auf Sorcerer

Patrick HÄNER

Es gibt bereits verschiedene Programme, die dem Sorcerer ein Pseudo-Plotten ermöglichen. Leider haben sie einen Nachteil: Da sie in Basic geschrieben sind, sind sie zu langsam. Dieser Mangel wird durch das hier vorgestellte Programm als Beispiel für ein BASIC/Assembler-Hybrid behoben.

Die Bedienung ist sehr einfach und trotzdem verhältnismässig komfortabel: Nach einer einmaligen Initialisierung und Wahl des Plot-Modus kann der Benutzer beginnen, auf einer 511\*239-Matrix zu Plotten. Die zwei Plot-Modi sind:

- Plot 1 (beliebig lange, punktierte Kurve)
- Plot 2 (beschränkte, ausgezogene Kurve)

Die Betriebsart wird durch "POKE 0,0" für Plot 1, bzw. "POKE 0,n" (n≠0) für Plot 2 gewählt. Bei Beginn des Plot-Vorganges oder bei Aenderung des Plot-Modus muss das betreffende Flag durch "POKE n,0" (n: = Nummer des Plot-Modus) rückgesetzt werden. Nach Berechnung der Koordinaten X und Y kann mit A=USR (A) geplottet werden.

Hier ein Beispiel:

```
GOSUB 9000:POKE 0,1:POKE 2,0
(Initialisierung; Plotmodus:
Plot 2)
FOR X=0 TO 511
Y=100*SIN(X/511*2*3,1415)
+120 (Plotten einer Sinuskurve)
NEXT
```

Das Programm ist ein Hybrid zwischen BASIC- und Assembler-Sub-routinen:

- in Basic: Initialisierung
- im Assembler: eigentliches Unterprogramm bestehend aus:
  - Steuerung
  - Plot 1

- Erzeugung der 64 Einzelpunkte
- Plot 2
- Verarbeiten eines Koordinaten-paares
- Finden und Umwandeln einer Basic-Variablen

Für die Platzierung von Assembler-unterprogrammen unter Basic-Kontrolle gibt es grundsätzlich zwei Möglichkeiten:

```
9000 POKE 260,22:POKE 261,0
9010 BE=256*PEEK(440)+PEEK(439):N=0
9020 IF PEEK(BE-N)+PEEK(BE-N-1)=0 THEN 9040
9030 N=N+1:GOTO 9020
9040 FOR M=0 TO 141
9050 POKE M,PEEK(BE-497-N+M)
9060 NEXT
9070 NN=N-256*PEEK(5)-PEEK(4)
9080 POKE 5,INT(N/256):POKE 4,N-256*PEEK(5)
9090 NN=256*PEEK(7)+PEEK(6)+NN
9100 POKE 7,INT(NN/256):POKE 6,NN-256*PEEK(7)
9110 RETURN
```

```
0 GOSUB 9000:RD=RND(4)
10 PRINTCHR$(12):PRINT"Bin am Berechnen der Funktionswerte."
20 PRINT"Bitte warten Sie einen Moment."
30 DIM Y(511)
40 FOR X=0 TO 511
50 XX=X/511*31.415
60 Y(X)=100*SIN(XX)*EXP(-.1*XX)+120
70 NEXT
80 PRINTCHR$(12):POKE 1,0:POKE 2,0
90 POKE 0,INT(RND(RD)+.5)
100 FOR X=0 TO 511:Y=Y(X):A=USR(A):NEXT
110 GOTO 80
9000 POKE 260,22:POKE 261,0
9010 BE=256*PEEK(440)+PEEK(439):N=0
9020 IF PEEK(BE-N)+PEEK(BE-N-1)=0 THEN 9040
9030 N=N+1:GOTO 9020
9040 FOR M=0 TO 141
9050 POKE M,PEEK(BE-497-N+M)
9060 NEXT
9070 NN=N-256*PEEK(5)-PEEK(4)
9080 POKE 5,INT(N/256):POKE 4,N-256*PEEK(5)
9090 NN=256*PEEK(7)+PEEK(6)+NN
9100 POKE 7,INT(NN/256):POKE 6,NN-256*PEEK(7)
9110 RETURN
```

- Das Assemblerprogramm wird als Basic "getarnt" in den Basic-Programmbereich gesetzt.

- Das Programm wird in Basic mit "DATA" gespeichert und dann mit "READ" und "POKE" in einen freien RAM-Bereich transportiert.

Als Nachteile der zweiten Methode sind grosser Bedarf an Speicherplatz, Gefahr des Ueberschreibens und mangelnde Eleganz zu nennen. Man wird aus diesen Grund in der Regel die Basic-"Tarnung" wählen. Damit der Computer die Assembler-routinen auch als Basic behandelt, sollte folgendes beachtet werden:

- Das Ende des 'echten' Basic ist durch drei Nullen zu markieren.
- Die Endadresse des Hybrids wird in 1B7/1B8 gespeichert.
- Das Assemblerprogramm hat nach dem Basic-Teil zu stehen (um die Handhabung zu erleichtern).

Um die Funktion des Assemblerprogrammes zu gewährleisten, muss ein Steuerungsprogramm eingesetzt werden, das die bei Änderungen im Basic-Programm auftretenden Adressänderungen korrigiert. Diese Steuerung funktioniert etwa folgendermassen: Zuerst wird das neue Programmende (in 1B7/1B8) mit dem alten (in 6/7) verglichen. Sind sie gleich, d.h. fand im Basic-Programm keine Änderung statt, wird direkt nach Plot 1 bzw. Plot 2 gesprungen. Andernfalls wird die Differenz zwischen dem alten und neuen Programmende berechnet.

Nun werden die zu verändernden Adressen im Assemblerprogramm und deren Inhalte um die Differenz erhöht. Auch die übrigen Parameter (z.B. altes Ende) werden auf den neusten Stand gebracht und gespeichert.

Ist das geschehen, wird wie normal (mit dem Aufruf einer Ploutroutine) fortgefahren. Die Steuerung und das Datenfeld werden am Anfang von der Initialisierungsroutine (in Basic) nach Page 0 transportiert, so dass Daten und Steuerung nicht mehr im Basic-Programmbereich liegen und folglich keine Adressänderungen mehr erleiden (wie könnte man sonst - ausser in langsamen Basic - bei Adressänderung der Steuerung deren Anfang finden?!).

Da in der Initialisierung (insbesondere bei Verschiebung von Datenfeld und Steuerung nach Page 0) die Kenntnis der Anfangsadresse des Datenfelds benötigt wird, und man

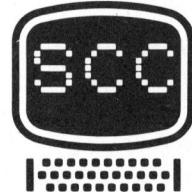
diese durch Subtraktion eines bestimmten Wertes von der Endadresse erhält, muss man folglich die Endadresse kennen. Die "Endadresse" in 1B7/1B8 darf man aber nicht benutzen, da sie nach jedem "CLOAD"- "CSAVE" incrementiert wird. Daher ist man gezwungen, das (mit zwei Nullen markierte) Ende selber zu suchen, was aber keinerlei Schwierigkeiten bietet.

Mit der Besprechung dieser allgemeingültigen Moduls wollen wir es bewenden lassen, da die übrigen Programmteile weniger interessant und anspruchsvoll sind.

Das Ziel dieser Arbeit war, andere Wege zur Darstellung von Assemblersubroutinen aufzuzeigen und unseren Lesern eine Anregung zu eigenen Ueberlegungen (und vielleicht auch Taten), zu vermitteln.

ADDR	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
02D0:	6E	23	4E	4E	AD	4E	A6	32	35	36	A7	BE	28	35	29	A6
02E0:	BE	28	34	29	00	05	03	78	23	96	20	35	2C	B0	28	4E
02F0:	A8	32	35	36	29	3A	96	20	34	2C	4E	A6	32	25	36	A7
0300:	BE	28	35	29	00	1D	03	82	23	4E	4E	AD	32	35	36	A7
0310:	BE	28	37	29	A5	BE	28	36	29	A5	4E	4E	00	3F	03	8C
0320:	23	96	20	37	2C	B0	28	4E	4E	A8	32	35	36	29	3A	96
0330:	20	36	2C	4E	4E	A6	32	35	36	A7	BE	28	37	29	00	45
0340:	03	96	23	8E	00	00	00	00	00	00	00	01	00	D0	04	6C
0350:	03	85	03	72	03	7B	03	A5	03	DE	03	E6	03	AF	2A	B7
0360:	01	ED	4B	06	00	ED	42	B4	B5	28	58	ED	5B	B7	01	ED
0370:	53	06	00	E5	C1	2A	08	00	09	22	08	00	2A	0A	00	09
0380:	22	0A	00	3E	16	DD	21	0C	00	DD	6E	00	DD	66	01	09
0390:	DD	75	00	DD	74	01	E5	5E	23	56	D5	E1	09	E5	D1	E1
03A0:	73	23	72	DD	23	DD	23	DD	BD	20	DE	B7	ED	5B	04	00
03B0:	2A	B7	01	ED	52	11	ED	01	ED	52	E5	D1	21	04	00	01
03C0:	13	00	ED	B0	3A	00	00	FE	00	20	05	2A	08	00	18	03

ADDR	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03D0:	2A	0A	00	E9	00	3A	01	00	FE	01	C4	97	04	01	00	58
03E0:	11	00	59	CD	DC	03	FE	FF	C0	3E	C0	80	77	C9	3A	02
03F0:	00	FE	01	28	12	21	00	FC	AF	77	23	BC	20	FB	3C	32
0400:	02	00	3E	7F	32	03	00	01	00	58	11	00	59	CD	DC	03
0410:	FE	FF	C0	3A	03	00	FE	FF	C8	96	CB	7F	20	05	7E	CB
0420:	7F	20	07	3A	03	00	3C	32	03	00	77	D6	80	6F	26	00
0430:	06	03	CB	25	CB	14	10	FA	3E	07	93	85	6F	11	00	FC
0440:	19	7E	B1	77	C9	D5	CD	1C	04	C1	FE	FF	C0	D5	CD	1C
0450:	04	C1	FE	FF	C0	3E	1D	BA	F8	92	C5	6F	26	00	06	06
0460:	CB	25	CB	14	10	FA	C1	51	48	06	00	09	01	80	F0	09
0470:	42	04	0E	01	CB	09	10	FC	7A	43	04	C6	F8	C6	08	10
0480:	FC	47	3E	FF	C9	ED	5B	B7	01	1B	1B	1B	1B	1B	1B	13
0490:	13	13	13	13	13	AF	2A	B9	01	ED	52	B4	B5	C8	AF	DD
04A0:	21	00	00	DD	19	DD	6E	00	DD	66	01	ED	42	B4	B5	20
04B0:	DE	DD	7E	05	FE	00	20	08	01	00	00	11	00	00	18	3D
04C0:	3E	B9	DD	BE	05	FB	DD	7E	04	D6	80	F0	67	DD	6E	03



## Was macht den Schweizer Computer Club so attraktiv?

- Vorteilhafte Clubangebote
- Günstige Programmierkurse
- Grösste Auswahl an Geräten, Programmen und Büchern
- Kompetente Fachberatung im SCC Computer Shop
- Schnelle Serviceleistung
- Gut ausgebauter eigener technischer Dienst
- Eigene Fachzeitschriften  
**Mikro- und Kleincomputer**  
**CBM/PET NEWS**  
**ComputerJournal**

Urteilen Sie selbst. Werden auch Sie Mitglied. Verlangen Sie unverbindlich Erstinformationen vom

**Schweizer Computer Club**  
Seeburgstrasse 18, CH-6002 Luzern  
Telefon 041 - 31 45 45

## Pascal-Spielprogramm

Leopold ASBÖCK

Immer mehr setzt sich PASCAL als Programmiersprache auf Computern aller Grössen durch, vor allem auch bei Kleincomputer. Selbst begeisterte BASIC-Programmierer werden nach kurzer Zeit nicht nur die höhere Geschwindigkeit, sondern vor allem die übersichtliche Programmgestaltung in PASCAL schätzen. "Seriöse" Programmierer rümpfen über Spielprogramme wohl die Nase, für den Einsteiger dagegen sind solche Programme amüsante Anschauungsbeispiele, aus denen er vielleicht mehr lernt als aus einigen trockenen Kapiteln eines Lehrbuches.

Das vorliegende Programm "BALLON" in PASCAL/M abgefasst, ist ein Spiel, das die Reaktionsfähigkeit des Spielers testet. Es läuft auf SUPERBRAIN, bei Aenderung einiger computerspezifischer Adressen auch auf anderen Kleincomputern mit CP/M-Betriebssystem.

PASCAL/M bietet dem Anfänger

einen preisgünstigen Einstieg in diese effiziente Programmiersprache. Der auf Diskette gelieferte Compiler setzt ein PASCAL-Programm in einen sehr komprimierten p-Code (Pseudocode) um, der vom p-Code-Interpreter mit grosser Geschwindigkeit ausgeführt wird. Das Manual beschreibt sämtliche Funktionen und Sonderfunktionen von PASCAL/M.

Zum Erlernen dieser Programmiersprache ist jedoch zusätzlich grundlegende Literatur über PASCAL notwendig. Zahlreiche gute Bücher in deutscher oder englischer Sprache ermöglichen ein rasches Vorwärtkommen.

Da PASCAL/M über keinen eigenen Editor verfügt, empfiehlt sich die Kenntnis des CP/M-Editors (im CP/M-Betriebssystem enthalten) oder eines Textverarbeitungssystems (z.B. Wordstar), das wesentlich komfortabler und bildschirmorientiert ist.

### DAS PROGRAMM

In der Prozedur "SPIELREGEL" wird

```
-- B A L L O N - Computerspiel in PASCAL/M auf SUPERBRAIN
-- *****
--
-- Leopold Asböck 010181
(*$D-*)

program BALLON (input,output);
-----
const BAELLE = 50;

var SPIEL,TEMPO,TREFFER: integer;
    M,MAXPUNKT,NEUPUNKT: integer;
    NOCHMALS: char;
    GETROFFEN: boolean;
    JANEIN: set of char;

procedure SPIELREGEL;
-----
var T: char;

begin
writeln;
writeln ('* * * B A L L O N * * *');
writeln;
writeln;
write ('Auf 10 Schiesständen erscheinen nacheinander ');
writeln (BAELLE, ' Luftballons,');
write ('die Sie durch Drücken der zugehörigen');
writeln (' Tasten 1,2,3,4,5,6,7,8,9,0');
writeln ('abschiessen sollen.');
```

```
procedure LEER;
-----
begin
write (chr(12));
end;

procedure SCHUSS;
-----
begin
write (chr(7));
end;

procedure ANZEIGE (X,Y,Z: integer);
-----
begin
gotoxy (X,Y);
write (Z:3);
end;

procedure BUDE;
-----
var I,J: integer;

begin
gotoxy (3,3);
writeln ('Spiel Nr:':15,'Treffer:':20,'Maximum:':30);

for I:= 1 to 5 do
begin
gotoxy (10,14+I);
for J:= 1 to 10 do write ('! ');
writeln ('!');
end;

for I:= 1 to 2 do
begin
gotoxy (10,12+4*I);
for J:= 1 to 10 do write ('+-----');
writeln ('+');
end;

for I:= 1 to 10 do
begin
gotoxy (13+6*(I-1),18);
if I=10 then write ('0') else write (I);
end;
end;
```

das Spiel erklärt: Ueber zehn Schiesständen erscheinen in zufälliger Reihenfolge fünfzig Luftballons, steigen in verschiedene Höhen auf und sinken wieder ab.

Der Spieler muss, um einen Ballon abzuschiesen, jeweils die zugehörige Taste 1,2,3,...8,9,0 drücken. Wurde ein Ballon getroffen, so wird dies akustisch und optisch signalisiert, der Ballon platzt und verschwindet.

Die Anzahl der Treffer und Spiele sowie der Maximalstand werden angezeigt.

Die Geschwindigkeit ist in drei

Stufen wählbar, und zwar für Anfänger, Amateur und Profi. Von Spiel zu Spiel wird die Geschwindigkeit unmerklich angehoben.

Das Programm ist selbstdokumentierend und auf Grund der Strukturierung leicht durchschaubar. Reservierte Wörter sind in Kleinschrift, Variablen- und Prozedurnamen in Grossschrift wiedergegeben.

Computerspezifisch sind die Adressen in den Befehlen "inport..." und "outport...", die unter Umgehung eines Keyboard-Files die Tastatur des SUPERBRAINS direkt abfragen.

Falls das Programm unter dem Namen "BALLON.PAS" mit Hilfe des Editors fehlerfrei eingegeben und auf Diskette gespeichert wurde, wird es mit

A PRUN PASCAL BALLON

kompiliert (Kompilierzeit etwa 55 Sekunden).

Dadurch entsteht des p-Code-File "BALLON.PCO" (Länge ca. 2,2 KByte), das mit

A PRUN BALLON

ausgeführt wird.

Gut Schuss und viel Vergnügen!

```

procedure PAUSE (K: integer);
-----
var I,J: integer;

begin
  for I:= 1 to K do J:=1;
end;

procedure LOESCHEN (X,Y,Z: integer);
-----
var I: integer;

begin
  gotoxy (X,Y);
  for I:= 1 to Z do write (' ');
end;

procedure START;
-----
begin
  gotoxy (10,11);
  write ('A c h t u n g - es geht los ! ');
  PAUSE (5000);
  LOESCHEN (10,11,60);
end;

procedure PENG (I,J: integer);
-----
begin
  gotoxy (11+6*(I-1),16-J);
  if TREFFER mod 3 = 2 then write ('PUFF!')
  else write ('PENG!');
  PAUSE (600);
  gotoxy (11+6*(I-1),16-J);
  write (' ');
end;

procedure LUFTBALLON (I,J: integer);
-----
var L,BYTE: integer;

begin
  GETROFFEN:= false;
  outport (107,15);
  gotoxy (13+6*(I-1),16-J);
  PAUSE (30+40*TEMPO-2*SPIEL);
  inport (105.BYTE);
  if BYTE mod 2 = 1 then inport (80,BYTE)
  else BYTE:= 999;
  outport (107,14);
  outport (107,15);
  if I=10 then L:=0 else L:=I;
  if BYTE-48 = L then
    begin
      SCHUSS;
      PENG (I,J);
      TREFFER:=TREFFER+1;
      ANZEIGE(38,3,TREFFER);
      GETROFFEN:= true;
    end;
end;

```

```

procedure BALLAUFAB;
-----
var I,J,JMAX,BYTE: integer;

begin
  I:= trunc(10*random(0))+1;
  JMAX:= trunc(10*random(0))+1;
  if JMAX < 4 then JMAX:=4;

  for J:= 1 to JMAX do
    begin
      LUFTBALLON (I,J);
      if GETROFFEN then exit(BALLAUFAB);
    end;

  for J:= JMAX-1 downto 1 do
    begin
      LUFTBALLON (I,J);
      if GETROFFEN then exit(BALLAUFAB);
    end;
end;

procedure FOLGESPIEL;
-----
begin
  gotoxy (10,7);
  write ('E N D E - Sie haben ', TREFFER);
  writeln (' Punkte erreicht!');
  PAUSE (1000);
  gotoxy (10,9);
  write ('Wagen Sie noch ein Spiel (J/N) ? ');
  outport (107,14);
  outport (107,15);
  repeat
    read (NOCHMALS);
  until NOCHMALS in JANEIN;
end;

-- Hauptprogramm BALLON !

begin
  LEER;
  SPIELREGEL;
  LEER;
  LEER;
  SPIEL:= 0;
  MAXPUNKT:= 0;
  NEUPUNKT:= 0;
  JANEIN:= ['J','j','N','n'];
  BUDE;
  repeat
    SPIEL:= SPIEL+1;
    ANZEIGE (18,3,SPIEL);
    TREFFER:= 0;
    ANZEIGE (38,3,TREFFER);
    if MAXPUNKT < NEUPUNKT then MAXPUNKT:=NEUPUNKT;
    ANZEIGE (69,3,MAXPUNKT);
    outport (107,14);
    START;
    for M:= 1 to BAELLE do BALLAUFAB;
    NEUPUNKT:= TREFFER;
    FOLGESPIEL;
    LOESCHEN (10,7,60);
    LOESCHEN (10,9,60);
  until (NOCHMALS='N') or (NOCHMALS='n');
  LEER;
end.

```

## V24 für Kassettenrekorder

Urs HUNZIKER

Wenn man mit Kleincomputern verschiedener Hersteller arbeitet, stösst man oft auf grosse Hindernisse, falls man Programme und Daten vom Gerät des Herstellers A auf ein Gerät vom Hersteller B übertragen will. Eine Uebertragung der Information über Disketten scheidet meist von vornherein aus, da fast alle Systeme ihre Information anders verschlüsselt auf den Disketten aufzeichnen. Das direkte Zusammenschalten der Geräte bietet bei grösseren Uebertragungsdistanzen ebenfalls Probleme.

Dieser Beitrag beschreibt ein Kassetteninterface zur Speicherung von Information aller Art, wie zum Beispiel Programme, Daten, Bilder, usw. mit einem handelsüblichen Kassettenrecorder. Man muss sich dabei lediglich im klaren sein, dass die vorgeschlagene Schaltung sehr einfach aufgebaut ist und deshalb keine ausgesprochen hohe Datensicherheit zu erwarten ist. Dafür aber ist das Gerät klein und handlich; es lässt sich ohne weiteres in einem Aktenkoffer oder gar in einer Manteltasche transportieren. Wenn das Interface zusammen mit einem Mini-Kassettengerät betrieben wird (z.B. Sony-Walkman 3), entsteht eine sehr kompakte Datenübertragungseinrichtung.

### VORAUSSETZUNGEN ZUM BETRIEB DES GERAETES

- Mikrocomputer mit serieller Schnittstelle
- Uebertragungsgeschwindigkeit zwischen 100 und 1200 Baud
- Möglichkeit Daten mit 100-1200 Baud aufzunehmen und zu verarbeiten. (Kein Handshaking erforderlich).

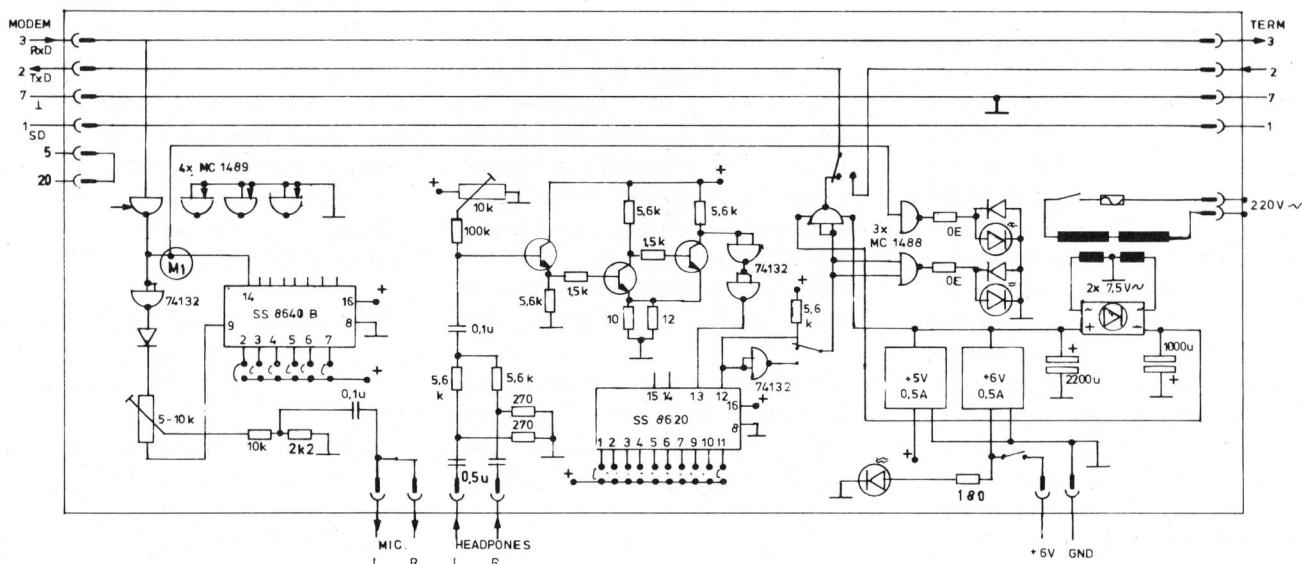
Geräte, die diese Voraussetzungen erfüllen, sind z.B. Superbrain, Apple mit serieller Karte, fast alle Grosscomputer, CBM/PET mit serieller Schnittstelle und der Sorcerer.

### DATEN SCHREIBEN

Die von der seriellen Schnittstelle kommende Information gelangt zuerst über einen RS232 Leitungsempfänger (MC 1489) und wird zu einem TTL-tauglichen Signal aufbereitet. Der nicht benutzte 'control'-Eingang des Empfängers wird offen gelassen.

Das serielle Signal gelangt darauf an den 'HALT'-Eingang eines Rechteckgenerators. Als Generator wird ein IC vom Typ SS 8640-B verwendet. Dieses ist ein programmierbarer Taktgenerator, der eine quarzstabile Ausgangsfrequenz liefert. Dieser IC wird auf eine Frequenz zwischen 3-8 KHz eingestellt, abhängig von der Uebertragungsgeschwindigkeit (der Baudrate) und vom verwendeten Tonbandgerät.

Das Rechtecksignal, das von der seriellen Leitung im Rhythmus der Information ein- und ausgeschaltet wird, gelangt nun auf ein Trimm-



KASSETTEN-INTERFACE

potentiometer, welches einen Spannungsteiler darstellt.

Wenn das Signal an M1 = LO ist, gelangt das invertierte Signal (also HI) über die Diode an den Spannungsteiler. Dieser ist so eingestellt, dass eine Spannung von ca. 1,6 V entsteht.

Wenn das Signal an M1 auf HI wechselt, wird ein LO über die Diode an den Spannungsteiler gegeben. Die Diode ist nun aber in Sperrichtung gepolt und deshalb ist der Spannungsteiler für das jetzt einsetzende Rechtecksignal ein normaler Widerstand.

Dieses Signal wird nun an einem weiteren Spannungsteiler abgeschwächt und über einen Kondensator an das Tonbandgerät abgegeben.

## DATEN LESEN

Das niederfrequente Signal vom Tonbandgerät gelangt über den Eingangsteiler an die Schaltung für die Aufbereitung des NF-Signals. Nach dieser Aufbereitung gelangt das Signal zu den Gattern (74132), welche saubere Flanken erzeugen. Mit dem TTL-gerechten Signal wird nun der Trigger-Eingang eines Monoflops angesteuert. Dies ist wiederum ein IC, diesmal vom Typ SS 6420.

Dieser IC (ebenfalls quartztabil) ist auf eine Zeit eingestellt, die etwas länger als die Periode des Rechteckgenerators ist. Dies hat zur Folge, dass jeder vom Tonband kommende Rechteckimpuls den Monoflop auslöst. Da der Monoflop nachgetriggert werden kann, entsteht aus dem NF-Signal wieder das serielle Ursprungssignal. Die Verfälschung ist +/- 1 Periodendauer des Rechtecksignals. Wenn die Frequenz des Rechtecksignals genügend hoch ist, kann dieser Fehler vernachlässigt werden.

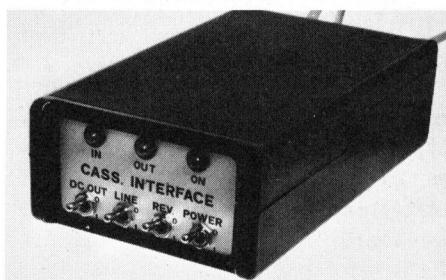
Das TTL-Signal kann noch (wenn gewünscht) invertiert werden und gelangt an den Leitungstreiber,

der wieder ein RS232 Signal erzeugt.

Die sonst überflüssigen Gatter des Treibers (1488) werden zur Ansteuerung der Leuchtdioden verwendet, die anzeigen, ob ein Signal empfangen oder gesendet wird.

Ein weiterer Schalter ermöglicht das Umschalten auf eine externe Informationsquelle; dies ist zum Beispiel dann nötig, wenn man an einem Multiuser-System zuerst ein LOGIN machen muss, bevor die Daten übertragen werden können.

Da die Stromversorgung grosszügig ausgelegt ist, wird noch ein 6 V-Regler betrieben, der als Batterie-Ersatz für das Tonbandgerät benutzt werden kann. Aus diesem Grund ist in der 6 V-Leitung auch ein Schalter; mit ihm kann das Tonbandgerät ein- und ausgeschaltet werden.



## EINSTELLUNGEN:

- 1) Der Spannungsteiler wird folgendermassen eingestellt: Am RS232-Eingang wird ein Testsignal von 500 Hz (rechteckig, +/- 6V) angelegt. Der Ausgang zum Kassettengerät wird auf dem Oszilloscope angezeigt. Der Trimmer ist so einzustellen, dass die Nulllinie zwischen den Rechteckpaketen etwa in der Mitte liegt.
- 2) Der Trimmer der Aufbereitungsschaltung muss so eingestellt sein, dass bei fehlendem NF-Signal der Eingang des 74132 auf LOW liegt.
- 3) Beim Tonbandgerät ermittelt man durch Versuche die günstigste Stellung des Lautstärkerreglers. Bei gewissen Tonbandgeräten müs-

sen die Mikrofonstecker bei der Wiedergabe ausgesteckt werden.

- 4) Die Programmierung: Wenn es sich um ein normales Tonbandgerät handelt, empfiehlt sich eine Taktfrequenz von 3 KHz. Hierfür werden die Timer folgendermassen programmiert:

	SS 8620	SS8640-B
	-----	-----
PIN 1	+5V	*
PIN 2	+5V	OFFEN
PIN 3	OFFEN	+5V
PIN 4	+5V	OFFEN
PIN 5	+5V	OFFEN
PIN 6	OFFEN	+5V
PIN 7	OFFEN	OFFEN
PIN 8	*	*
PIN 9	OFFEN	*
PIN 10	OFFEN	*
PIN 11	OFFEN	*

\* = Keine Programmiergänge

## AUSFUEHRUNG:

Um einen möglichst kompakten Aufbau zu erreichen, wird ein grosser Teil der Widerstände stehend montiert sowie als Netztransformator eine Flachausführung gewählt. Die Platine lässt sich einseitig ausführen, wenn man die Schalter und Leuchtdioden von Hand montiert.

Beim Anschluss des Steckers der seriellen Schnittstelle muss in den Unterlagen der verwendeten Geräte nachgesehen werden, an welchen Pins die entsprechenden Signale anliegen. Normalerweise hat man folgende Konfiguration:

- PIN 1 =Shld (Schutzmasse (nur zur Schirmung, nicht immer angeschlossen))
- PIN 2 =TxD (Information von Terminal zum Modem)
- PIN 3 =RxD (Information von Modem zum Terminal)
- PIN 7 =Gnd (OV (=MASSE, 'Signal Return'))

# GEWUSST WIE!

- Je nach Gerät müssen PIN 2 als RS232 DATA IN und PIN 3 als RS-232 DATA OUT, oder umgekehrt angeschlossen werden.
- Gewisse Geräte benötigen an Pin 5 eine positive Spannung, damit sie überhaupt senden können. (Beim Superbrain: Pin 5 und Pin 20 verbinden)

## Speichern eines Files:

- 1) Einstellen der gewünschten Uebertragungsgeschwindigkeit (Baudrate). Die Anleitung hat für Computer mit CP/M-Betriebssystem Gültigkeit.

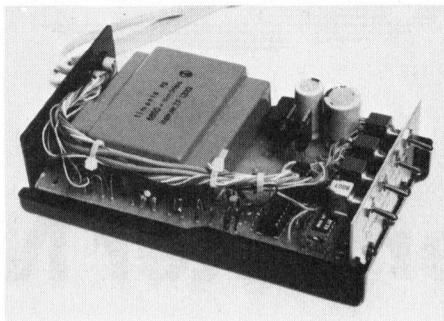
100 Baud = ca. 10 Zeichen/Sec  
1200 Baud = ca. 100 Zeichen/Sec

Je höher die Baudrate, desto kleiner die Datensicherheit!

- 2) 1 oder 2 Stopbits wählen; kein Paritätsbit.

- 3) Uebertragen des gewünschten Informationsblockes mit:

PIP LST:='name.art'



Vor dem Eingeben des "RETURN", Starten des Tonbandgerätes. Jetzt sollte die Uebertragung stattfinden und die Eingangs-Leuchtdiode zu flackern beginnen.

Wenn sich das Betriebssystem wieder mit dem Promptzeichen meldet, ist die Uebertragung beendet.

## Laden eines Files;

- 1) Kassette einlegen und zurückspulen.
- 2) Eingabe von:  
PIP 'name.art'=INP:
- 3) Kassettengerät starten. Jetzt sollte die 'DATA OUT'-Leuchtdiode zu flackern beginnen; d.h. es werden Informationen zum Computer übertragen.
- 4) Wenn das File übertragen ist, sollte sich das Betriebssystem wieder melden. Falls dies nicht der Fall ist, kann man die Uebertragung durch 'ctrl. z' stoppen.

# Wegen Inventar...

Hat Sie dieser Hinweis auch schon verärgert? Nein? Ist Ihr diesjähriges Inventar schon glücklich überstanden? Ja? Sie hätten es einfacher und schneller haben können – mit unserem NUMICRON Lager- und Produktionsprogramm. Durch den MICROCOMPUTER wurde es möglich, den Bedürfnissen eines schweizerischen Klein- bis Mittelbetriebes voll zu entsprechen. Das System verfügt über hohen Anwendungskomfort und Transparenz. Das Programm speichert und organisiert 1750

Artikel pro Floppy Disk und erlaubt Datenzugriffe und Änderungen, Auskünfte und Mutationen innerhalb Sekundenbruchteilen. Standardmässig sind 9 versch. Listen wählbar: Inventarliste, Irregularitäten in den Lagerbeständen, Lagerliste, Preisliste, Dispositionsliste, Lagerbewegungen, Ausstehende Bestellungen, Lagerübersicht.



Av. Louis Casati 81  
CH-1216 Genève  
Tél. 022 98 78 77

**STOLZ  
AG**

Täferstrasse 15  
CH-5405 Baden-Dättwil  
Tel. 056 84 01 51 Tx 54070

**Coupon** Wir möchten unseren Betrieb auch bei Inventur offen halten.

Firma \_\_\_\_\_  
Zuständig \_\_\_\_\_  
Tel. \_\_\_\_\_  
Adresse \_\_\_\_\_



**Unser guter Name  
im Kleincomputergeschäft  
bürgt für Seriosität,  
Kontinuität und grosses  
Hard- und Software-  
Know-how**



DIALOG COMPUTER  
TREUHAND AG  
Seeburgstrasse 18  
6002 Luzern  
☎ 041-3145 45

## Eigene BASIC-Befehle

Christian BUESS

Für gewisse Anwendungen erscheint der Befehlsvorrat des PET als nicht ausreichend, und man wünscht sich neue BASIC-Befehle. Oft wird ein Maschinenprogramm geschrieben, welches mit PEEK, POKE, SYS, USR bedient wird. Dies erfüllt zwar momentan den Zweck, ist aber auch nicht gerade elegant. Wir möchten hier nun zeigen, wie dieses Problem besser gelöst werden kann.

Zuerst muss kurz auf die Arbeitsweise des BASIC-Interpreters eingegangen werden. Die einzelnen Programmzeilen stehen hintereinander im Speicher, wobei jede in eine Zwischenform zerlegt wurde (Befehle belegen nur ein Byte, dieses sogenannte Token ist z.B. für PRINT ein "?"). Wird nun ein Programm abgearbeitet, so liest der Interpreter diese Zeilen zeichenweise durch, und führt die entsprechenden Befehle aus. Um die momentane Position nicht zu verlieren, enthält die Zeropage einen Zeiger, welcher auf das gerade eingelesene Byte hinweist. Zwei Orte erlauben nun ein relativ leichtes Eingreifen in das Betriebssystem:

- 1) Character-Get-Routine (CHRGET): Diese Routine holt den nächsten Character des Programms und rückt obengenannten Zeiger um eins nach vorne. Da sie sich in der Zeropage befindet, kann sie abgeändert werden.
- 2) Interrupt-Routine: Fragt die Tasteratur ab, zählt die Uhr weiter etc., auch sie kann leicht mit Zusätzen versehen werden, da in der Zeropage ein Vektor vorhanden ist, der abgeändert werden kann.

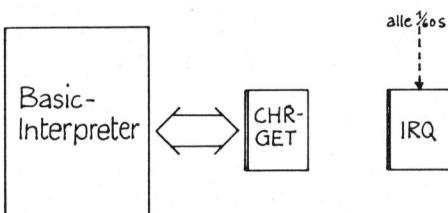


Bild 1

Änderungen des IRQ-Vektors wurden schon früher besprochen (IRQ-Uhr, Repeatfunktion etc.), hier geht es nun um einen Eingriff in die CHRGET-Routine, wobei wir dies an einem Beispiel erläutern wollen.

Es soll ein neuer Basic-Befehl eingeführt werden, der den Cursor in die Zeile  $\alpha$  und die Spalte  $\beta$  bringt. Zu Gunsten der Uebersichtlichkeit soll der Befehl die folgende einfache Struktur haben:

CU ab,cd ; wobei  $ab=\alpha$ ,  $cd=\beta$  und a,b,c,d die Ziffern von zweistelliger Dezimalzahlen sind.

Dieser Befehl wird vom Basic-Interpreter natürlich nicht ausgeführt, denn er erkennt ihn nicht. Wir müssen den Befehl irgendwie kennzeichnen und separat vom eigentlichen Interpreter ausführen. Mit dem Sonderzeichen " ' " führt uns dies auf die neue Schreibweise 'CU ab,cd. Sobald das Sonderzeichen " ' " erscheint, muss in einen neuen Interpreter gesprungen werden, der den Befehl entsprechend verarbeitet. Hierbei sind drei Fälle zu unterscheiden:

- Befehlsausführung im Direct Mode
- Befehlsausführung während des Programmablaufs
- Eingabe einer neuen Programmzeile

Nur in den ersten zwei Fällen muss der Befehl wirklich ausgeführt werden, im letzten hingegen

nicht. Wir erhalten also folgenden Ablauf:

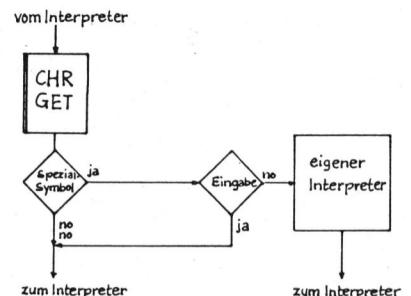


Bild 2 Prüfen auf "neuen" Basic-Befehl

Ist das Sonderzeichen erschienen, und befinden wir uns in einem der ersten beiden Fälle, so muss als nächstes im neuen Interpreter die Syntax des Befehls geprüft werden (hier: die Zeichenfolge CU).

Ist diese richtig, so wird der Befehl ausgeführt (hier: lesen der ersten Zahl, Kommatest, lesen der zweiten Zahl, Bereichstest, Cursorposition ändern). Bei der Rückkehr

### FREIE AUTOREN

Für die verschiedenen Rubriken - Kleincomputer aktuell, Small business, Lehrgänge, PPC/HHC, Hobby mit Mikros, Gewusst wie - sind wir immer an fachlich instruktiven Artikeln von freien Autoren interessiert.

Beiträge, die wir nach sorgfältiger Prüfung abdrucken, honorieren wir angemessen. Legen Sie bitte Ihren Artikeln die notwendigen Diagramme, Zeichnungen und Listings (inklusive Kasette oder Diskette) bei.

VERLAG SCC AG

# GEWUSST WIE!

zum eigentlichen Interpreter muss darauf geachtet werden, dass im Akkumulator das nächste Zeichen nach dem neu definierten Basic-Befehl vorhanden ist, damit der Interpreter normal weiterarbeiten kann.

Aus dem geschilderten Ablauf ist ersichtlich, dass der neue Befehl vom eigentlichen Interpreter gar nicht wahrgenommen wird. Die Abänderung der CHRGET-Routine gestaltet sich sehr einfach, denn es muss

nur auf ein Zeichen (!) getestet werden. Die Routine wird jedoch dadurch um mindestens drei Byte länger und hat deshalb in der Zeropage nicht mehr Platz, muss also teilweise verlegt werden (z.B. in den zweiten Kassettenpuffer). Pro gelesenen Zeichen werden somit also nur drei zusätzliche Befehle (JMP, CMP, BEQ) ausgeführt, was zu praktisch keiner Geschwindigkeitseinbuße führt.

ABGEÄNDERTE CHRGET-ROUTINE UND EIGENER INTERPRETER FUER DEN CURSOR-BEFEHL 'CU ab.cd  
\*\*\*\*\*

```

NEX JMP $70      Sprung zur CHRGET-Routine
IN1 CMP #$3A    1.Einsprungsstelle
BCS OUT
IN2 CMP #$20    2.Einsprungsstelle
BEQ NEX
CMP #$27       Test auf das Sonderzeichen
BEQ INT        Sprung zum neuen Interpreter
SEC
SBC #$30
SEC
SBC #$D0
OUT RTS        Normale Rueckkehr zu Basic
INT CPX #0     Anfang des neuen Interpreters
BNE AD2       Unterscheidung der folgenden Faelle
CPY #1
BNE AD2       -Direct-Mode > AD2
LDA #37       -Programm-Ausfuehrung > AD2
CMP #$FF      -Programm-Einsabe > Zurueck zum Interpreter
BNE AD2
RTS           Zum Interpreter
AD2 JSR $70    Holen des naechsten Zeichens
CMP #$43     Testen auf C
BNE FE0     falls nicht, dann SYNTAX ERROR
JSR $70     Holen des naechsten Zeichens
CMP #$55     Testen auf U
BNE FE0     falls nicht, dann SYNTAX ERROR
JSR INP     Input der naechsten Zahl
CMP #$2C     Test auf Komma
BNE FE0     falls nicht, dann SYNTAX ERROR
LDY $FF     Laden der eingelesenen Zahl
SEC
CPY #25     Testen auf Zeilenzahl >24.
BCS FEH     falls ja, dann ILLEGAL QUANTITY ERROR
JSR INP     Input der naechsten Zahl
LDA $FF     Laden der eingelesenen Zahl
SEC
CMP #40     Testen ob Spaltenzahl >39.
BCS FEH     falls ja, dann ILLEGAL QUANTITY ERROR
STA 198
LDA #0
LDX #128
STY 216
CPY #0
BEQ AUS
ADD CLC     Verändern der Cursorposition
ADC #40
BCS NIC
INX
NIC DEY
BNE ADD
AUS STA 196
STX 197
JMP $76    Ruecksprung zum Interpreter mit neuem Zeichen
FEH LDX #35 Sprung zur Fehler-
JMP 50007 meldung ILLEGAL QUANTITY ERROR
FE0 LDX #10 Sprung zur Fehler-
JMP 50007 meldung SYNTAX ERROR
INP JSR $70 Lesen des naechsten Zeichens
SEC
SBC #$30   Umrechnung zu einer Zahl a
TAX
JSR $70    Lesen des naechsten Zeichens
SEC
SBC #$30   Umrechnung zu einer Zahl b
CPX #0
BEQ NAD
CLC
AD2 ADC #10 Berechnung der HEX-Darstellung der
DEX     zweistelligen DEC-Zahl ab
BNE AD2
NAD STA $FF Abspeichern in $FF
JSR $70    Lesen des naechsten Zeichens
RTS

```

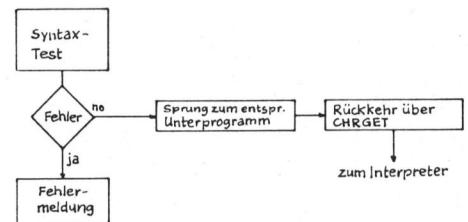


Bild 3 Flussdiagramm des Zusatz-Interpreters

Um dieses Beispiel auszuprobieren geht man folgendermassen vor: Als erstes wird das Maschinenprogramm eingegeben (z.B. ab Adresse 033A, wobei IN1:=033D, IN2:=0341), als zweites muss die CHRGET-Routine geändert werden. Am besten geschieht dies vorerst so:

Auslesen der Speicherplätze 0078-007F über TIM:

```

.M 0078,007F
.: 02 C9 3A B0 0A C9 20 F0
diese werden abgeändert in
.: 02 4C 3D 03 0A 4C 41 03
           IN1      IN2

```

(Diese Änderung kann nicht im Basic erfolgen, sondern nur über TIM oder Maschinenprogramm!)

Der neue Basic-Befehl kann nun beliebig benützt werden, beispielsweise: 'CU 09,15: PRINT"CURSOR TEST".'

Das Prinzip der Einführung individueller Befehle wurde bewusst an einem sehr einfachen Beispiel demonstriert, auf die gleiche Art können auch sehr viel komplexere Anweisungen ausgeführt werden.

# LISTING

\*\*\*\* K U G E L K N A C K N U S S \*\*\*\*

KUGELKNACKNUSS

ROLAND WIRTH

DAS PROGRAMM BEHÖRT IN DIE KATEGORIE 'DENKSPORT' UND BEANSPRUCHT EINEN SPEICHERPLATZ VON 7K BYTE. ES BEFRAGT SICH MIT DEM PROBLEM, WIE MAN AUS ZWELF REISSERLICH GLEICHEN KUGELN MIT DREI WÄGUNGEN DIE JEINIGEN HERRAUSFINDET, DIE SICH VON DEN ANDEREN DURCH EINEN KLEINEN GEWICHTSUNTERSCHIED UNTERSCHIEDET. DIE WÄGUNGEN WERDEN AUF EINER VERGLEICHENDEN WÄGE DURCHFÜHRT UND ES MUSS NICHT NUR DIE KUGEL, DIE ANDERS IST HERRAUSGEFUNDEN WERDEN, SONDERN AUCH OB SIE SCHWERER ODER LEICHTER IST ALS DIE ANDEREN.

DAS PROGRAMM WELCHES SICH SELBST AM BILDSCHIRM ERKLÄRT, BESTEHT EIGENTLICH AUS ZWEI PROGRAMMTHEILEN: DEN ERSTEN, IN DEM DER COMPUTER DIE KUGEL BESTIMMT DIE SICH VON DEN ANDEREN UNTERSCHIEDET UND IN DEM DER MENSCH FESTSTELLEN KANN WIE SCHWERER DIE GESTELLTE AUFGABE IST UND DEN ZWEITEN TEIL, IN DEM DER COMPUTER ZIELSTREBIG DIE KUGEL FINDET, WELCHE VOM 'SPIELER' FESTBELEGT WURDE.

```

160 DIM L(6),R(6)
170 KN=INT(12*RND(1)+1)
180 Y=INT(2*RND(1)+1)
190 IF Y=1 THEN GM#= "S"
200 IF Y=2 THEN GM#= "L"
210 :
220 PRINT "TAB(11)"; KUGELAUFGABE; PRINT : PRINT
230 PRINT "*****"
240 PRINT : PRINT
250 PRINT "DIE SIND ZWELF KUGELN."
260 PRINT "REISSERLICH SEHEN SIE VÖLLIG GLEICH AUS"
270 PRINT : PRINT
280 PRINT "EINE DAVON IST ABER SCHWERER ODER
290 PRINT "LEICHTER ALS DIE ÜBRIGEN !"
300 PRINT "MIT DREI WÄGUNGEN SOLLEN SIE HERRAUS-
310 PRINT "FINDEN WELCHE KUGEL DRS IST UND OB
320 PRINT "SIE LEICHTER ODER SCHWERER IST ALS
330 PRINT "DIE ANDEREN."
340 PRINT : PRINT
350 INPUT "IST DIE AUFGABENSTELLUNG KLAR (J/N)"; K$
360 IF K$="J" THEN GOTO 220
370 W=0
380 PRINT "VORBEREITUNG ZUR 'W' WÄGUNG:"
390 PRINT : PRINT : PRINT
400 PRINT " 0 0 0 0 0 0 0 0 0 0 "
410 PRINT "1 2 3 4 5 6 7 8 9 10 11 12"
420 PRINT : PRINT
430 PRINT "  "
440 PRINT "  "
450 PRINT "  "
460 PRINT "  "
470 PRINT "  "
480 PRINT "  "
490 INPUT "WIEVIELE KUGELN IN JEDE SCHALE"; J
500 IF J=0 OR J>6 THEN GOSUB 1448
510 PRINT "J"
520 FOR I=1 TO J
530 PRINT "KUGEL "I" FUER DIE LINKE SCHALE"; INPUT (NR); L(I)
540 NEXT I
550 PRINT : PRINT
560 FOR I=1 TO J
570 PRINT "KUGEL "I" FUER DIE RECHTE SCHALE"; INPUT (NR); R(I)
580 NEXT I
590 PRINT : PRINT : PRINT
600 GOSUB 1060
610 PRINT "OK DANKE !"
620 FOR I=1 TO 1000 : NEXT I
630 PRINT "ZERGEBNIS DER 'W' WÄGUNG:"
640 PRINT : PRINT
650 PRINT "KUGEL LINKS ";
660 FOR I=1 TO J : PRINT L(I); : NEXT I
670 PRINT : PRINT
680 PRINT "KUGELN RECHTS ";
690 FOR I=1 TO J : PRINT R(I); : NEXT I
700 PRINT : PRINT : PRINT
710 GOSUB 1310
720 PRINT : PRINT : PRINT "FERTIG? -DANN DRUECKEN SIE EINE TASTE"
730 GETC: IFC#="" THEN I=730
740 IF I=3 THEN I=760
750 GOTO 370
760 PRINT "SO, DAMIT SOLLTEN SIE EIGENTLICH 'INDER'
770 PRINT "LAGE SEIN, DIE GESUCHTE KUGEL ZU"
780 PRINT "IDENTIFIZIEREN - LAUT 'SPIELREGELN'."
790 PRINT : PRINT : PRINT
800 PRINT "ALSO, WELCHES IST DIE FRAGLICHE."
810 INPUT "KUGEL"; KM
820 IF KN=KM THEN PRINT "RICHTIG" : GOTO 850
830 PRINT "DAS IST LEIDER FALSCH GERATEN UND DIE
840 PRINT "FRAGE NACH DEM GEWICHT ERUEBRIGT SICH." : GOTO 988
850 PRINT : PRINT "UND IST SIE SCHWERER"
860 PRINT "ODER LEICHTER" ALS DIE ÜBRIGEN?"
870 INPUT "S/L"; G$
880 IF G$="S" THEN I=970
890 PRINT "CORATULIERE! RICHTIG !!"
900 PRINT "SIE SIND JA EIN ASS !!!"
910 PRINT "ODER HABEN SIE DAS VIELLEICHT NUR
920 PRINT "GUT GERATEN? ---"
930 PRINT : PRINT "STELLEN SIE IHR KOENNEN UNTER BEWEIS
940 PRINT "UND VERSUCHEN SIE ES NOCH EINMAL!"
950 PRINT : PRINT
960 GOTO 1020
970 PRINT "DAS WAR LEIDER FALSCH !"
980 PRINT : PRINT
990 PRINT "DIE RICHTIGE LOESUNG IST:"
1000 PRINT "KUGEL NR. "KN" UND "GM#" FUER'S GEWICHT.
1010 PRINT : PRINT
1020 PRINT "WOLLEN SIE'S NOCHMAL PROBIEREN ?"
1030 GETAN$: IFA#="" THEN I=1030
1040 IF AN#="" THEN I=490
1050 RUN
1060 REM PRUEFUNG KUGELUELTIGKEIT
1070 REM KUGEL UNGUELTIGER NUMMER
1080 FOR I=1 TO J
1090 IFL(I)<10R(I)>12 THEN I250
1100 IFR(I)<10R(I)>12 THEN I250
1110 NEXT I
1120 REM GLEICHE KUGEL RECHTS UND LINKS
1130 FOR I=1 TO J
1140 FORK=1 TO J
1150 IFL(I)=R(K) THEN I250
1160 NEXT K : I
1170 IF J<I THEN RETURN
1180 REM GLEICHE KUGEL IN GLEICHER SCHALE
1190 FOR I=1 TO (J-1)
1200 FORK=(I+1) TO J
1210 IFL(I)=L(K) THEN I250
1220 IFR(I)=R(K) THEN I250
1230 NEXT K : I
1240 RETURN
1250 PRINT "SIE VERWENDETEN KUGELN DIE ES NICHT
1260 PRINT "GIBT BZW. DIE GLEICHE KUGEL MEHRFACH
1270 PRINT : PRINT
1280 PRINT "VERSUCHEN SIE'S NOCHMAL"
1290 FOR I=1 TO 3000 : NEXT I
1300 GOTO 988
1310 REM AUSWERTUNG DER WÄGUNG
1320 FOR I=1 TO J
1330 IFL(I)=KN THEN I380
1340 IFR(I)=KN THEN I400
1350 NEXT I
1360 PRINT : PRINT "ODER WÄRGEBALEN BLEIBT WÄRGE RECHT!"
1370 RETURN
1380 IF GM#="S" THEN I420
1390 IF GM#="L" THEN I430
1400 IF GM#="S" THEN I430
1410 IF GM#="L" THEN I420
1420 PRINT : PRINT "ODER WÄRGEBALEN SENKT SICH LINKS"; RETURN
1430 PRINT : PRINT "ODER WÄRGEBALEN SENKT SICH RECHTS"; RETURN

```

```
1440 PRINT "SIE MUESSEN MINDESTENS 1 UND KOENNEN
1450 PRINT "HOECHSTENS 6 KUGELN IN JEDE WARRSCHALE
1460 PRINT "LEGEN !
1470 FOR I=1 TO 3000: NEXT
1480 GOT02380
1490 PRINT "WOLLEN SIE WIRKLICH AUFGEBEN, ODER
1500 PRINT "SOLL ICH EINMAL VERSUCHER DIESE AUFGABE
1510 INPUT "ZU LOESEN (JA/N)"; R$
1520 CLR: PRINT "I'WHEN END
1530 FOR C: PRINT "ZUT, DANN WOLLEN MIR ES EINMAL SO HERUM
1540 PRINT "VERSUCHEN;
1550 PRINT "PRINT "SIE MUESSEN SICH DAZU EINE IER ZWEIFL
1560 PRINT "KUGELN AUSWAELHEN UND DAZU FESTLEGEN, OB
1570 PRINT "SIE SCHWERER ODER LEICHTER SEIN SOLL
1580 PRINT "ALS DIE UEBRIGEN;
1590 PRINT "MIR DWERFEN SIE DAS NATUERLICH VORHER
1600 PRINT "NIGHT VERZETEN; ABER AM BESTEN, SIE
1610 PRINT "NOTIEREN SICH IHRE FESTLEGUNG;
1620 PRINT "PRINT "BITTE DRUECKEN SIE EINE TASTE
1630 GET$: IF R$="R" THEN L=30
1640 PRINT "SIE HABEN SICH NUN DIE KUGEL GEWAHLT
1650 PRINT "UND DAZU NOTIERT OB SIE SCHWERER ODER
1660 PRINT "LEICHTER SEIN SOLL ALS DIE ANDEREN;
1670 PRINT "PRINT "ICH WERDE IHEN NACHGEHENDER DREI
1680 PRINT "WAEGLUNGEN NENNEN FUER DIE SIE MIR DAS
1690 PRINT "WAEGLUNGS NENNEN MUESSEN, D.H. WIE SICH
1700 PRINT "DAS WAEGERLAKEN VERHAHLT;
1710 PRINT "SIE DIE GESUCHTE KUGEL KENNEN, IST
1720 PRINT "DAS SICHER NICHT SCHWERER;
1730 PRINT "PRINT "INPUT "FANGEN MIR'S AN (JA/N)"; C$
1740 IF C$="Y" THEN L=90
1750 CLR: DIRD$(3)
1760 WHEN+1
1770 PRINT "HIER DIE KUGELVERTEILUNG ZUR"; W; "WAEGLUNG
1780 IF W=2 THEN L=900
1790 IF W=3 THEN L=1940
1800 PRINT "PRINT "####KUGELN, LINKE SCHALE NR. 1, 2, 3, 4
1810 PRINT "####KUGELN RECHTE SCHALE NR.5, 6, 7, 8
1820 PRINT "PRINT "BITTE SAGEN SIE MIR NUN, WIE SICH DER
1830 PRINT "WAEGERLAKEN VERHAHLT;
1840 PRINT "PRINT "#### BLEIBT ER WAERERECHEIT
1850 PRINT "PRINT "#### SENKT SICH DIE LINKE SEITE OBER
1860 PRINT "PRINT "#### SENKT SICH DIE RECHTE SEITE
1870 PRINT "INPUT "GABE DES ENTSPRECHENDEN BUCHSTABEN"; D$(K)
1880 IF W=3 THEN L=2040
1890 GOT01750
1900 IF D$(1)="B" THEN PRINT "PRINT "####KUGELN LINKS 9 10 11"; PRINT: PRINT: PRINT "####KUG
KUGELN RECHTS 5, 6, 7";
1910 IF D$(1)="L" THEN PRINT: PRINT "####KUGELN LINKS 4 6 7"; PRINT: PRINT: PRINT "####KUG
ELN RECHTS 3 8 9";
1920 IF D$(1)="R" THEN PRINT: PRINT "####KUGELN LINKS 3 8 9"; PRINT: PRINT: PRINT "####KUG
ELN RECHTS 4 6 7";
1930 GOT01820
1940 IF D$(2)="R" AND D$(1)="B" THEN PRINT: PRINT "LINKS 12"; PRINT: PRINT: PRINT "RECHTS 5"
1950 IF D$(2)="B" AND D$(1)="L" THEN PRINT: PRINT "LINKS 1"; PRINT: PRINT: PRINT "RECHTS 2"
1960 IF D$(2)="B" AND D$(1)="R" THEN PRINT: PRINT "LINKS 2"; PRINT: PRINT: PRINT "RECHTS 1"
1970 IF D$(2)="L" AND D$(1)="B" THEN PRINT: PRINT "LINKS 9"; PRINT: PRINT: PRINT "RECHTS 10"
1980 IF D$(2)="L" AND D$(1)="L" THEN PRINT: PRINT "LINKS 9"; PRINT: PRINT: PRINT "RECHTS 8"
1990 IF D$(2)="L" AND D$(1)="R" THEN PRINT: PRINT "LINKS 9"; PRINT: PRINT: PRINT "RECHTS 9"
2000 IF D$(2)="R" AND D$(1)="B" THEN PRINT: PRINT "LINKS 8"; PRINT: PRINT: PRINT "RECHTS 10"
2010 IF D$(2)="R" AND D$(1)="L" THEN PRINT: PRINT "LINKS 6"; PRINT: PRINT: PRINT "RECHTS 7"
2020 IF D$(2)="R" AND D$(1)="R" THEN PRINT: PRINT "LINKS 7"; PRINT: PRINT: PRINT "RECHTS 6"
2030 GOT01820
2040 REM ENDEBETRACHTUNG
2050 IF D$(1)="L" THEN L=2080
2060 IF D$(1)="R" THEN L=2320
2070 IF D$(1)="R" THEN L=2320
2080 IF D$(2)="L" THEN L=2110
2090 IF D$(2)="B" THEN L=2140
2100 IF D$(2)="R" THEN L=2170
2110 IF D$(3)="L" THEN L=8: GW#="L": GOT02440
2120 IF D$(3)="B" THEN L=4: GW#="S": GOT02440
2130 IF D$(3)="R" THEN L=12: GW#="S"
2140 IF D$(3)="L" THEN L=30: TA(3)=50: PG#=-2112
2150 IF D$(3)="B" THEN L=5: GW#="L": GOT02440
2160 IF D$(3)="R" THEN L=2: GW#="S": GOT02440
2170 IF D$(3)="L" THEN L=7: GW#="L": GOT02440
2180 IF D$(3)="B" THEN L=3: GW#="S": GOT02440
2190 IF D$(3)="R" THEN L=6: GW#="L": GOT02440
2200 IF D$(2)="L" THEN L=2330
2210 IF D$(2)="B" THEN L=2260
2220 IF D$(2)="R" THEN L=2290
2230 IF D$(3)="L" THEN L=9: GW#="S": GOT02440
2240 IF D$(3)="B" THEN L=11: GW#="S"
2250 IF D$(3)="R" THEN L=10: GW#="S": GOT02440
2260 IF D$(3)="L" THEN L=12: GW#="S"
2270 IF D$(3)="B" THEN L=13: GW#="S": GOT02440
2280 IF D$(3)="R" THEN L=12: GW#="L": GOT02440
2290 IF D$(3)="L" THEN L=18: GW#="L": GOT02440
2300 IF D$(3)="B" THEN L=11: GW#="L": GOT02440
2310 IF D$(3)="R" THEN L=9: GW#="L": GOT02440
2320 IF D$(2)="L" THEN L=2350
2330 IF D$(2)="B" THEN L=2380
2340 IF D$(2)="R" THEN L=2410
2350 IF D$(3)="L" THEN L=8: GW#="S": GOT02440
2360 IF D$(3)="B" THEN L=4: GW#="L": GOT02440
2370 IF D$(3)="R" THEN L=10: GW#="L": GOT02440
2380 IF D$(3)="L" THEN L=11: GW#="L": GOT02440
2390 IF D$(3)="B" THEN L=5: GW#="S": GOT02440
2400 IF D$(3)="R" THEN L=2: GW#="L": GOT02440
2410 IF D$(3)="L" THEN L=7: GW#="S": GOT02440
2420 IF D$(3)="B" THEN L=3: GW#="L": GOT02440
2430 IF D$(3)="R" THEN L=6: GW#="S"
2440 PRINT "#####MEINE LÖSUNG: ■
2450 PRINT "#####DIE VON IHEN GEWAHLTE KUGEL WAR ■R, ■KN
2460 PRINT
2470 IF GW#="L" THEN L=2: GW#="S" LEICHTER■
2480 IF GW#="S" THEN L=3: GW#="R" SCHWERER■
2490 PRINT "UND SIE IST "E" ALS DIE ANDEREN !
2500 PRINT: PRINT: INPUT "NOCHMAL (JA/N)"; K$
2510 IF K$="Y" THEN GOTO
2520 GOT01750
2530 PRINT "IRGEND EINE IHRER ANTWORTEN MUSS FALSCH
2540 PRINT "GEWESEN SEIN --- DENN DIE KOMBINATION
2550 PRINT "IHRER DREI ANTWORTEN IST UNMOEGLICH !
2560 GOT02500
READY.
***** H A N O I ****
-----
CHRISTIAN MEIER
-----
IN HANOI LEBENDE MOENCHE BESASSEN DREI NADELN UND 64 VERSCHIEDEN GROSSE
GOLDSCHREIBEN; DIE ALLE DER GROESSE NACH AUF DER LINKEN NADEL LAGEN. DIE
KLEINSTE SCHEIBE ZUERST DAS ZIEL DER MOENCHE WAR NUN, DIE SCHEIBEN SO
UMZUSCHICHTEN, DASS SIE AM ENDE IN GLEICHER REIHENFOLGE AUF DER RECHTEN
NADEL AUFGESCHICHTET SEIN SOLLTEN WAEREND DES VORGANGES ABER DURFTE NIE
EINE KLEINERE SCHEIBE UNTER EINE GROESSERE ZU LIEGEN KOMMEN. WERDEN DIE
MOENCHENS IHR WERK VOLLENDEN, SO WIRD DIE WELT UNTERGEHEN. SO SAGT ES WE-
NIGSTENS DIE LEGENDE.
1 REM Die Tueme von Hanoi: 22.2.81 Christian Meier
2 REM Sorcererversion mit graphischer Darstellung
3 REM Variablen
4 REM N: Scheibenzahl, Turmhoehe; L: noetige Zuege
5 REM TA: Tabulator fuer Grafik; PG: Grundposition Grafik
6 REM S: Scheibenposition; B: zu bewegende Scheibe
7 REM ST: Scheibenstapel; H: Stapelhoehoe
8 REM BS: Gerade zu bewegende Scheibe; BN die naechste Scheibe
9 REM A: neue Position; AA: alte Position von 1
10 PRINT CHR$(12): INPUT "Anzahl Scheiben"; N: IF N < 1 OR N > 10 THEN 1
20 L=2: N=1: DIM BE(L), ST(3), S(N), S(N)
30 TA(1)=10: TA(2)=30: TA(3)=50: PG#=-2112
40 FOR X=-512 TO -505: POKE X, 0: NEXT X: POKE -512, 255: POKE -505, 255
```

# LOTINO

```

125 ; "WIEVIEL DER 4 ZIFFERN DU RICHTIG HERAUS"
126 ; "GEFUNDEN, ABER NOCH NICHT AN RICHTIGER "
127 ; "STELLE PLAZIERT HAST. (-GUT-) "
128 ; "AUSSERDEM SAGE ICH DIR WELCHE ZIFFERN "
129 ; "RICHTIG SIND UND AN RICHTIGER STELLE "
130 ; "STEHEN. (-ASS-) "
131 ; "DU HAST ^BRIGENS NUR 4 MINUTEN ZEIT !!"
132 ; "VIEL SPASS !"
133 ; "DR^CK <RETURN> WENN DU SOWEIT BIST. "
134 GET J#
135 ; CHR$(12)
140 RANDOMIZE
141 R=(10*(RND))
142 H=INT(R)
143 IF H=0 THEN 140
144 I=I+1
145 A(I)=H
146 IF I<=3 THEN 140
147 I=1
148 T=2
149 IF A(I)=A(T) THEN 138
150 T=T+1
151 IF T<=4 THEN 149
152 I=I+1
153 T=I+1
154 IF I<=3 THEN 149
155 I=0
156 T=0
157 O=0
-UHR BEGINNT
159 S%Z=PEEK(65010%); S%Z=PEEK(65009%); SIX=PEEK(65008%)
160 ; CUR(0,0)"COUNT-DOWN BEGINNT: 4 : 00"
161 GOTO 163
162 ; "DIE ZAHL HEISST:"; ; A(1);A(2);A(3);A(4); GOTO 246
163 S=S+1
164 IF S>15 THEN 241
165 S=S-1
166 ; TAB(0)"GIB DIE ZAHL!"; ; ONERRORGOTO 254 ; INPUT Z ; I=0 ; GOTO 234
INPUTAUSWERTUNG
168 B=Z
169 IF B>9999 THEN 243
170 H=Z/1000
171 R=INT(H)
172 IF R=0 THEN 242
173 F=H-R
174 Z=F*10000
175 I=I+1
176 D(I)=R
177 IF I<=3 THEN 170
178 I=1
179 T=2
180 IF D(I)=D(T) THEN 247
181 T=T+1 ; IF T<=4 THEN 180
182 I=I+1
183 T=I+1
184 IF I<=3 THEN 180
185 I=1 ; R=0 ; U=0
186 T=1
187 IF A(I)=D(T) THEN 192
188 T=T+1 ; IF T<=4 THEN 187
189 I=I+1 ; IF I<=4 THEN 186
190 S=S+1
191 GOTO 196
192 IF I=T THEN 195
193 R=R+1
194 GOTO 188
195 U=U+1 ; GOTO 188
196 Q(S)=R ; P(S)=U ; L(S)=B
197 ; CHR$(12)
198 ; CUR(0,0)"COUNT-DOWN: ",3-M%"; "60-S%";
DIE GRAFIK BEGINNT

```

```

50 FOR X=1 TO N:ST(1,N-X+1)=X:S(X)=1:NEXT X:Y,X
60 FOR X=1 TO N:ST(1,N-X+1)=X:S(X)=1:NEXT X:H(1)=N
70 INPUT"Geschwindigkeit";W:BS=:PS=S(BS):GOSUB 4000
80 A=3:IF N/2=INT(N/2) THEN A=2
90 ST(1,H(1))=0:H(1)=1:H(A)=H(A)+1:ST(A,H(A))=1:S(1)=A
100 GOSUB 1000:IF N=1 THEN 3000
110 FOR DL=2 TO L-1
120 BS=BE(DL) ; PS=S(BS):REM Position der Scheibe ST(PS,H(PS))
130 BN=BE(DL+1):PN=S(BN):REM Position der Scheibe ST(PN,H(PN))
140 IF PS=PN THEN 200
150 A=6-PN-PS:ST(PS,H(PS))=0:H(PS)=H(PS)-1
160 H(A)=H(A)+1:ST(A,H(A))=BS:S(BS)=A:IF BS=1 THEN AA=PS
170 GOSUB 1000:NEXT DL
180 GOTO 210
200 A=6-PN-AA:ST(PS,H(PS))=0:H(PS)=H(PS)-1:GOTO 160
210 ST(S(1),1)=0:ST(3,N)=1
220 PS=S(1):A=3:H(1)=0:H(3)=N:BS=1
230 H(2)=0:GOSUB 1000
240 PRINT"Fertig!!";SFC(10);:INPUT"Nachmals";A$
250 IF A$="Ja" OR A$="JA" THEN RUN
260 END
990 END
1000 THH(PS)+:PO=PG-TH*64+TA(PS)
1005 FOR X=1 TO W:NEXT X
1010 FOR Q=PO-BS TO PO+BS:POKE Q,32:NEXT Q
1020 THH(A):PO=PG-TH*64+TA(A)
1030 POKE PO-BS,135:FOR Q=PO-BS+1 TO PO+BS-1:POKE Q,192:NEXT Q
1040 POKE Q,128:RETURN
3000 END
4000 FOR X=N TO 1 STEP -1
4010 THN-X+1:PO=PG-TH*64+TA(1):BS=X:GOSUB 1030:NEXT X
4020 RETURN
READY
BYE

```

\*\*\*\*\* M A S T E R M I N D \*\*\*\*\*

-----  
ANDREAS WARTSCHITSCH  
-----

MASTERMIND-PROGRAMME FINDET MAN MOHL BALD AUF ALLEN COMPUTERN UND PROGRAMMIERBAREN TASCHEPCHEM. DIE VORLIEGENDE VERSION IST FUER DEN REC 80 GESCHRIEBEN WORDEN. ES ZEICHNET SICH DURCH SEINEN KOMFORT AUS UND ERKLART SICH SELBER AM BILDSCHIRM. MIR WUENSCHEN IHNEN VIEL SPASS MIT DIESEM WELTBEKANNTEN SPIEL.

```

102 REM &&&&& MASTER - MIND &&&&&
SPRUNG ZUR ANFANGSGRAFIK
104 GOSUB 221
105 FOR V=1 TO 2300 : NEXT V
EINF^HRUNG
107 ; CHR$(12):CUR(11,0);"BENUTZT DU INFORMATIONEN ?(J/N)"; GET Q# ; Q#
108 IF Q#="J" OR Q#="j" THEN 112
109 GOTO 135
110 S=0
111 ;
112 ; CHR$(135) ; ; CHR$(12)
113 ;
114 ; TAB(13)"MASTER-MIND";TAB(38)";"
115 ;
116 ;
117 ; "HALLO, ICH BIN DEIN MASTER-MIND PARTNER!"
118 ; "ICH DENKE MIR EINE 4-STELLIGE ZAHL UND "
119 ; "DU SOLLST DIESE ANHAND DER ANGABEN, DIE "
120 ; "ICH DIR JENEILS GEBE,HERAUSFINDEN!"
121 ; "DU KANNST DIES INNERT 15 VERSUCHEN PRO-"
122 ; "BIEREN. "
123 ; "SOBALD DU DIE ZAHL, DIE DU F^R RICHTIG"
124 ; "HCLST EINGEGEBEN HAST, ZEIGE ICH DIR AN. "

```

```

200 FOR C=0 TO 38
"
202 NEXT C
"
204 FOR V=0 TO 38
"
206 NEXT V
207 ; ""
208 FOR N=1 TO S
"
210 NEXT N
"
212 N=N-1
213 IF P(N)=4 THEN 244
214 GOTO 163
215 ; CHR$(12)
"
217 ; CUR(10,0)"BIS ZUM NICHSTEN MAL ! ABC 80 "
"
219 ; CUR(35,0) : END
TITELGRAFIK
221 T%=1%
222 FOR I=1 TO 24 ; ; ; CHR$(151) ; NEXT I
223 FOR K%=0% TO 22%
224 FOR K2=1% TO 39%
225 IF R%<3% OR R%>19% THEN 227
226 IF K%>4% AND K%<36% THEN 228
227 ; CUR(R%,K2) ; IF T%>0% THEN ; "f" ; ELSE ; "9" ;
228 NEXT K2 ; T%=-T% ; NEXT R%
229 ; CUR(8,10);CHR$(135)"ABC80 Pr(sentient: "CHR$(151));
230 ; CUR(11,10);CHR$(135) MASTER - MIND"CHR$(151);

```

\*\*\*\* Z A H L E N L O T T O \*\*\*\*

-----  
A. Martschitsch  
-----

```

40 PRINT CHR$(12)
20 PRINT "PRINT "
30 PRINT CHR(10,6) *****
31 PRINT CHR(10,6) "VERMEHREN AUCH SIE IHR GELD!"
40 PRINT CUR(15,9) "Nach dem alten Motto:"
50 PRINT CUR(18,7) "-LIEBER GESUND UND REICH-"
60 PRINT CUR(19,16) "-ALS ARM UND KRANK-"
70 FOR I=1 TO 10000 : NEXT I
80 FOR I=1 TO 23 : PRINT CHR$(151) : NEXT I
90 K1=5 : Z1=5 : E=70 : F=60 : N=0
100 N=N+1
110 K1=K1+1 : Z1=Z1+1 : E=E-2 : F=F-2
120 FOR Z=0 TO E : SETDOT K1,Z1+Z : NEXT Z
130 Z2=Z1+Z
140 FOR Z=0 TO F : SETDOT K1+Z,Z2 : NEXT Z
150 K2=K1+Z
160 FOR Z=0 TO E : SETDOT K2,Z2-Z : NEXT Z
170 Z1=Z2-Z
180 FOR Z=0 TO F : SETDOT K2-Z,Z1 : NEXT Z
190 K1=K2-Z
200 IF N=5 THEN 210 ELSE 100
210 PRINT CUR(5,10)"LOTTOZAHLENGENERATOR"
220 PRINT CUR(6,10)"-----"
230 PRINT CUR(10,9);CHR$(131)

```

```

231 ; CUR(12,10);CHR$(135) " *****"CHR$(151);
232 RETURN
GEHVRT ZUR UHR
234 E3%=PEEK(65010%); E2%=PEEK(65009%); E1%=PEEK(65008%)
235 T=S1%-E1%+256*((S2%-E2%)+256*((S3%-E3%))
236 T=T/50
237 H%=T/3600 ; T=T-3600*H% ; M%=T/60 ; S%=T-60*M%
238 IF T>240 THEN 249
239 GOTO 168
GEHVRT ZUR INPUTAUSWERTUNG : AUSSAGEN ZU VERSCHIEDENEN ZUSTANDEN
241 OUT 6,7 : FOR I=1 TO 2000 : NEXT I ; "DU HAST VERLOREN - FLASCHE!!!" : OUT
6,0 : GOTO 162
242 FOR T=1 TO 250 : OUT 6,0 ; "NULL NICHT ERLAUBT !" : G
OTO 163
243 FOR T=1 TO 250 : OUT 6,195 : NEXT T : OUT 6,0 ; "ZAHL ZU GROSS !" : GOTO 1
63
244 FOR I=1 TO 20 : FOR H=1 TO 36 : OUT 6,137 : NEXT H : OUT 6,0 : NEXT I ; "B
RAVO, DIESE ZAHL STIMMT GENAU!!!"
245 FOR T=1 TO 50 : OUT 6,6+32+4+1 : NEXT T : OUT 6,0
246 ; "WILLST DU NOCH EINMAL?(J/N)"; ; GET M$ ; ; M$ : IF M$="J" OR M$="j" THEN
107 ELSE 215
247 FOR T=1 TO 250 : OUT 6,195 : NEXT T : OUT 6,0 ; "GLEICHE ZIFFERN NICHT ERL
AUBT" : GOTO 163
SIRENE
249 OUT 6,7
250 ; CHR$(12)CUR(11,0)"DEINE ZEIT IST ABGELAUFEN - PECH !"
251 FOR I=1 TO 5000 : NEXT I
252 OUT 6,0
253 GOTO 162.
254 FOR T=1 TO 60 : OUT 6,255 : NEXT T : OUT 6,0 : GOTO 166
255 END

```

```

234 PRINT CUR(10,10)" 6 AUS 42 "
236 PRINT CUR(10,30);CHR$(151)
240 PRINT CUR(17,10) " DRUECKEN SIE EINE "
250 PRINT CUR(18,10) " BELIEBIGE TASTE "
260 GET B$
270 PRINT CUR(5,9);CHR$(131)
274 PRINT CUR(5,10) " NEIN TIP-VORSCHLAG":
276 PRINT CUR(5,30);CHR$(151)
280 PRINT CUR(12,10) " "
290 A=1
300 A=A-1
310 A=A+1
320 IF A=7 THEN 430
330 RANDOMIZE
340 Z=INT((42*RND)+1)
350 IF Z=0 THEN 300
360 Z(A)=Z
370 IF A=1 THEN 310
380 I=0
390 I=I+1
400 IF I=A THEN 310
410 IF Z(A)=Z(I) THEN 300 ELSE 390
420 GOTO 290
430 PRINT CUR(12,9);CHR$(131)
440 FOR I=1 TO 6 : PRINT CUR(12,8+3*I) : NEXT I
450 PRINT CUR(12,30);CHR$(151)
460 PRINT CUR(14,10) " WOECHTEN SIE EINEN "
470 PRINT CUR(15,8);CHR$(131)
474 PRINT CUR(15,9) " WEIEREN TIP-VORSCHLAG?"
476 PRINT CUR(15,32);CHR$(151)
480 GET B$ : GOTO 270

```



## ERFA-GRUPPEN

Nach wie vor hören wir wenig von Erfahrungsaustausch-Gruppen oder regionalen Computer Clubs. Meistens sind es dazu nur Strohfeder. Dies ist eigentlich sehr schade für unser Hobby. Neben viel Engagement (und Niemand ist dann dankbar...) und Durchhaltewillen von einem oder einigen wenigen benötigt ein solcher Club sicher auch ein Sprachrohr, wenn er Bestand haben soll. Für letzteren Zweck stehen diese Clubinformationen offen.

Ein kurzer Brief mit Informationen genügt. Am besten geben Sie Ihre Zielsetzung oder Ihr Programm, Ort und Datum der Treffs sowie eine Kontaktadresse bekannt. Oder wie wär's mit interessanten Feststellungen über ein System, ein Spiel, eine Programmiermethode etc.?

Wir können uns kaum vorstellen, dass andere Organisationen - die wir zum Teil bereits erwähnen konnten - lediglich "Einjahres-Fliegen" waren und hoffen, dass diese Zeilen dazu ermuntern das Textverarbeitungs-System einmal für Meldungen zuhanden der nächsten Clubinformationen zu verwenden. Wie wir erfahren haben, verhilft eine solche Veröffentlichung zu einer Bereicherung der Gruppe mit neuen Mitgliedern und neuen Ideen. Immerhin lesen weit mehr als 10'000 interessierte Personen diese Zeitschrift und viele wären froh, in Ihrem näheren Umkreis Kollegen zu finden.

Eine löbliche Ausnahme in dieser Szene bilden eigentlich nur der MICROCLUB LAUSANNE und die SCC-SORCERER GRUPPE, welche uns regelmässig mit ihren News informieren und die neuen Daten ihrer Treffs mitteilen:

## SCC-SORCERER-GRUPPE

Es gibt aus unserer Gruppe einige Neuigkeiten zu berichten, die zum Teil auch für Neueinsteiger interessant sind.

Als Erstes haben wir eine Kassette zusammengestellt, die durch den SCC ab sofort mit jedem bei ihm gekauften SORCERER gratis abgegeben wird. Es finden sich auf dieser Kassette eine kurze Vorstellung unserer Gruppe und einige Demo- und Spielprogramme.

Zweitens wird in Kürze eine Lehrkassette erhältlich sein, welche für Beginner vorgesehen ist. Diese

Kassette wird jedem neuen Mitglied unserer Gruppe ebenfalls gratis abgegeben und dient als Starthilfe. Es finden sich auf dieser Kassette die gängigsten Anweisungen in Form von kurzen Demo-Programmen.

Als Drittes sind ab Mitte Juli unsere Mitteilungsblätter 1980/81 in Umfang von ca. 80 - 100 Blatt A4 als geheftete Broschüre für Nichtmitglieder erhältlich. Diese Broschüre kann bis Ende Juni gegen Voreinsendung von Fr. 25.-- (Schweiz), resp. DM 40.-- (Deutschland) inkl. Porto bei unserer Kontaktadresse bestellt werden. Der Inhalt beschreibt hardwaremässige Probleme und bringt viele Problemlösungen und Vorschläge zum Thema Programmieren.

Der Elan in der Gruppe hält nach wie vor unvermindert an. Interessant ist, wie die jungen und jüngsten Mitglieder mit brillanten Arbeiten in Erscheinung treten. Es wird uns Mittelalterlichen daher nichts anderes übrigbleiben, als uns richtig ins Zeug zu legen, um nicht hoffnungslos ins Hintertreffen zu gelangen.

SCC-SORCERER-Gruppe, Postfach 9,  
CH-3294 Büren a.A.

## MITGLIEDER HELFEN EINANDER

Unter diesem Titel veröffentlichen wir regelmässig Adressen von Mitgliedern für deren Systeme wir nur gelegentlich Anfragen erhalten. Bitte nehmen Sie direkten Kontakt auf, wenn Sie ein gleiches System besitzen und an einem Gedankenaustausch interessiert sind.

### KIM

- Alfred Ammann  
Weingartenstrasse 6  
9242 Oberuzwil

### AIM 65

- Ernst Burgermeister  
Im Zentrum 11  
8604 Volketswil  
- Manfred Wälchli  
Seftigenstrasse 30  
3007 Bern  
- Arno Staub  
Senzachstrasse 27  
8413 Neftenbach  
- Walter Spahn  
Brüttenweg 32  
8052 Zürich  
- Reinhard Burkert  
Wildsbergstrasse 8  
8610 Uster  
- Thomas Zeidler  
Haldenstrasse 15  
2502 Biel

- Bernhard Piller  
Les Grands Champs  
1164 Buchillon  
Z80 Selbstbau  
- Rudolf Stocker  
Müllerwis 24  
8606 Greifensee  
MC 6800 Selbstbau  
- Peter Bickel  
Neuweg 19  
8125 Zollikerberg  
INS 8070 Selbstbau  
- Christoph Pape  
Binzhofstrasse 7  
8404 Winterthur  
IMSAI 8080  
- Gabe Hauer  
Kügeliloo Strasse 97  
8046 Zürich  
SINCLAIR  
- Franz G. Rötger  
Freih. vom Steinstrasse 14a  
D-6550 Bad Kreuznach  
SESAM  
- Hans Gyger  
Kirchweg 8  
8604 Volketswil  
- R. Dähler  
Im Sträler 23  
8047 Zürich  
DAI  
- Heinz Ursprung  
Grosswiesenstrasse 140  
8051 Zürich  
- Emil Zahner  
Zürichstrasse 156  
8910 Affoltern  
- Christian Gnägi  
Ipsachstrasse 16  
2563 Ipsach  
HP 85  
- Karl Baer  
Holzhäusernstrasse 13  
6313 Menzingen  
SORD  
- Rolf Winter  
c/o Jacques Arber AG  
Solothurnerstrasse 15  
4053 Basel  
- Nicolas Saalfrank  
Via Arch. Frizzi 34  
6648 Minusio  
- Moritz Holenstein  
Neugasse 14  
9532 Rickenbach  
- Peter Gränicher  
Ruhbergstrasse 11  
9230 Flawil  
ABC 80  
- Hans Habegger  
Langenfuhren  
3294 Büren a.A.  
- Thomas M. Schmidhofer  
Oberdorfstrasse 445  
1712 Tafers  
- Josef Minder  
Schulstrasse 21  
9323 Steinach  
- Lorenzo Lardelli  
Ottostrasse 8  
7000 Chur

# Clubinformationen

- Erhard Hofmann  
Postfach 13  
D-8934 Grossaitingen
- Richard Sidler  
Oberbühl 151  
9491 Gamprin  
SHARP MZ-80K
- Bruno Trachsler  
Langackerstrasse 23  
8952 Schlieren
- TRS 80
- Richard Godenne  
Bedastrasse 15  
9000 St.Gallen
- Peter Gasser  
Gasser Elektronik  
Bucheggstrasse 29  
8037 Zürich
- Oliver Tschichold  
Grundackerstrasse 24  
4414 Füllinsdorf
- Urs Schwob  
51, ch. Moise Duboule  
1209 Genf
- Joseph J. Baetschmann  
Rue du Jura 13  
1196 Gland
- Rolf Röthlisberger  
34, rue Daubin, Appt. 71  
1203 Genf
- Roger Widmer  
Blauenstrasse 17  
4153 Reinach
- Roman A. Gubser  
in Bruggen 20  
8907 Wettswil
- Dr. Heinz Beat Winzeler  
Technikum Winterthur  
Postfach  
8401 Winterthur
- Riwos  
Gesellschaft für Wirtschafts-  
beratung mbH  
Marienstrasse 4  
D-8500 Nürnberg 1
- Martin Baur  
Postfach 14  
FL-9491 Nendeln
- DEC PDP II
- Mühlethaler  
Electronics AG  
Falken 100 A  
4511 Rumisberg
- ALPHATRONIC
- Theo Rydmann  
Oberhaide 36  
D-6719 Hettemleidelheim
- NORTH STAR
- Michel Dysli  
Belleroche 3  
2000 Neuenburg
- GA 216/10
- Franz Muheim  
Kirchstrasse  
6454 Flüelen

## COMPUTER-CLUB SÜDTIROL

Kurz vor Redaktionsschluss haben wir noch eine Mitteilung erhalten, wonach in Südtirol (Italien) seit

einem Jahr ein Computer-Club existiert. Dieser Club hat sich ausschließlich dem PET/CBM verschrieben.

Kontaktadresse:  
Georg Stuffer, Oswaldweg 63  
I-39100 Bozen

## WOLLEN SIE IHRE PROGRAMME VERKAUFEN?

Falls Sie interessante Programme erstellt haben, bieten wir Ihnen zwei Möglichkeiten, um diese weiterverkaufen zu können:

1. Sie bieten uns schriftlich Ihr Programm an unter grober Angabe, was es kann. Wir laden Sie dann zu einer Demonstration ein und prüfen Ihr Programm. Darauf vereinbaren wir einen Verkaufspreis und das weitere Vorgehen. Je nachdem erstellen wir noch kundenreife Unterlagen und verkaufen Ihr Produkt aktiv mit oder ohne Einführung beim Kunden durch uns.
2. Sie schreiben einen fertigen Bericht für die Rubrik "Small business". Wenn dieser Beitrag einen gewissen Qualitäts-Standard erreicht, so veröffentlichen wir Ihren Artikel gerne mit Angabe Ihres Namens und Ihrer Adresse, sodass sich Interessenten direkt mit Ihnen in Verbindung setzen können.

Im Fall zwei übernehmen wir keine Qualitätsgarantie für den Käufer, werden aber dennoch darauf achten, dass es sich nicht um "Wunschprogramme" handelt. Das Computersystem spielt dabei keine Rolle.

## COMPUTER VERKÄUFER

Im SCC-Computer-Shop wäre ein "Verkäufer" verloren, denn es gilt die Kundschaft zu beraten und über vieles (wenn auch nie alles) Auskunft geben zu können. Damit unser Team, welches wir im Heft 80-5 vorstellten, seinen Aufgaben genügen kann, benötigen wir nach wie vor einen APPLE-BERATER. Bitte melden Sie sich telefonisch bei unserer Personal-Assistentin, Fräulein E. Zumstein (041-31 45 45) zur Vereinbarung eines unverbindlichen Gespräches bei uns, wenn es Sie interessiert, mehr über die kleinen Computer zu lernen. Eine Karriere mit Zukunft ist Ihnen sicher, wenn Sie über das entsprechende Rüstzeug verfügen und gewillt sind, gefördert und gefördert zu werden.

## Veranstaltungskalender rund um den Computer

# Was Wann Wo?

NCC  
National Computer Conference  
and Show  
4. - 7. Mai 1981  
Chicago

MICRO EXPO 81  
6eme Congres-Exposition  
Microordinateurs  
5. - 7. Mai 1981  
Paris

INTEL  
Internationale Ausstellung für  
Elektrotechnik und Elektronik  
23. - 27. Mai 1981  
Mailand

IMMM/DATACOMM 81  
International Microcomputers,  
Minicomputers, Microprocessors  
23. - 26. Juni 1981  
Genf

THE COMPUTER-FESTIVAL  
Small business and Personal  
Computers  
Seminar and Exposition  
24. - 25. Juni 1981  
Zürich

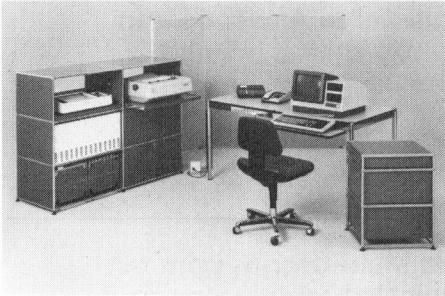
EUROMICRO 81  
Seventh Symposium on  
Microprocessing and  
Microprogramming  
8. - 10. September 1981  
Paris

INELTEC 81  
10. Internationale Fachmesse  
für industrielle Elektronik  
und Elektrotechnik  
8. - 12. September 1981  
Basel

BUEFA 81  
Fachmesse für Bürotechnik  
und Datenverarbeitung  
8. - 12. September 1981  
Zürich

FACHAUSSTELLUNG FUER  
HOBBY ELEKTRONIK  
21. - 25. Oktober 1981  
Stuttgart

COMPEC  
Ausstellung für Computer  
und Peripheriegeräte  
3. - 5. November 1981  
London



## BILDSCHIRM-TISCH MIT AUSZIEHBAREN TASTATURKONSOLE

Das vielseitige USM Büromöbel-System hat, zu seinem ausgesuchten Tischprogramm, nun auch Bildschirm-Tische in verschiedenen Grössen, die mit einer AUSZIEHBAREN TASTATURKONSOLE ausgerüstet sind.

Dank weniger, aber flexibler Grundelemente sind USM Bildschirm-Arbeitsplätze und -Boys ausgesprochene Massarbeit und ermöglichen den Ausbau zum systemintegrierten Arbeitsplatz, der individuellen Anforderungen entsprechend gestaltet werden kann.

RUEEGG-NAEGELI + CIE AG,  
8022 ZUERICH

## WANG ATTACKIERT DEN MARKT FUER GROSS-SYSTEME

Mit gleich drei neuen Modellen der VS-Computer-Serie, dem VS 50, dem VS und dem VS 100, tritt die Wang AG neuerlich auch in eine direkte Konkurrenz mit den sogenannten, vor allen den Markt der Gross-Systeme beherrschenden Mainframern.

Mit der VS-Computer-Serie, die sich durch eine virtuelle Speicher-verwaltung auszeichnet, bietet Wang benutzer-orientierte, aufwärtskompatible Bildschirm-Computer an, die neben Multiprocessing und -programmierung eine für alle angeschlossenen Terminals hard- und softwaremässig einfache Bedienbarkeit garantieren.

Die VS-Computer sind modular aufgebaut, schnittstellen-verträglich und sowohl untereinander wie auch zu Fremdsystemen kompatibel. Dabei ist insbesondere die Kompatibilität mit dem SNA (Systems Network Architecture) der IBM über Prozeduren 3274 und 3277 von Bedeutung.

Die auf dem VS-Computern entwickelten Applikations-Programme - diese können in COBOL, BASIC, RPG II, Assembler oder einer systemab-

hängigen Prozedurensprache abgefasst sein - können auf allen Anlagen dieser Serie ohne Modifikation Verwendung finden. Dabei erfolgt die eigentliche Dateiverwaltung sprachunabhängig. Auch ist auf allen VS-Computern Textverarbeitung und damit eine integrierte Informations-Verarbeitung möglich.

Die VS-Computer-Serie besteht beim Modell VS 50 in der Grundkonfiguration aus einer Zentraleinheit mit 128-KB-Hauptspeicher und einer 1,2-MB-Diskettenstation sowie einer 28-MB-Winchester-Platte. Bereits dieses Modell kann auf 512-KB-Hauptspeicher und maximal 32 Bildschirm-Arbeitsplätze sowie eine Plattenkapazität von 2300-MB aufgerüstet werden.

Die VS 100 ist dann der eigentliche Gross-Rechner innerhalb dieser neuen Computer-Serie. In seiner Grundkonfiguration verfügt dieser Computer über eine Zentraleinheit 256-KB-Hauptspeicher sowie 32-KB-Puffer-Speicher. Die VS 100 kann auf 2-MB-Hauptspeicher, 128 Bildschirm-Arbeitsplätze und maximal 4600-MB-Plattenkapazität aufgerüstet werden. In dieser Grössenordnung richtet sie sich eindeutig an Anwender im Gross-Rechnerbereich.

WANG AG, 8152 GLATTBRUGG

## STARKES INTERESSE FUER "SCHULCOMPUTER"

Tischcomputer gewinnen immer mehr an doppelter Bedeutung: als Lehr- und Lernmittel im Unterricht sowie als unersetzlicher "Mitarbeiter" in der Schulverwaltung. Commodore als Marktführer in diesem EDV-Bereich ist mit seiner breiten Palette von Computer-Systemen (PET 2001 sowie CBM 3001 und 8001) nicht nur an Universitäten vertreten, sondern auch in Realschulen, Gymnasien, höheren technischen Schulen, bestens eingeführt.

An komplizierte Verkehrspläne für optimierte Bahn- oder Flugverbindungen erinnern die handgeschriebenen und gezeichneten Darstellungen des Fachunterrichts in Klassen und Kursen, wie auch die Belegungspläne für Schulräume und -labors. Häufig sind die Wände im Lehrer- oder Konferenzzimmer offensichtlich zu klein, um die riesigen Tafeln aufzunehmen. Verborgen bleibt der grosse Aufwand an Zeit und oft auch an Nervenkraft von Lehrkräften und Schulsekretariat, der in dieser

Unterrichts- und Raumplanung sowie der täglichen Anpassung steckt. Hier erklärt sich das Interesse am "Schulcomputer" von selbst.

## COMPUTER MIT FUEHRUNGS-AUFGABEN

Die Verfeinerung des Angebots, die Differenzierung der Bildungschancen entsprechend den beruflichen Interessen und der Begabung des Schülers, wie sie Ziel der Reformbemühungen waren, haben die Schulsysteme kompliziert gemacht; gegenüber früher ist eine pädagogische Beratung und Führung des Schülers während seiner gesamten Schulzeit unumgänglich. Voraussetzung dafür ist - bei allem Unbehagen - die "bürokratische" Führung in der Schülerdatei der Schulverwaltung. Nur der Computer kann sicherstellen, dass trotz Zeitmangels der Lehrkräfte und der kaum noch zu überblickenden Vielfalt von Kursen, Kombinationen und Bildungswegen eine für jeden einzelnen Schüler "massgerechte" Optimierung seines Lern- und Ausbildungsprogramms erreicht wird. Stundenpläne und Schülerkarteien herkömmlicher Art werden den qualifizierten Anforderungen an die Schulverwaltung nicht mehr gerecht.

Die Bedeutung des Computers im Unterrichtsraum und Schullabor bedarf kaum einer Erklärung. In dem Masse, in dem Computer wichtige Funktionen im öffentlichen, beruflichen und privaten Bereich übernehmen, muss schon in der Schule Kenntnis über ihren Aufbau und ihre Handhabung, über Möglichkeiten und Grenzen sowie über ihre Wirkungen auf allen Gebieten vermittelt werden - nicht nur in Fächern wie z.B. "Computerrechnen" und "Informatik".

Die Tischcomputer der neuen Generation können innerhalb der Schule bewegt und überall aufgestellt werden. Sie werden auch an Universitäten und Hochschulen verwendet, die über einen Grossrechner mit verschiedenen Terminals verfügen. Die Systeme, zu denen nebst der Zentraleinheit mit Bildschirm zusätzliche Speicher (Floppy Disk oder Datenkassette) und Drucker gehören, bedienen sich überwiegend der einfachen Programmiersprache Basic/Pascal und sind problemlos im Dialog zu handhaben. Auch wenn praxiserrechte und erprobte Programme vorhanden sind, erlaubt Basic/Pascal Variationen bzw. das Schreiben eigener Programme.

COMMODORE AG, 4010 BASEL

## Microcomputersystem für das fortschrittliche Unternehmen



CROMEMCO Dialog-Computersystem für alle kommerziellen Anwendungen wie:

- Textverarbeitung
- Fakturierung
- Buchhaltung
- Lagerbewirtschaftung
- Spezialprogramme

Ausbaubar als Multi-User-System mit bis zu 7 Terminals beziehungsweise 7 Benutzern.

Speichererweiterung mit Floppy Disk oder Magnetplatten bis zu 22 Mio Zeichen.

Verlangen Sie unverbindlich Unterlagen.

**COMICRO AG**

CH-8045 Zürich, Eichstrasse 24, Tel. (01) 66 04 66  
Telex 58738 micom ch, Telegramm micom

**AMERA**  
ELECTRONICS AG

Die Matrix-Printer der 80er Jahre

**TermiNet 2000** von  
**GENERAL ELECTRIC**



- Kompaktes, elegantes Styling ● mit oder ohne Tastatur
- 10 kg leicht, 58 dBa leise ● 30 oder 120 Zeichen/sec
- 7x9 Matrix: **Echte Unterlängen!** ● RS 232 C ● 1000 Zeichen Datenpuffer ● Horizontal- und Vertikalformatierung
- Anwerback ● **batteriegepufferter Konfigurationsspeicher.**

Sichern Sie sich die Unterstützung des Amera-Ingenieurteams.

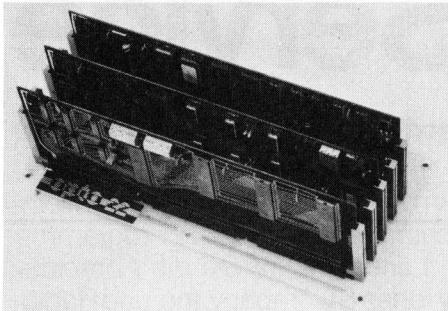
**AMERA**  
ELECTRONICS AG

8046 Zürich, Switzerland, Telefon: 01/57 11 12, Telex: 59837

# Offsetdruck Buchdruck Buchbinderei



Unionsdruckerei AG Luzern  
Kellerstrasse 6, 6005 Luzern  
Telefon 041 - 44 24 44



DAS NEUE I/O-KONTROLLSYSTEM  
SAMUX II VON OPTO 22

SAMUX II ist ein seriell adressierbarer Multiplexer zur Steuerung peripherer Leistungen über die serielle Schnittstelle eines jeden Computers, Prozessors etc.

Jedes SAMUX ist individuell bis max. 64 SAMUX II Stationen adressierbar, verbunden mit dem Steuerprozessor über eine asynchrone, full duplex serielle 4-Drahtleitung. Dadurch kann eine serielle Schnittstelle eines Steuerprozessors bis zu 64 Multiplexstationen ansprechen. Jede der max. 64 SAMUX II Stationen kann im Vollausbau 192 Leistungs-I/Os steuern.

Output: 220 V ~ / 3 A  
60 V - / 3 A  
200 V - / 1 A  
Input: 220 V ~ / 6 bis 60 mA  
10-32 V - / 32 mA

SAMUX II ist ein auf Mikroprozessor basierendes System und verfügt gleichfalls über A/D- und D/A-Wandler-Eingänge.

## VEROEFFENTLICHUNGEN IN DEN NEWS...NEWS...

Die Publikation von Pressemitteilungen über neue oder wesentlich verbesserte Produkte in den News...News... erfolgt kostenlos. Diese Dienstleistung ist in erster Linie den Inserenten vorbehalten. Für vorgesehene Abbildungen sind Fotos oder Reinzeichnungen erforderlich, von denen Fotolithos zu Selbstkosten zu Lasten des Einsenders angefertigt werden. Ueber die Auswahl der Texte und Bilder sowie deren Präsentation entscheidet die Redaktion.

VERLAG SCC AG

Bei einem Ausbau mit 96 Leistungs-I/Os, anstelle von max. 192, kann SAMUX II beispielsweise gleichzeitig die Steuerung von 48 Kanälen eines 12 Bit A/D-Wandlers und 8 Kanäle eines 8 Bit D/A-Wandlers übernehmen.

SAMUX II besteht aus einer Systemgrundplatte und dem Systemprozessorboard. Vier freie Steckplätze ermöglichen es dem Anwender, das System seinen Anforderungen entsprechend zu bestücken.

Treiber-Board DB-1:  
Steuerung von 48 I/O-Moduln  
(optisch entkoppelt 4 kV)  
A/D-Wandler Board ADC-1:  
Steuerung von max. 48 Analog-  
Eingängen  
D/A-Wandler Board DAC-1:  
Steuerung von 8 Analog-Ausgängen

METRONIK GMBH,  
D-8025 UNTERHACHING

## EIN AUSSERGEWOEHNLICHES BUCH UEBER EINEN AUSSERGEWOEHN- LICHEN PROZESSOR

Die 16-bit-CPU Z8000 hat erheblich die Gemüter der Mikroprozessoranwender erregt, da sie sich im technologischen Grenzbereich bewegt.

Detaillierte und fundierte Auskunft über Aufbau und Anwendung war für die Allgemeinheit bisher aber nicht in Erfahrung zu bringen. Dieses Buch - als erstes seiner Art in deutscher Sprache - ändert den Informationsmangel grundlegend.

Es bietet auf über 450 Seiten eine elementare Darstellung dieses aussergewöhnlichen Computer-Bauteils. Dies bezieht sich auf den grundlegenden Aufbau und die Funktion (hard- und softwaremässig) der CPU. Die sehr umfangreiche Hardware-Darstellung gipfelt in der Beschreibung eines kapitalen Parallelrechnersystems. Wer eine Information mit 16-bit-Bereich generell sucht und seinen extremsten Vertreter kennenlernen will, kommt an diesem Buch nicht vorbei.

Und das sind die Hauptthemen:

- Einführung
- Die Einchip-16-bit-Zentraleinheit Z8000
- Adressiersprachen und Befehlsatz
- Hardwarekonfigurationen für beide CPU-Versionen

- Der Speicherorganisations-Baustein Z8010 MMU, ein Kernstein des Z8000-Systems
- Assembler-Programmietechnik mit den Z8000-Befehlen
- Ein Multiprozessor-Konzept mit Z8000-Bausteinen
- Z8000 - und wer noch?
- Bestehende Z8000-Systeme
- Das Z8000-System für die 80er Jahre

TE-WI VERLAG GMBH,  
D-8000 MUENCHEN 70

## 16-BIT MIKRO- COMPUTERSYSTEM ONIX

Das zur Zeit erfolgreichste und leistungsfähigste 16-Bit Mikrocomputersystem C 8002 basiert auf der Z8002 Zentraleinheit. Der Schreib-/Lesespeicher kann wahlweise 256 KB bzw. 512 KB sein. Die Adressierung geschieht mittels einer schnellen Bankswitching-Logik.

Weitere wichtige Merkmale sind DMA, Memory Manager und 64-Bit Floating-point-Prozessor. Als Massenspeicher sind vorgesehen:

- wahlweise 10, 18 oder 40 MBit 8 Zoll Winchester Festplatte
- 12 MB Magnetband für back-up

Bis maximal 8 Anwender können an ein System angeschlossen werden. Jeder Anwender kann sein Anwenderprogramm vor unerlaubten Zugriffen schützen, mittels eines eigenen Passwortes.

Der Druckeranschluss ist eine Standard (Centronics) Parallel-Schnittstelle. 2 serielle Ein-/Ausgänge für Datenkommunikation erlauben den Systemanschluss an ein DFU-Netzwerk. Das multitaskfähige Multiuser Betriebssystem ONIX erlaubt, wie vorher erwähnt, gleichzeitig 8 Anwender unabhängige Tasks ablaufen zu lassen.

Das ONIX Betriebssystem ist eine abgewandelte Version von Western Electric's UNIX-Version 7. Dieses weitverbreitete Betriebssystem besitzt einen sehr leistungsfähigen Editor mit Steuerzeichen für Textverarbeitung.

Als höhere Programmiersprachen sind BASIC, RM COBOL, C und PASCAL (UCSD) lieferbar. FORTRAN ist in Vorbereitung.

KONTRON MIKROCOMPUTER GMBH,  
D-8057 ECHING/MUENCHEN

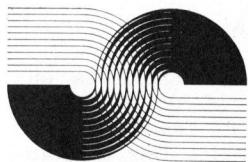
# Dow Jones News/Retrieval Service

● Stock Quote Reporter ● Historical Stock Quote ● News Retrieval ● Disclosure Online

Tagtäglich, rund um die Uhr, erfasst Dow Jones weltweit Tausende von Informationen in ihren Datenbanken. Wir ermöglichen Ihnen ab sofort den uneingeschränkten Zugriff zu diesen Daten über: Politik, Finanzwelt, Wirtschaft, Regierungskreise, Angaben spezieller Branchen, ausführliche Firmenprofile, Entscheidungen, Trends, Kurse – kurz, einfach alles was Sie interessieren könnte ist für Sie problemlos zugänglich.

Wir liefern Ihnen einen betriebsbereiten APPLE II-Microcomputer mit Bildschirm (inkl. sämtlicher Programme und Passwörter) und erledigen für Sie die Anmeldeformalitäten bei Dow Jones & Company, Inc. und Radio-Schweiz AG, die für die Telekommunikations-Verbindungen verantwortlich ist (DATAC-Dienst).

**Besuchen Sie uns vom 9.-11. April an der Micro-Comp 81, Stand Nr. 18.**



radio-schweiz ag  
telekommunikation +  
flugsicherung  
3000 Bern 14

pro data

aktiengesellschaft pro data  
abteilung micro-computer  
industriestrasse 30  
8302 kloten 01/814 31 60

- Bitte senden Sie uns Detailunterlagen  
 Wir wünschen eine unverbindliche Vorführung

Firma: \_\_\_\_\_

Sachbearbeiter: \_\_\_\_\_

Strasse: \_\_\_\_\_

PLZ/Ort: \_\_\_\_\_

Telefon: \_\_\_\_\_

## «AUTO-RANGE»



### Messen ohne Umschalten

LCD-Digital-Anzeige  
Automatische Bereichswahl  
10 A – Strombereich DC und AC  
Praktisch, robust, preiswert  
Fr. 189.– inkl. Messkabel mit Klemmen

Verlangen Sie Datenblatt oder  
Mustergerät auf Probe

**SEYFFER+CO. AG**  
**CH-8048 ZÜRICH**  
Abt. Industrie-Elektronik Telefon 01/62 82 00

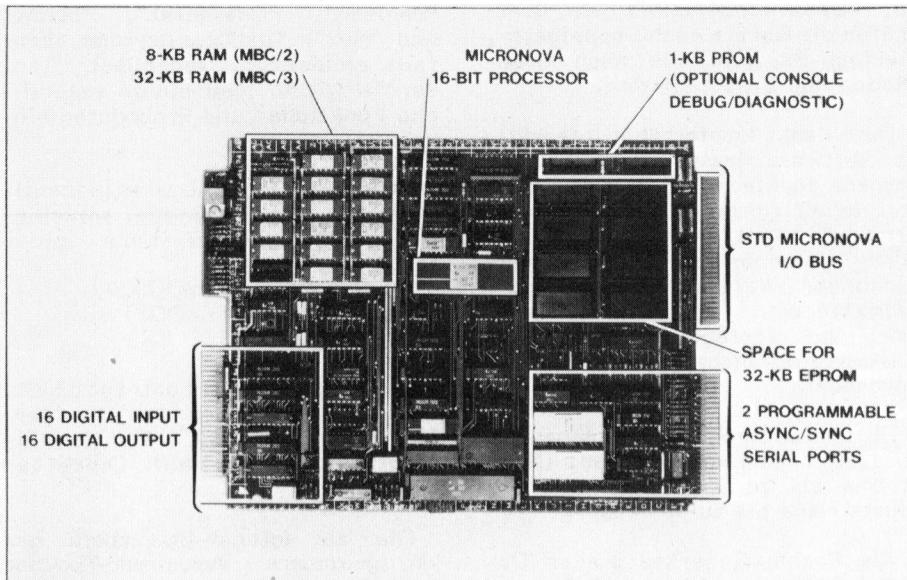
## Wir sind im Bild über Bildschirm-Arbeitsplätze

Verlangen Sie mit diesem Inserat Informationen über das RN Angebot an vielfältigen und anpassungsfähigen Tischen für Terminals.



**Rüegg-Naegeli**

8022 Zürich Beethovenstrasse 49  
Telefon 01-201 22 11



PLATINEN-COMPUTER MICRONOVA MBC/2.../3

Die Platinen-Computer microNOVA MBC/2 und MBC/3 von Data General unterscheiden sich durch den RAM-Speicher der MBC/2 mit 8 KB und den 32 KB ausgelegten RAM-Speicher der MBC/3. Sowohl die MBC/2 als auch die MBC/3 haben Anschlüsse für einen PROM-Speicher bis 1 KB und einen EPROM-Speicher bis 32 KB. Beide Platinen bieten ausserdem zwei separat programmierbare asynchrone/synchrone Kommunikationsschnittstellen und 16 digitale Eingabe- und 16 digitale Ausgabe-Kanäle.

SYCON ELECTRONIC GMBH, D-8011 PUTZBRUNN

## FOTOSATZ IM BUERO - EXKLUSIVE MOEGlichkeiten DES SYSTEMS BG 1000

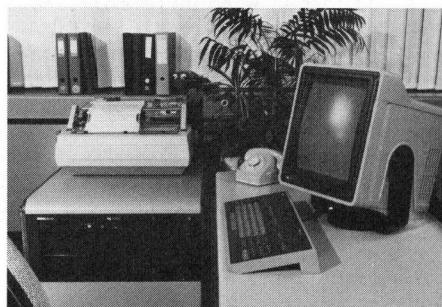
Bobst Graphic hat sich dank der Qualität der von ihr entwickelten und vertriebenen Fotosatzanlagen auf der ganzen Welt einen hervorragenden Namen geschaffen und schickt sich nun an, das offene Feld zwischen Büro und Fotosatz ebenfalls zu bearbeiten.

Als Ergebnis sorgfältiger Marktuntersuchungen bietet die BG 1000 eine Reihe von Eigenschaften, welche Texte und Daten von der Erfassung bis zum Ausdrucken voll meistert. Damit ermöglicht das System BG 1000 nicht nur ein sauberes Formatieren, sondern auch ein beliebiges Umstellen, Sortieren und Mischen mit anderen Daten. Des Weiteren kann man je nach Bedarf vervielfachen, archivieren, in Fotosatz umsetzen oder sogar über die Telefonleitung an eine andere Verarbeitungsstelle übermitteln.

Augenfällig ist vor allem die spielend leichte Bedienung des Gerätes auch für eine Schreibkraft nach kurzer Einführungszeit. Die BG

1000 ist für hohe Leistung ausgelegt und umfasst eine vollständig interaktive Dialog-Software. Dazu wird jede Verarbeitungsstufe durch genaue Meldungen gesteuert, welche nacheinander am Bildschirm erscheinen. Das Endergebnis dieser Operationen ist am Video-Bildschirm sofort ersichtbar und stimmt mit den endgültigen Dokumenten überein.

Zum Drucken, Uebermitteln oder Sortieren sind sämtliche Verarbeitungsmoduln dank einem "Menu" am Bildschirm unmittelbar zugriffsbereit. Während des ganzen Verfahrens sind keine Magnetplatten auszuwechseln. Die BG 1000 ist mit einer nach Wunsch aufstellbaren, bild-



schirmunabhängigen Tastatur ausgerüstet. Jede Funktion besitzt ihre eigene, visuell klar unterscheidbare Taste mit der entsprechenden Software-Beschriftung. Am Bildschirm selbst ist ein Textvolumen einer hochstehenden A4 Seite mit einem Blick erfassbar.

Im bezug auf den technischen Aufbau umfasst die BG 1000 zwei Halter für je eine 1,2 MBytes Platte, was einer Million Zeichen bzw. 300 Seiten A4 Format entspricht, sowie einen Typenrad-Schnelldrucker mit einer Leistung von 60 Zeichen pro Sekunde. Der dynamische RAM Speicher mit 128 K ist ein weiterer Garant für die hervorragende Leistungsfähigkeit der BG 1000. Ausserdem ist ein Standard-Interface RS232 vorhanden.

Die BG 1000 kann an eine unabhängige Fotosatzmaschine angeschlossen werden. Dies ist eine Exklusivität von Bobst Graphic und ermöglicht den Unternehmungen mit betriebseigener Druckerei das Ausdrucken von auf Platten gespeicherten Texten. In den meisten Fällen ergibt sich jedoch für den Anwender mindestens die Möglichkeit, einer Druckerei druckreife Texte zu übergeben, ohne dass diese typografisch noch weiterbehandelt werden müssen.

Hier liegt eine umwälzende Neuheit, welche sich auf die Qualität der Manuskriptbearbeitung günstig auswirken wird und auf dem neugeschaffenen COMPO-Modul beruht. Heute ist der Fotosatz bereits mit normalem Büropersonal, insbesondere Sekretärinnen, einsetzbar, indem mit einer sehr einfachen Sprache die gewünschte Textform in Fotosatz erzeugt werden kann. Mithin berücksichtigen lassen sich dabei Schriftgrad und -art sowie Silbentrennung und vertikale und horizontale Linien.

Als zweite Besonderheit bietet die BG 1000 die Möglichkeit, auf ihrem Standard-Schnelldrucker Kontrollabzüge herzustellen. Diese sind eigentliche Faksimile Drucke in natürlicher Grösse des endgültigen Fotosatzes.

Ein einziges Gerät, eine einzige Texterfassung, durch eine einzige Person bedient sowie voll beherrschte Verarbeitung des Schriftstücks bis zum Druck sind nur einige Vorzüge, welche Ihnen die neu konzipierte BG 1000 zu bieten vermag.

BOBST GRAPHIC, 1001 LAUSANNE

# NEWS... NEWS...

## DYNAMISCHER SPEICHERSTEUERBAUSTEIN FUER 16K UND 64K RAMS

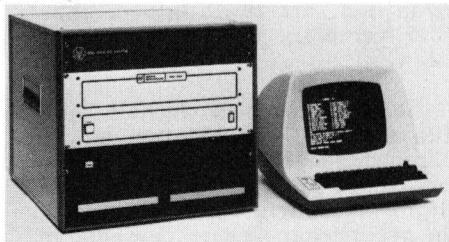
Advanced Micro Devices hat den dynamischen Speichersteuer-Baustein (DMC) Am2964B vorgestellt. Dieser bipolare LSI-Baustein beinhaltet Zeilen-Spalten-Adressumschaltung, eine Refresh-Adresserzeugung und die Ausgabe von RAS/CAS zur einfachen Steuerung umfangreicher dynamischer Speicherwerke mit 16K oder 64K RAMs. Er ersetzt eine Reihe von schnellen Schottky (LS) MSI Bausteinen kleiner Größe, die bisher diese Steuerfunktionen übernahmen. Ueberdies bietet er zusätzliche Speicherzugriffszeit, in dem der Zeitverzug zwischen kritischen Adressen sowie RAS und CAS-Steuersignalen eliminiert wird.

Der Am2964B ist das letzte Glied der Am2960-Familie von AMDs Produkten zur Unterstützung von dynamischen Speichern. Diese Bausteinserie liefert die vollständige Lösung zur Entwicklung von dynamischen Speichersystemen und beinhaltet:

Die Fehlererkennungs- und Korrekturereinheit (EDC) Am2960, dynamische RAM-Treiber, die eben vorgestellten Am2964B und die bald erhältlichen Mehrfach-Bus-EDC-Buffer. Eine AmZ-8160-Serie basiert auf der gleichen Bausteinentwicklung, jedoch mit für Mikrocomputer-Systeme charakteristischen Parametern.

Der Am2964B DMC und sein Äquivalent, der AmZ8164B sind in 40 Pin-DIPs ab Lager lieferbar.

AMD, D-8000 MUENCHEN 80



## AUTOMATISCHES PRUEFSYSTEM FUER FET TRANSISTOREN

Speziell für das Testen von FET Transistoren inklusive JFETs, MOSFETs, VMOS, DMOS, HEXFETs, SIPMOS, GaAsFETs, N- oder P-Kanal Anreicherungs- oder Verarmungstypen bietet die SYNTEL GMBH geeignete mikroprozessorgesteuerte Systeme

an. Neben Einzelhalbleitern überprüfen die Geräte auch Doppelgate-, vierfach-Bauteile wie auch Gunnedioden und Impatt Dioden.

Durch sehr komfortable interaktive Software lassen sich Testprogramme in kürzester Zeit erstellen und modifizieren. Ausser für den Produktionstest stehen vielfältige Softwaremodule für Bauteiluntersuchungen, Waferapplikationen, zum Selektieren, für Burn-In Tests usw. zur Verfügung. Statistische Auswertprogramme sind ebenfalls vorhanden.

Durchbruchspannungen können bis zu 1000 V gemessen werden. Gateströme bis zu 1 pico Ampere und Flusströme bis zu 30 Ampere.

Als Peripheriegeräte stehen Datensichtgerät, Floppy Disk und Drucker zur Verfügung. Handler und Prober Interface sind ebenfalls lieferbar. Durch den Anschluss spezieller Remote-Teststationen kann das zentrale Prüfsystem mehrere Testplätze steuern.

SYNTEL GMBH,  
D-8025 UNTERHACHING

## ECHTER 6809 PASCAL-COMPILER

Die amerikanische Software-Firma "Omeegasoft" bietet nun einen echten PASCAL-Compiler für den Motorola-6809-Prozessor an. Dieser Compiler ist für Entwicklungssysteme der Typen "Exorciser", "Exorset" und "SWT-6809" lieferbar.

Bei der Programmentwicklung stehen dem Anwender zwei Wege offen: Ein interaktiver symbolischer Debugger unterstützt die Fehlersuche, während für die Generierung des Objekt-Moduls ein relozierender Assembler und ein Link-Modul zur Verfügung stehen.

Die produzierten Binärprogramme sind "ROMable", "reentrant" und Positions-unabhängig (!). Die Runtime-Bibliothek wird im Quellen-Code mitgeliefert. Separat kompilierte PASCAL-Prozeduren sowie Assembler-Routinen können zusammengefügt werden. Der Compiler erzeugt Output in 6809-Assemblersprache, sodass "Patches" von geübten Programmieren vorgenommen werden können.

Das Omeegasoft-PASCAL-System eignet sich vor allem für industrielle Anwendungen. Es unterstützt die Variablentypen "Integer", "Real",

"Boolean", "Character", "String" und "Hex". Textfiles werden ebenfalls ermöglicht. Gegenüber "Standard"-PASCAL sind einige zusätzliche Funktionen und Prozeduren eingebaut.

Omeegasoft-PASCAL wird in identischer Ausführung für drei verschiedene Entwicklungssysteme geliefert:

- Exorciser (MDOS09)
- Exorset (XDOS)
- SWT-6809 (FLEX)

Der Speicherbedarf beträgt 32 KB. Compiler, relozierender Assembler, Linking Loader sowie Runtime-Bibliothek werden auf Disketten geliefert.

Die als Integer-Benchmark bekanntgewordene Ackermann-Funktion F (3,4) benötigt unter Omeegasoft-PASCAL nur ca. 2 sec Rechenzeit (2 MHz-6809).

DIGICOMP AG, 8004 ZUERICH



## ZUSAMMENARBEIT KONTRON UND ADDS

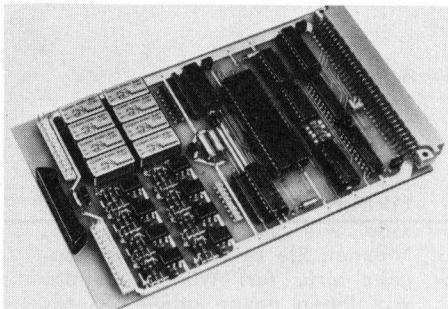
Entgegen kürzlich verbreiteten Meldungen legt KONTRON AG DATASYSTEMS Wert darauf, dass sie nach wie vor die Vertretung der Bildschirmterminals der Firma ADDS (Applied Digital Data Systems Inc. Hauppauge, USA) inne hat.

KONTRON AG DATASYSTEMS vertreibt die ADDS Bildschirmterminals seit über 5 Jahren und verfügt deshalb über ein in der Schweiz wohl einmaliges know-how auf diesen Geräten.

Das angebotene Produktprogramm umfasst die Geräte der REGENT-Serie, die Modelle REGENT 20, 25, 30, 40, 60, 100 und 200 sowie das Modell CONSUL 980 und die Bildschirmterminal-Controller in den Industrieversionen MRD 380 und MRD 980.

KONTRON AG DATASYSTEMS,  
8048 ZUERICH

# NEWS... NEWS...



## EUROLOG - ECB BUS-KOMPATIBLE DIGITAL-INTERFACE-KARTE

Bitronic stellt eine neue Platine aus dem umfangreichen uP-System EUROLOG (Z 80 und Z 80 A) vor. Diese industriemässig ausgelegte Digital-Interface-Karte EML/DIC 88 kann überall dort eingesetzt werden, wo Spannungen im Bereich von 12 bis 60 V benötigt werden und wo eine galvanische Trennung zwischen den individuell zu beschaltenden Ein- und Ausgangskreisen erforderlich ist. Typische Einsatzfälle sind dort zu finden, wo logische Eingangssignale von Kontroll- oder Messsystemen Motoren oder Transportsysteme steuern.

### Technische Daten:

- 8 isolierte digitale Eingangskanäle (optoentkoppelt 4,4 kV)
- 8 isolierte digitale Ausgangskanäle (Relais, 60 V, 1 A AC und DC)
- Eingangskanäle mit RC-Filterung (Störspannungsunterdrückung)
- Eingangsspannungsbereich einstellbar und überspannungsgeschützt (12 V DC, 24 V DC, 48 V DC)
- LED-Zustandsanzeige der Ein- und Ausgänge
- Volldecodierbarer I/O-Adressbereich mittels DIL-Schalter
- ECB-Bus-kompatibel
- Europakartenformat mit +5 V Spannungsversorgung
- 48 Std.-Burn-in getestet
- Beschreibung in deutsch und Testsoftware
- Weitere EUROLOG-Karten ab Lager lieferbar: CPU, PIO, SIO usw.

BITRONIC GMBH,  
D-8000 MUENCHEN 80

## THE CINCINNATI KID...EIN FASZINIERENDES SPIELPROGRAMM

Hatten Sie schon mal Lust eine Runde POKER zu spielen, fanden aber keinen Mitspieler? Falls Sie Besitzer eines PET-CBMs sind, brauchen

Sie nicht mehr weiter zu suchen - der willige Mitspieler befindet sich schon bei Ihnen.

Das Ziel dieser Pokervariante, STUD-POKER, ist den Spieleinsatz, die Wette, zu gewinnen. Dies wird erreicht indem man wertmässig eine bessere Karten-Kombination hat als der Gegner. Man spielt mit 52 Karten und den vier Farben Kreuz, Herz, Karo und Pik. CBMs grafische Kapazität wurde voll ausgenutzt um die Karten originalgetreu darzustellen.

Der Computer mischt und teilt 5 Karten aus, die erste mit der Bildseite nach unten. Eine komplizierte Programm-Routine macht aus der Maschine einen fähigen Poker-Spieler: Wird er die Wette eröffnen? Oder sogar erhöhen? Will er abwarten bis zur nächsten Wettrunde? Oder sogar ganz aussteigen? - Aufgepasst, er kann auch bluffen!

Erhältlich in Deutsch, Französisch oder Englisch. Beachten Sie dazu bitte auch das Inserat in dieser Ausgabe.

MICRO-SOLUTIONS,  
1232 CONFIGNON

## EIN NEUER MASSSTAB IN DER RADDRUCKERKLASSE

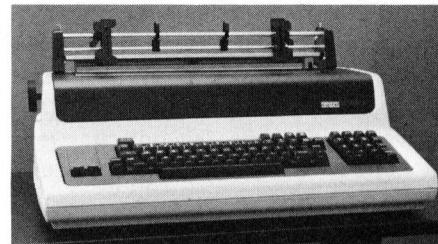
Qualität und Leistung zu einem Preis, welche dem Anwender die Alternative in der Dateneingabe und Datenausgabe bietet.

Überall dort, wo das gestochene, scharfe Schriftbild bei hoher Zuverlässigkeit das ausschlaggebende Kriterium ist, findet das Schreibwerk sein Einsatzgebiet. Die mikroprozessor gesteuerte Elektronik reduziert die Mechanik auf ein Minimum und gibt dem Schreibwerk eine überdurchschnittliche Flexibilität und Vielseitigkeit.

Die Ansteuerbaren von OLYMPIA mit und ohne Tastatur. ESW-100-KSR, das Schreibwerk, das gleichzeitig als Top-Schreibmaschine Ihre Korrespondenz erledigt, ESW-100-RO, das anpassungsfähige Ausgabeschreibwerk mit der umfassenden Basis Ausstattung.

### Technische Daten:

- Druckprinzip:  
Typenrad auswechselbar,  
96 Zeichen
- Druckgeschwindigkeit:  
17 Zeichen pro Sekunde
- Zeilenlänge:  
115 Zeichen bei 1/10" Teilung  
139 Zeichen bei 1/12" Teilung



### Schnittstelle:

- EIA RS-232-C, CCITT V-24 und TTY-current loop- 20 mA
- Uebertragungsart: asynchron
- Uebertragungsweg: maximal 15 m
- Uebertragungs-Geschwindigkeit: 150 Baud

OLYMPIA DATENTECHNIK,  
8153 RUEMLANG

## LA34-AA UND LA34-DA

Mit dieser neuen Terminalreihe hat DEC erstmals Terminals vom Standfüss (Tisch) weggenommen. Der Benutzer kann die "Tabletop Terminals" überall hinstellen, auf einen gewöhnlichen Tisch oder ein Pult (auch zu Hause) oder aber, als direkter LA36-Ersatz, auf einen Terminalstandfüss, der als Option von Digital Equipment bezogen werden kann.

Obwohl nur 10 kg schwer, haben die neuen MATRIXDRUCKER mit einer ausgezeichneten Schriftqualität nichts an Funktionalität und Qualität eingebüsst. Im Gegenteil, die Funktionalität wurde gegenüber dem LA36 noch wesentlich erhöht:

- Attraktives Schreibmaschinen-Styling
- Keine Einschränkungen bei der Druckpapierverwendung, 7,6 cm bis 37,8 cm breit, randgelocht oder Einzelblatt
- 30 cps Schreibgeschwindigkeit (mit 45 cps Aufholmode)
- 7x9 Punktmatrix
- 4/8 verschiedene Schriftbreiten mit Titelschrift
- 6 verschiedene Zeilenschaltungen mit Walzenrücklauf (Plotten)
- Funktionen und Fehlermeldungen über LSI-Mikroprozessor

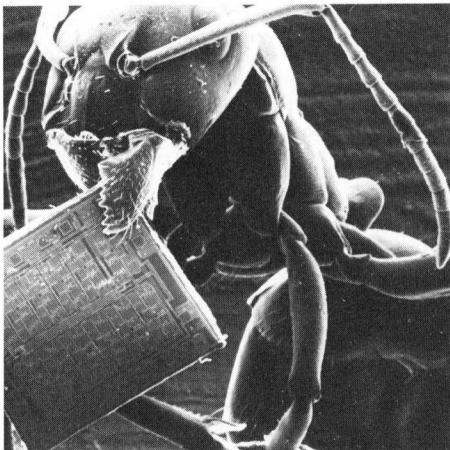
Die Qualität entspricht dem sogenannten Industrie-Standard:

- 2800 Stunden MTBF Zeit (entspricht höchstens einer Störung pro Jahr).
- 100 Millionen Zeichen Schreibkopf.
- 24 Stunden Dauertest bei jedem, das Werk verlassende Terminal.

DIGITAL EQUIPMENT  
CORPORATION SA, 8302 KLOTEN

# Vorschau

Diese Vorschau beginnen wir für einmal nicht mit dem Ausblick auf die nächste Ausgabe sondern befassen uns kurz mit unserem aktuellen Titelbild, das uns freundlicherweise von der Philips AG zur Verfügung gestellt wurde.



Das Ungeheuer entstammt nicht etwa einem Science-Fiction-Film sondern ist eine mit einem Philips Elektronen Mikroskop fotografierte Ameise, welche in ihren Mundwerkzeugen einen Halbleiterchip hält. Beachtenswert ist der Grössenvergleich zwischen dem winzig wirkenden Mikroprozessorchip, dem Herz eines Kleincomputers, und dem massigen Ameisenkopf.

In unserer Nummer 81-3 berichten wir nebst vielen anderen, wie immer interessanten Beiträgen, über das Eumig-Kassettengerät FL-1000. Nicht dass wir uns plötzlich vom Computerfan zum Tonjäger gewandelt hätten, aber das FL-1000 stellt eine sehr gelungene Symbiose zwischen Computer und Audiogerät dar. Und das allerschönste daran, sämtliche Funktionen des FL-1000 können von einem Kleincomputer ferngesteuert und programmiert werden.

Eine kommerzielle Radiostation in Amerika steuert bereits ihr gesamtes Radioprogramm mit einem CBM/PET und 8 Eumig-Geräten. Wir werden das Gerät sowie das Steuerprogramm für einen CBM/PET und eine in der Praxis bewährte Applikation vorstellen.

Die bekannte Programmiersprache PASCAL setzt sich immer mehr auch für den Kleincomputer durch. Verschiedentlich planten wir schon den Abdruck eines PASCAL-Lehrganges; da solche Lehrgänge aber schon in grosser Zahl vorhanden sind, wollen wir davon absehen und den zur Verfügung stehenden Raum besser nutzen. Um eine Programmiersprache zu erlernen, sollte man nicht nur einen Kurs absolvieren, sondern

WOLLEN SIE FUER  
UNSERE FACHZEITSCHRIFT  
SCHREIBEN?

An fachlich lehrreichen Artikeln von freien Autoren sind wir immer interessiert. Nehmen Sie bitte mit uns Kontakt auf. Auf Wunsch senden wir Ihnen gerne unsere Autorenanweisungen.

Beiträge, die wir nach sorgfältiger Prüfung abdrucken, honorieren wir angemessen. Legen Sie bitte Ihren Artikeln die notwendigen Diagramme, Zeichnungen und Listings (inklusive Kassette oder Diskette) bei.

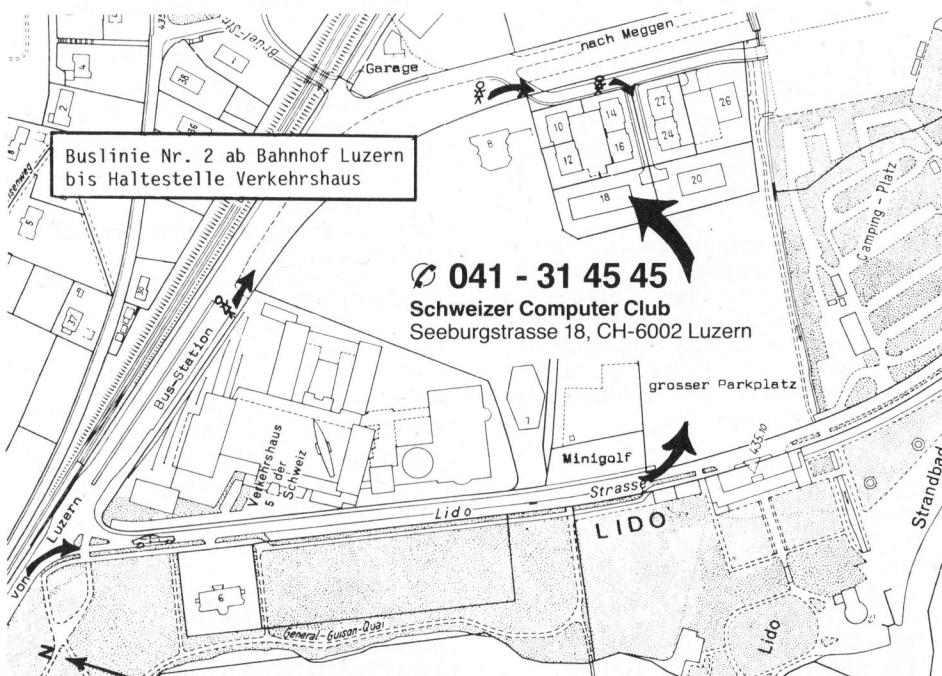
VERLAG SCC AG

vor allem mit dieser Sprache praktisch arbeiten. Sie finden deshalb in der nächsten Ausgabe wieder ein in Pascal geschriebenes Spielprogramm welches sich zum Eintippen, Spielen und Analysieren eignet. Trotzdem sind wir selbstverständlich bei genügend grossen Nachfrage gerne zum Abdruck eines PASCAL-Grundkurses bereit. Schreiben Sie uns doch Ihre Vorschläge zu diesem Thema.

Die Fortsetzung des Artikels Eprom-Programmierung mit dem TM-990 die in dieser Nummer aus technischen Gründen ausfallen musste, folgt ebenfalls im nächsten Heft.

Kleincomputer werden nicht nur für Hobbyanwendungen eingesetzt, sondern stehen je länger desto mehr auch in der Industrie im Einsatz. Ueber eine solche Applikation werden wir Ihnen ebenfalls im nächsten Heft berichten.

Die nächste Ausgabe von MIKRO-UND KLEINCOMPUTER erhalten Sie ca. um den 6. Juni herum. Und Sie wissen ja, im Abonnement ohne Umwege direkt an Ihre Postanschrift. Haben Sie übrigens Ihre Abonnements-Erneuerung schon einbezahlt? Die Postcheck-Nummer steht im Impressum auf Seite 3.



Ich/Wir bestelle(n) hiermit  Ich/Wir habe(n) den Betrag auf Ihr **PC 60 - 26496** einbezahlt

Anzahl	Artikel Nr.	Bezeichnung	Betrag Fr.

**Privatmitgliedschaft**  SFr. 56.-  Ausland SFr. 64.-/DM 71.-  
 **Firmenmitgliedschaft**  SFr. 86.-  Ausland SFr. 94.-/DM 105.-  
 Porto und Verpackung für Kleinartikel (**Systeme Fr. 20.-**) Fr. 3.-  
Total Fr.

Unterschrift \_\_\_\_\_ Ort und Datum \_\_\_\_\_

Genauere Adresse auf der Rückseite (Telefon G/P \_\_\_\_\_ )

Bitte geben Sie Ihre Telefon-Nummer an, damit wir Sie für eventuelle Rückfragen sofort erreichen können.

Auch Sie können Mitglied werden im SCC. **Einmalige** Eintrittsgebühr für Private Fr. 20.- (für Firmen Fr. 50.-) **plus** Jahresabo **m+k computer** (6 Ausgaben) Fr. 36.-, also erstmals Fr. 56.- (Fr. 86.-). Für das Ausland inkl. Porto erstmals Fr. 64.-/DM 71.- (Fr. 94.-/DM 105.-)

Die DCT (Dialog Computer Treuhand AG, Luzern) verfügt über 70 EDV-Fachleute und eigene Schulungsräume. Lassen Sie sich vor dem Kauf eines Kleinsystemes objektiv beraten. Informieren Sie sich vor einem (eventuell falschen) Schritt, denn jedes System hat seine Besonderheiten!

## Small Business-Info Karte

Was tun die Kleincomputer? Senden Sie mir Informationen über «Small Business»

Ich besitze bereits ein System \_\_\_\_\_ und würde gelegentlich gerne andere gleichartige Anwender kennenlernen.

Für mich käme ein Kleincomputer für folgenden Einsatz in Frage:

- Fakturierung / Auftragsabwicklung
- Lagerbuchhaltung
- Finanzbuchhaltung / Debi / Kredi
- Adressierung / Textverarbeitung
- Andere: \_\_\_\_\_

Die Lösung darf kosten:  bis 8000.-  8-15000.-  15-25000.-  25-35000.-  
 35-50000.-  darüber, da Mehrplatz-System.

Bin an einer Demonstration in Luzern interessiert – geben Sie uns Ihren Vorfahrtstermin bekannt.

Bin an Programmierkursen für  kaufm.  techn. Anwendung interessiert.

Genauere Adresse auf der Rückseite (Tel. G/P \_\_\_\_\_ )

Bitte Telefon-Nummer angeben, damit Rückfragen möglich.

SCC-Mitglieder können sich hier melden zwecks Informationsaustausch. Bitte nebenstehende Karte verwenden. Sie erhalten dann auch die Adressen Ihrer Kollegen.

## Mitglieder helfen einander

Bin Privatmitglied  Bin Firmenmitglied  möchte Regionalgruppe beitreten  
 Besitze seit \_\_\_\_\_ einen Computer.

Marke: \_\_\_\_\_ Typ: \_\_\_\_\_ Speicher: \_\_\_\_\_ K  
 gekauft bei \_\_\_\_\_

Programmiere vor allem  BASIC  Assembler  Andere: \_\_\_\_\_  
 und löse \_\_\_\_\_

Habe  möchte Peripherie  Drucker \_\_\_\_\_  Floppy \_\_\_\_\_ / \_\_\_\_\_ K

Anderes: \_\_\_\_\_

Mein Beruf: \_\_\_\_\_ Geburtsdatum: \_\_\_\_\_

Bemerkungen: \_\_\_\_\_

Genauere Adresse auf der Rückseite (Tel. G/P \_\_\_\_\_ )

Bitte Telefon-Nummer angeben, damit Rückfragen möglich.

Weitere  
Karten  
vorne

bitte  
frankieren

Name \_\_\_\_\_  
Vorname \_\_\_\_\_  
Firma oder Beruf \_\_\_\_\_  
Strasse \_\_\_\_\_  
PLZ/Ort \_\_\_\_\_  
Geburtsdatum \_\_\_\_\_

SCC  
Seeburgstrasse 18  
6002 Luzern

bitte  
frankieren

Herr \_\_\_\_\_  
Frau \_\_\_\_\_  
Vorname \_\_\_\_\_  
Name \_\_\_\_\_  
Firma oder Beruf \_\_\_\_\_  
Strasse \_\_\_\_\_  
PLZ/Ort \_\_\_\_\_

Dialog Computer Treuhand AG  
«Small Business»  
Postfach 841  
6002 Luzern

bitte  
frankieren

Herr \_\_\_\_\_  
Frau \_\_\_\_\_  
Vorname \_\_\_\_\_  
Name \_\_\_\_\_  
Firma oder Beruf \_\_\_\_\_  
Strasse \_\_\_\_\_  
PLZ/Ort \_\_\_\_\_

SCC  
Seeburgstrasse 18  
6002 Luzern

**Auflage 10 000 Exemplare**

Mit einem Inserat erreichen Sie mehr als 10 000 interessierte und engagierte Personen – direkt zu Hause!

☎ 041 - 31 45 45

# Entdecken Sie die professionelle Leistung des HP-Kompakt-Computers.

Der HP-85 ist der Basiscomputer eines umfassenden Systems. Jetzt können Sie die Leistungsfähigkeit des HP-85 durch einfaches Anschliessen von Drucker, Plotter oder Floppy-Disk erheblich erweitern.



## Professionelle Leistung, wo sie gebraucht wird.

Schon der HP-85 allein bringt diese Leistung. Auf Ihrem Schreibtisch. Im Ingenieurbüro. Im Kleinbetrieb. Im Labor. Zuhause. Alles, was Sie dazu benötigen, ist in einer Einheit von weniger als 10 kg enthalten: Ein Bildschirm mit hoher Auflösung und Editier-Möglichkeiten. Ein leiser Thermodrucker zur Erstellung von Kopien des gesamten Bildschirm-inhaltes. Ein 16 KByte Arbeitsspeicher, der auf 32 KByte erweiterbar ist sowie ein 8 KByte Bildschirmspeicher. Ein Kassettenlaufwerk für Kassetten bis zu 217 KByte Speicherkapazität. Und eine vollständige Tastatur, einschliesslich acht frei belegbarer Funktionstasten. Dieses Leistungspaket ist einfach zu bedienen – dank der erweiterten Programmiersprache HP-Basic.

## Entscheiden Sie persönlich über Ihre Peripherie.

Ab sofort können Sie den HP-85 Ihren Anforderungen entsprechend ausbauen. Mit dem HP-IB Interface (IEEE-Standard 488) lassen sich gleichzeitig bis 14 verschiedene Peripheriegeräte anschliessen. So bauen Sie Ihr massgeschneidertes Computer-System auf, ohne dafür ein neues Programm schreiben zu müssen. Das übernehmen die ROMs, die als Zubehör erhältlich sind. Mit diesen neuen HP-ROMs können Sie das Betriebssystem mit leistungsstarken Befehlen und Funktionen auf 80 KByte erweitern. Lassen Sie sich den HP-85 bei einem der speziell geschulten HP-Fachhändler vorführen. Er wird Sie kompetent beraten und bedienen. Oder verlangen Sie bei Ihrem Fachhändler detaillierte Unterlagen.



**HEWLETT  
PACKARD**

**Aarau:** Otto Mathys AG, Kasinostrasse 32, Tel. 064/221493; **Agno:** Informatica Kauffmann, Palazzo San Provino, Tel. 091/594019; **Basel:** J.F. Pfeiffer AG, St. Jakobstrasse 59, Tel. 061/506300; **Bern:** Bärtschi & Co., Zeitglockenlaube 4, Tel. 031/225081; **J.F. Pfeiffer AG,** Effingerstrasse 16, Tel. 031/256262; **Chur:** J.F. Pfeiffer AG, Alexanderstrasse 16, Tel. 081/223026; **Einsiedeln:** Kälin Computer Systeme, Eisenbahnstrasse 13, Tel. 055/533500; **Genf:** Glanzware SA, 142-144, rue de Genève, Tel. 022/492977; **IRCO Electronic Center,** 3, rue Jean Violette, Tel. 022/203306; **Lausanne:** Glanzware SA, 70, avenue de Tivoli, Tel. 021/258434; **Schaer,** Grand-Pont 2 bis, Tel. 021/235555; **Luzern:** Dialog Computer Treuhand AG, Seeburgstrasse 18, Tel. 041/314545; **Lötscher AG,** Pilatusstrasse 18, Tel. 041/236366; **Neuenburg:** Reymond, Faubourg du Lac II, Tel. 038/252505; **Rüschlikon:** Instrumatic AG, Weingartenstrasse 9, Tel. 01/7241410; **St. Gallen:** Muggler AG, Neugasse 20, Tel. 071/223821; **Wetzikon:** Ing. Büro Heiniger, Turnhallenstrasse 2, Tel. 01/9302777; **Yverdon:** Schaer, Place Pestalozzi 12, Tel. 024/212378; **Zürich:** Amera Electronics AG, Lerchenhalde 73, Tel. 01/571112; **A. Baggenstos & Co. AG,** Waisenhausstrasse 2, Tel. 01/2213694; **J.F. Pfeiffer AG,** Löwenstrasse 61, Seestrasse 346, Tel. 01/459333.

Senden Sie mir detaillierte Unterlagen über das System HP-85. F 11

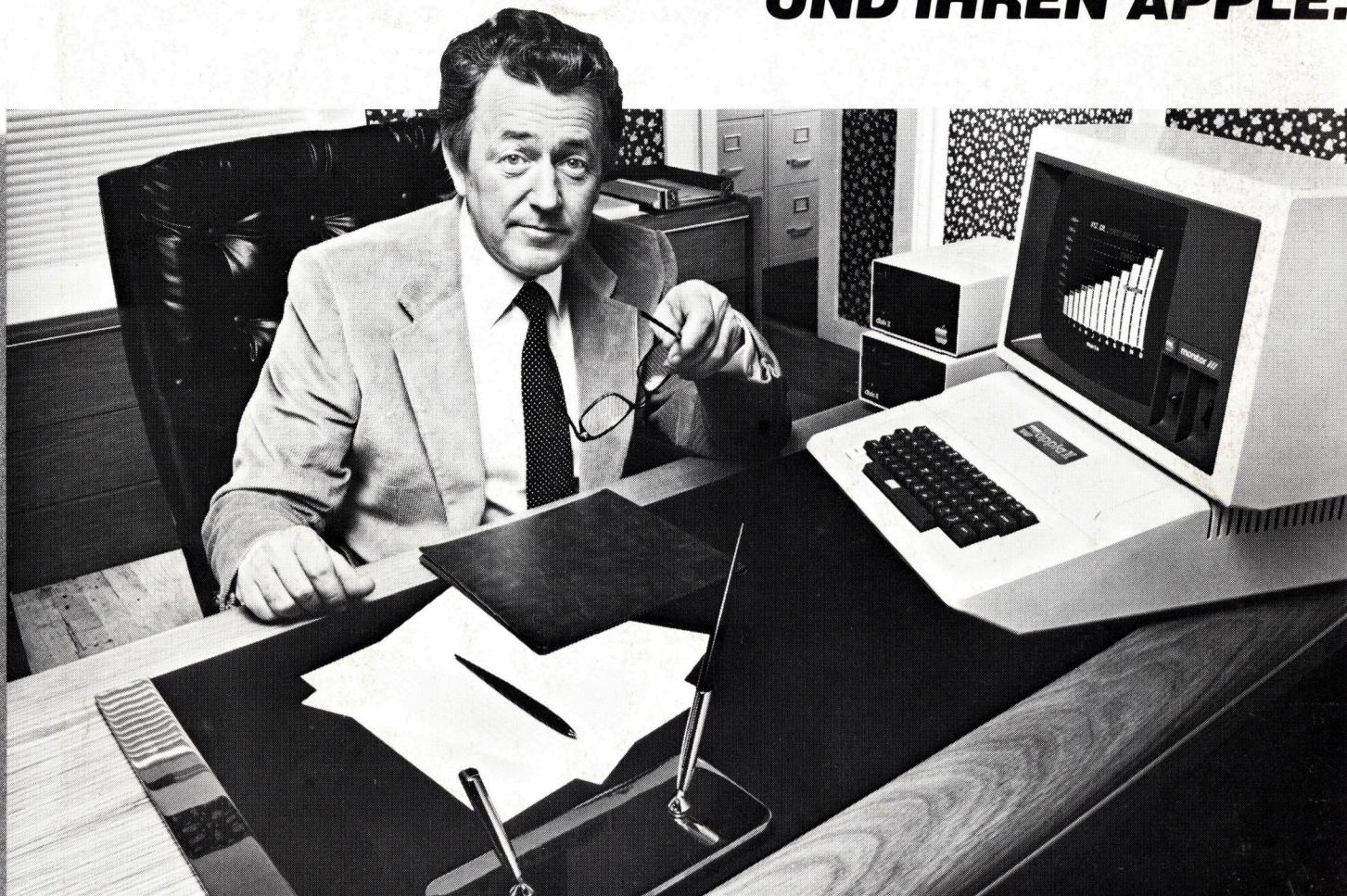
Name

Firma

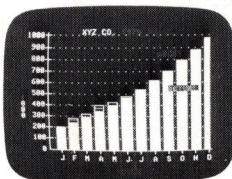
Adresse

Funktion

# FACHLEUTE GEBRAUCHEN IHREN VERSTAND UND IHREN APPLE.



Fachleute wie Sie erkennen den Wert von Hilfsmitteln, mit denen Probleme auf ein Minimum reduziert und die Leistungsfähigkeit maximiert



werden können. Deshalb kaufen immer mehr Fachleute Apple-Computer. Mit Hilfe eines Apple-Privatcomputers können

Sie tägliche Einzelheiten unter Kontrolle halten. Und Sie können besser für den nächsten Tag planen.

**Ein Apple hilft bei allem.** Früher konnten sich nur einige Unternehmen den Kauf eines Computers leisten. Heute bietet Ihnen ein Apple-Privatcomputer-System auf Ihrem Schreibtisch die Leistung eines großen Computers ohne die mit einem großen Computer verbundenen Sorgen. Und es bietet vielseitige Verwendbarkeit.

Ihr Apple kann Ihre Kreditoren- und Debitorenkonten führen. Er kann Übersicht über Ihren Warenbestand

halten. Ihre Postversandlisten führen. Und er kann Ihre Briefe,

Mitteilungen, Berichte und Artikel redigieren — mit nur ein paar schnellen Tastenanschlägen. Ihr Apple kann gespeicherte Informationen, numerische Daten oder Textmaterial, als Tabelle oder Diagramm ausdrucken.

Und dank seiner Vielseitigkeit kann Ihr Apple bei der Verkaufsprognose, den Rentabilitätsanalysen und der Budgetplanung behilflich sein. Der Apple ist Ihr Partner bei der Planung, das Hilfsmittel, mit dem Sie jede Möglichkeit untersuchen können, *bevor* Sie eine feste, endgültige Entscheidung treffen.

**Das Hilfsmittel, das mit Ihnen wächst.** Besuchen Sie noch heute Ihren Apple-Vertragshändler, um alles über die Familie der Apple-Privatcomputer zu erfahren. Lassen Sie sich die Apple-Softwarepakete zeigen, die schon auf Ihre besonderen

fachlichen Erfordernisse zugeschnitten sind. Entdecken Sie, wie einfach und wirtschaftlich es ist, Ihr Apple-System um einen Drucker, weitere Plattenlaufwerke oder mehr Speicherplatz zu erweitern.

Entdecken Sie, daß Sie nicht das geringste über Computer zu wissen brauchen, um einen Apple benutzen zu können. Und lassen Sie sich von Ihrem Vertragshändler erklären, wie Apple dank seines Service, seiner Zuverlässigkeit und seiner technischen Überlegenheit zu einem der führenden Namen auf dem Gebiet der Privatcomputer geworden ist.

Mit Ihrem eigenen Apple haben Sie die Leistung von Computern endlich dort, wo sie hingehört: unmittelbar zu Ihrer Verfügung.

apple computer

Wenn Sie weitere Informationen und den Namen des Ihnen am nächsten gelegenen Apple-Fachhändlers wünschen, wenden Sie sich bitte an die Generalvertretung der Schweiz

Industrade AG, Gemsenstrasse 2, 8021 Zürich