



Schneider Magazin

DM 6,- ÖS 50 SF 6,-

11

November '86
2. Jahrgang

Thema

- ★ Bericht: Schachcomputer-WM
- ★ Programm: Schacharchiv

Topprogramme

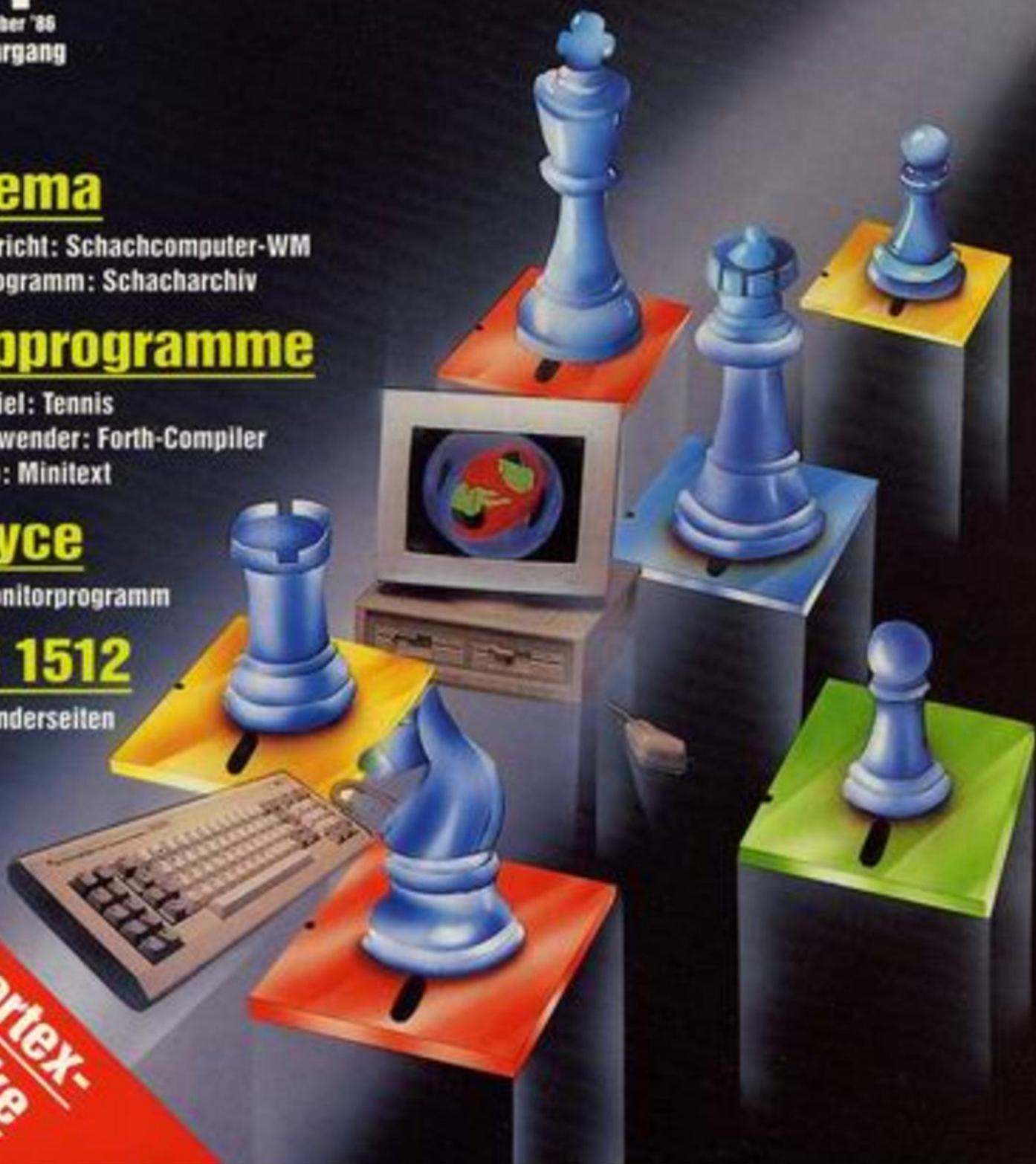
- ★ Spiel: Tennis
- ★ Anwender: Forth-Compiler
- ★ Tip: Minitext

Joyce

- ★ Monitorprogramm

PC 1512

- ★ Sonderseiten



vortex-
Ecke

**Wurden Sie,
was Sie alles
können?**

Schneider CPC
Arbeiten
mit Turbo Pascal



Karl-Hermann Rollke

Hans Lorenz Schneider
Das Schneider CPC Grafikbuch

Die vielfältigen Grafikmöglichkeiten Ihres Schneider CPC (464, 664 und 6128). Von einer allgemeinen Einführung über ergänzende Grafikbefehle, Erstellung von Grafiken mit dem Joystick, Darstellung zwei- und dreidimensionaler Diagramm-Formen, künstlerische Grafiken, Zusammenspiel zwischen Grafik und Datenträger bis hin zur Hardcopy.

328 Seiten / zahlr. Abb. ISBN 3-88745-611-4 (1986)
Best.-Nr. 3611 DM 48,-/sFr. 44,20/S 374,-

Schneider CPC
Erfolg mit Multiplan



Thorsten Ritter

Arthur Naiman
Schneider CPC - Einführung in WordStar

Der Bestseller zum populärsten Textverarbeitungsprogramm wurde für die Besitzer des CPC überarbeitet. Und damit wichtige Hilfe und Nachschlagewerk bei der Arbeit mit WordStar und Mail-Merge auf dem CPC. Neben der klaren Einführung in den effektiven Umgang mit WordStar gibt es u. a. auch wertvolle Hinweise für die Installation von Druckern und zu Systempatches.

280 Seiten / ca. 40 Abb. ISBN 3-88745-646-7 (1986)
Best.-Nr. 3646 DM 42,-/sFr. 38,60/S 328,-

**Arbeiten mit dem
Schneider CPC**



Hans Lorenz Schneider

Günter Woigk
Das Schneider CPC Systembuch

Erfahren Sie jetzt alles Wissenswerte über das Innenleben Ihres Schneider CPC 464, 664 oder 6128. Der Autor verrät Ihnen alle Interna Ihres Rechners und deren Nutzen für Ihre Anwendung. Aus dem Inhalt: Programmierstil und Programmelemente; die Feinheiten des Locomotive BASIC; Überblick über den Z80-Assembler; Maschinensprache auf dem CPC; die Firmware in- und auswendig; Unterschiede und Gemeinsamkeiten der CPC's. Das alles wird abgerundet durch viele Beispiele und Darstellungen sowie einen Anhang mit Grafiken, Tabellen und Verzeichnissen. Nutzen Sie das mit Informationen randvolle Nachschlagewerk, um das System Ihres Schneider CPC von Grund auf zu kennen und sicher in die Programmier-Arbeit einzusteigen.

Erscheint November '86

ca. 680 Seiten / mit Abb.
Best.-Nr. 3606

ISBN 3-88745-606-8 (1986)
ca. DM 58,-/sFr. 53,40/S 452,-

Karl-Hermann Rollke
Schneider CPC - Arbeiten mit Turbo Pascal

Der Bestseller „Das Turbo Pascal Buch“ wurde speziell für Besitzer eines Schneider CPC überarbeitet. So ermöglicht es den optimalen Einsatz der leistungsfähigen Programmiersprache – auf die CPC-Arbeits- und Systemumgebung zugeschnitten und durch zusätzliche Informationen ergänzt. Zahlreiche Beispiele veranschaulichen die vorgestellten Programmierkonzepte. Übungen zu jedem Kapitel machen dieses leicht verständlich geschriebene Buch auch für Einsteiger zur unentbehrlichen Arbeitshilfe.

296 Seiten / mit Abb. ISBN 3-88745-649-1 (1986)
Best.-Nr. 3649 DM 48,-/sFr. 44,20/S 374,-



Thorsten Ritter
Schneider CPC - Erfolg mit Multiplan

Ein didaktisch hervorragendes Lehrbuch und Nachschlagewerk für CPC-Besitzer, die das Tabellenkalkulationsprogramm Multiplan anwenden wollen. Das Buch ist die überarbeitete, auf den CPC zugeschnittene Version des Bestsellers „Erfolg mit Multiplan“ und führt Schritt für Schritt in das Programm ein. An konkreten Beispielen wird erklärt, welchen Nutzen CPC-Besitzer mit Multiplan in der Praxis haben.

200 Seiten / ca. 45 Abb. ISBN 3-88745-639-4 (1986)
Best.-Nr. 3639 ca. DM 48,-/sFr. 44,20/S 374,-

Schneider CPC
Einführung in
WordStar



Arthur Naiman

Hans Lorenz Schneider
Arbeiten mit dem Schneider CPC

Der Autor vermittelt alle Informationen zur BASIC-Programmierung Ihres Rechners, wie z. B.: allgemeine Routinen, die Sie immer wieder benötigen; Einsatzmöglichkeiten des Zeichensatzes; hochauflösende Grafik und ihre Anwendung; Routinen zur Darstellung von Sprites und Hardcopies; Dateiverwaltung und Sound-Editor; wichtige Betriebssystem-Routinen; Möglichkeiten der BASIC-Erweiterung. Eine umfassende und didaktisch aufbereitete Arbeitshilfe für Anfänger und fortgeschrittene CPC-Anwender.

228 Seiten / 113 Abb. ISBN 3-88745-603-3 (1985)
Best.-Nr. 3603 DM 38,-/sFr. 35,-/S 296,-

**Das
Schneider CPC
Systembuch**



Sybox
überall wo es gute Computerbücher
und Software gibt

SYBEX

– die guten Seiten Ihres Computer

Sybox Verlag GmbH
Vogelsanger Weg 11f
4000 Düsseldorf 30
Telefon: 02 11 / 61 80 20



Liebe Leser,
 im letzten Heft hatte ich Ihnen an dieser Stelle einige wichtige Neuerungen angekündigt. Bis auf einige kleine Probleme hat alles geklappt: Wir freuen uns, Ihnen ein Heft vorlegen zu können, das noch umfangreicher, informativer

und farbiger geworden ist. Da macht das Lesen dann noch einmal soviel Spaß.

Da sind als erstes die Seiten über den Schneider-PC. Sie werden in Zukunft einen festen Platz im Schneider-CPC-Magazin einnehmen. Auch hier haben wir an die Einsteiger gedacht, aber natürlich auch an die Spezialisten, für die wir einen 8086-Assemblerkurs beginnen.

Unser Schwerpunktthema in diesem Heft betrifft Schach und Computer. Daß diese beiden Dinge viel miteinander zu tun haben, ist bei Insidern inzwischen ein alter Hut. Wie es aber zu dieser Verbindung kam, darüber berichten wir anlässlich der diesjährigen Weltmeisterschaft für Computerschach in Köln.

Was die Farbseiten anbelangt, werden besonders die Spielefans ihren Augen kaum trauen: Wir haben viele neue Spiele nicht nur getestet, sondern sie auch so abgebildet, wie sie sind: nämlich farbig.

Jetzt noch ein wichtiger Hinweis für alle, die unseren Programmservice nutzen wollen. Neben den gesammelten Programmen aus den einzelnen Heften, die es jetzt übrigens auch auf Diskette gibt, bieten wir zusätzlich die neuen Programmsammlungen Player's Dream, Codex und Pascal Disc an. Da ist für jeden etwas dabei.

Bis zum nächsten Mal
 Ihr

Thomas Eberle, Chefredakteur

P.S. Vergessen Sie nicht: Ab dem neuen Jahr heißt das Schneider-CPC-Magazin nur noch Schneider-Magazin. Das gibt dann auch eine kleine grafische Titelländerung.

Alle neuen Leser haben die Möglichkeit, die zurückliegenden Hefte nachzubestellen. Bestellt wird mit untenstehendem Bestellschein. Die Lieferung erfolgt aber nur gegen Vorauskasse in Form von Briefmarken oder gegen Scheck.



CPC Schneider Magazin

- Exemplare 12/85 (5.50 DM)
- Exemplare 1/86 (5.50 DM)
- Exemplare 2/86 (5.50 DM)
- Exemplare 3/86 (5.50 DM)
- Exemplare 4/86 (5.50 DM)
- Exemplare 5/86 (5.50 DM)
- Exemplare 6/86 (5.50 DM)
- Exemplare 7/86 (5.50 DM)
- Exemplare 8-9/86 (5.50 DM)
- Exemplare 10/86 (6.00 DM)

Plus Versandkosten
 (1 Heft 1,40 DM, 2 Hefte 2,- DM)
 3 - 9 Hefte 3,- DM, 10 - 15 Hefte
 5,- DM)

Summe

Name: _____
 Straße: _____
 PLZ/Ort: _____
 Datum/Unterschrift: _____

**Bestellschein ausschneiden, ausfüllen, Scheck oder Briefmarken beilegen und abschicken an:
 Verlag Rätz - Eberle
 Postfach 1640, 7518 Bretten**



Ein kleines Modul, das einiges zu bieten hat. Acht verschiedene Trommeln stehen in digitalisierter Form zur Verfügung. Die Programmierung ist denkbar einfach.



Das Schachspiel selbst erfordert bekanntlich hohes Konzentrationsvermögen und ein gewisses Maß an Intelligenz. Ein Computer ist immer konzentriert. Wie er aber das Problem mit der Intelligenz meistert, soll in unserer neuen Serie etwas beleuchtet werden.



Das Top-Spiel des Monats ist eigentlich ein alter Hut. Handelt es sich dabei doch um den Urvater der Computerspiele, nämlich Tennis. Allerdings würde sich dieser Vorfahre selbst nicht mehr erkennen, wenn er sich in der neuen Form sehen könnte.

RUBRIKEN

Vorwort	3
Inhaltsverzeichnis	4
News	6
Preisausschreiben	12
Bücher	14
Buchversand	57
Softwareservice	82
Tip des Monats	88
Topprogramm	106
Topspiel	114
Händlerkarussell	119
Leserfragen	120
Kleinanzeigen	122
Leserecke	127
Software-Hitparade	130
Vorschau, Impressum, Inserentenverzeichnis	137

SOFTWARE-REVIEWS: SPIELE

Johnny Reb II	131
Arnhem	131
Rescue on Fractalus	132
Glass	132
Mission Omega	133
Split Personalities	134
Bounder	134
Stainless Steel	135
Doppelgänger	136
Spytrek	136
Eye Spy	136

SOFTWARE-REVIEWS: ANWENDERPROGRAMME

Rembrandt	18
StarDatei	20
Laser Genius	21
Disc Demon	22
Tastset	24

BERICHTE

Comal	26
Amdrum	31
Computerschach Teil 1	32

JOYCE

McLin-Nucleus	38
512 KByte RAM	39
Colossus Chess 4.0	40
Disc Mate	41

SERIEN

CP/M – die neue Perspektive (Teil 11)	42
Pascal-Kurs (Teil 10)	46
Basic-Stufen (Teil 6)	54
Z 80-Assemblertips (Teil 11)	58
Z 80-Assemblerkurs (Teil 11)	62
Datenverwaltung unter Turbo-Pascal (Teil 3)	66

PC 1512

Editorial	72
Hintergrund MS-DOS	72
Buch: Textverarbeitung auf dem Prüfstand	73
Buch: Das kann der Schneider PC	73
Assemblerkurs 8086 (Teil 1)	74

TIPS + TRICKS

Sieben auf einen Streich	80
Schacharchiv	84
Mini-Texter	88
Window Creator	94
Puzzlebild (PUZZ-MAD.PIC)	96
Ypsilon-Schalter	98
Ypsilon-Adapter	100

VORTEX-ECKE

Funktionstasten für den Monitor	102
File-Examiner	103
Speicherplatz errechnen	104
PARA.COM mit einem Laufwerk	105

ANWENDUNG DES MONATS

Forth-Compiler	106
----------------	-----

SPIEL DES MONATS

Tennis	114
--------	-----



Einen Riesenrummel hat der PC 1512 bis jetzt verursacht. Auch im Schneider-Magazin hat er diesmal schon 7 Seiten erobert. Hinweise auf Bücher für den PC und anderes wird der Übersichtlichkeit halber auf den PC-Seiten untergebracht. Ein 8086-Assembler beginnt in dieser Ausgabe, auch ein Textprogramm wird vorgestellt.



Der jugendliche Superheld Ricky Steel, sein Wunderauto Nightwind und die abgründige Gemeinheit von Dr. Vardos sorgen in einem durch Grafik und Animation bestehenden Spiel für pausenlosen Nervenkitzel



Tolle Grafik und hohe Geschwindigkeit bietet das Action-Game "Glass" aus dem Hause Quicksilva, einem der führenden englischen Softwarehäuser

SIREN die Schneider-Utilities

SOFTWARE DER SPITZENKLASSE

Lesen Sie den Testbericht im CPC-Magazin 8/9.86.

NEU* **DISCOVERY PLUS** IHRE KASSETTENSOFTWARE AUF DISKETTE
Das einzigartige "Tape to Disc" Transferprogramm
- Jetzt mit 4 Programmen - Transfer von normalen und headerlosen Programmen auf Tastendruck - Kopiert die verschiedensten Speedlock- und Turboladerprogramme - Packt den Großteil Ihrer Kassettensoftware auf Diskette, auch die, bei der bisher nichts zu machen war - Alle Teile des Programmpakets sind kinderleicht zu handhaben, auch für den Anfänger geeignet - Keine Programmierkenntnisse erforderlich - Komplett mit leicht verständlicher Anleitung und 100 Beispielen.
Die Zeitschrift AMSTRAD-ACTION sagte über das alte DISCOVERY: „Probably the best of the tape to disc copiers.“ Eines ist sicher: **Das neue DISCOVERY PLUS ist noch besser geworden.** Kompatibel zu CPC 464/664/6128 **3"-Diskette nur DM 59,90**

Discus — DISCOVERY User Service — Mit DISCUS erhalten DISCOVERY-User eine Sammlung von insgesamt mehr als 160 Transferproblemlösungen. DISCUS 1,2 u. 3 je **DM 5,—** Bei separater Bestellung + DM 1,— für Rückporto.

NEU* **HANDY MAN - 416 k form. Speicherkapazität**
auf dem CPC. Vergrößern Sie Ihren wertvollen Speicherplatz um 60 k pro Diskette. Das Utilitypaket **HANDY MAN** schafft es.
- Superschnelles Formatierprogramm für IBM-DATA-SYSTEM-VENDOR und SUPERMAT (208 K pro Diskettenseite). Das neue Format ist voll softwareunterstützt (Programmtransfer von normalen Formaten auf SUPERMAT etc. etc.). Floppy-Speedup (20%) unter Supermat
- Auch für zwei Floppystationen: Wahlfrei Erst- oder Zweitlaufwerk unter SUPERMAT oder beide, auch im Wechsel - MENU MAKER installiert ein Programm, mit dem sie alle Programme (auch alle USER-Ebenen) menügesteuert starten und alle wichtigen Programmparameter verwalten können. Alle Unterprogramme können versteckt werden - DISC SEARCH und FILE SEARCH: ganze Disketten nach Strings absuchen (Ideal z.B. für Adventures/Textsuche), Sektoren/Files aller Formate auslesen, verändern und wieder speichern (auch kopieren) - Komfortabler und sicherer Backup von Diskette auf Kasette - + + + weitere Utilities!!! Auch ideal für größere Dateien etc. Läuft unter AMSDOS.
Ein Weihnachts-Programm!!! CPC 464/664/6128 3"-Diskette nur DM 59,90

NEU* **MASTERDISC** — Disketten-Utilities Transfer Backup etc.
- Disketten kopieren, auch illegale Sektoren, nicht formatierte und zusätzliche Spuren.
- Alle Directory-Einträge editieren u. verändern, Files verstecken u. sichtbar machen, - Sektoren editieren u. verändern - Files kopieren, auch auf Kasette. - Listschutz aufheben - doppelt schnell formatieren - Arbeitet mit 1 und 2 Laufwerken - und, und, und... Voll menügesteuert
- **SPITZE** - Für CPC.464/664/6128 **3"-Diskette nur DM 59,90**

TAPE UTILITY U 3.4 — HIGH SPEED TAPE COPIER
Backup von Kasette auf Kasette, normale, headerlose, Turbo- und Speedlock-Programme. Header-Reader. **NUR CPC 464 Kasette nur DM 39,90**

DISC MATE — CP/M plus UTILITIES für den **Joyce, Joyce plus**
Gelöschte Files jeglicher Art regenerieren. Files vor Löschen schützen. User wechseln. Files verstecken/sichtbar machen. Sehr komfortables und schnelles Kopierprogramm, ideal für Backups, Diskettenzugriffszeiten um 20% beschleunigen.
Für CPC 6128, JOYCE und JOYCE plus **3"-Diskette nur DM 79,90**

3"-Zweitlaufwerk AMDRIVE zum Hammerpreis! **333,-**
Für alle CPC's incl. Netzteil. Qualitätslaufwerk in massivem Aluminiumgehäuse (3 mm stark) farblich passend zum CPC. Wie's funktioniert und bedient wird, können Sie in Ihrem Schneider-Handbuch nachlesen (d.h. voll kompatibel). Für CPC 664 und 6128 Besitzer: AMDRIVE paßt in der Größe genau auf das eingebaute Laufwerk. **Der Preis versteht sich inclusive Anschlußkabel!!!** Bei Bestellung CPC-Typ angeben. Preis + DM 8,— Versandkostenanteil.

MIRAGE IMAGER — Die Hardwarelösung für Kopien von Cass. auf Disc
- Kopiert Alle (III) Programme auf Diskette. - Kopiert auf Knopfdruck (Speicher wird „eingefroren“ und abgesaved d.h. an jeder Stelle des Programms kann gespeichert werden.)
MIRAGE IMAGER DM 219,— Adapter f. CPC 6128 DM 26,—

HARDWARE — zu Superpreisen!!!
SCHNEIDER, VORTEX etc. Preise auf Anfrage.

Weitere Infos gegen DM 1,50 in Briefmarken (Absender nicht vergessen).
Softwarebestellung unter DM 120,— zuzüglich DM 3,— f. Porto und Verpackung. Lieferung erfolgt gegen Vorauskasse (Euroscheck) oder per Nachnahme (zuzügl. DM 3,20 NN-Gebühren). Ausland gegen Vorausscheck.

Telefonische Eilbestellung und Information 09306/8735

PR8-Softwaredienst **Händleranfragen erwünscht**
Postfach, D-8702 Elsingon

Bestell-Coupon auf Postkarte kleben oder in Kuvert stecken und ab geht die Post!

INFO Bestellung Verrechn.-Scheck anbei per Nachnahme

_____ Stück _____ DM
_____ Stück _____ DM
_____ Stück _____ DM

Versandanteil (siehe oben) Nachnahme + 3,20 DM _____ DM
Absender: (bitte deutlich schreiben) Summe _____ DM

Name: _____
Straße: _____
PLZ/Ort: _____
Tel.-Nr.: _____ Unterschrift: _____

Direkt aus Hannover: 3.5"- und 5.25"- Laufwerke für den Joyce

Einen entscheidenden Schritt nach vorne wagte Innowave Data aus Hannover. Als erste Firma in Deutschland bietet sie 3.5"- und 5.25"-Laufwerke für den Joyce an, die 100%ig kompatibel zum FD-2-Laufwerk von Schneider sind. Der Anschluß ist auch für den ungeübten Anwender durch einfache Steckverbinder gewährleistet. Die Laufwerke verfügen über eine Kapazität von 1 Mbyte (unformatiert) bei 80 Tracks pro Seite mit einer Dichte von 135 tpi. Die Übertragungszeit beträgt 250 Kbits/sec bei einer Steprate von 3 ms Track zu Track. Die modernen Kompakt-Laufwerke verfügen über einen kollektorlosen Direktantrieb und leise Schrittmotoren und sind in transparente Gehäuse eingebaut.

Zwei weitere wichtige Punkte wären die niedrigeren Diskettenpreise des 3.5"- und vor allem des 5.25"-Formates sowie die praktisch nicht vorhandenen Lieferengpässe bei Disketten dieser beiden Formate. Bei einem Diskettenverbrauch von 50 Stück pro Jahr ergibt sich eine eindeutige Kostenersparnis. Die Laufwerke sind bei jedem guten Fachhändler oder direkt bei Innowave Data zu beziehen. Die Preise der Laufwerke liegen bei 689.- DM für 3.5" und bei 789.- DM für 5.25".

Innowave Data
Kurt-Schumacher-Straße 26/28
3000 Hannover 1
Tel. 05 11 / 1 30 39

Das Iron On - Coloring Kit

Es gibt wieder ein neues Produkt, daß man in die Rubrik "Für die, die schon alles haben" einordnen könnte. Worum es geht? Natürlich um das Iron On - Coloring Kit. Wer jetzt nicht weiß, was darunter zu verstehen ist, braucht sich nicht zu verstecken. Mir ging es ähnlich.

Das Kit besteht aus 20 Blatt Papier und fünf farbigen Filzschreibern sowie einer kurzen, englischen Anleitung. Dieser Anleitung kann man entnehmen, daß das neue Produkt dazu dienen

kann, farbige Computergrafiken auf Stoff zu übertragen, also eigene T-Shirt-Designs zu erstellen. Dazu nehme man ein Blatt des Papiers, entferne die Schutzschicht und lege das Papier dann in den vorhandenen Drucker (Thermo- oder Laserdrucker sind ungeeignet, aber Matrixdrucker können verwendet werden). Jetzt muß man nur noch die gewünschte Grafik (selbst gezeichnet oder aus einem Programm geklaut) in den CPC laden und eine möglichst gute Hardcopy-Routine zur Verfügung haben.

Aus Kostengründen sollte die ganze Aktion erst einmal mit einem normalen Blatt Papier ausprobiert werden. Hat man alles seinen Wünschen entsprechend vorbereitet, kann der richtige Ausdruck auf das Spezialpapier erfolgen. Dabei wird nach Art und Weise eines Durchschlag- oder Blaupapiers die Grafik auf ein zweites Blatt übertragen, d.h. durchgeschrieben. Natürlich entsteht dabei nur ein schwarz/weißer Ausdruck. Dieser kann aber unter Verwendung der beiliegenden Filzstifte rot, blau, grün, gelb oder orange eingefärbt werden. Ist man damit fertig wird die Grafik mit einem Bügeleisen auf ein Kleidungsstück übertragen. Der Hersteller empfiehlt Kunstfaser, Baumwolle ist scheinbar nicht so gut geeignet. Im Handumdrehen hat man die Grafik jetzt aufgebügelt.

Übrigens muß man nicht unbedingt einen Drucker verwenden, man kann auch Grafiken von Hand erstellen und aufbügeln. Wer sich schon immer geärgert hat, daß er seine Kunstwerke nicht einem breiteren Publikum vorstellen konnte, wird sicher sofort das Iron On - Kit bestellen. Ein Anwendungstip: Mit einer Videokamera und einem Digitizer den eigenen Kopf digitalisieren, ausdrucken und aufbügeln - schon hat man prima Werbe-T-Shirts für die nächste Klassensprecherwahl oder so. Arbeitnehmer könnten auf diese Weise im Büro oder Betrieb demonstrieren, wer hier die Nummer 1 ist.

Ich halte das Iron On - Coloring Kit für eine wirklich nette Spielerei, mit der man bei anderen Eindruck schinden oder auch nur zur eigenen Freude spielen kann. Wer Geld dafür übrig hat, warum nicht.

Bezugsquelle:
J. W. Wussow, Nürnberg
Rolf Knorre

PCW-Show-News

Seit einiger Zeit haben wir in Great Britain einen Presseemann, der für uns eifrig auf der Insel recherchiert. Einen ausführlichen Messebericht von Peter Cornforth finden Sie in dieser Ausgabe des Schneider-Magazins. Unter der Rubrik "News" haben wir einige Highlights für die CPCs herausgezogen, die in den nächsten Wochen auch auf dem deutschen Markt erhältlich sein werden.

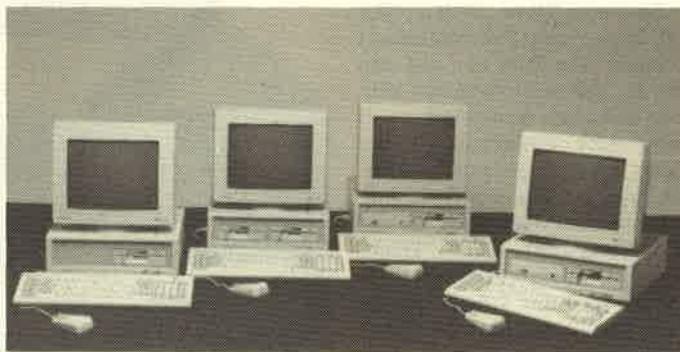
Am Stand der Firma Britannia war eine Eprom-Karte zu bestaunen, die sage und schreibe 15 Eproms aufnehmen kann. Möglich wird das Ganze durch ein Steuer-Eprom, das die einzelnen Steckplätze verwaltet. Wenn der CPC eingeschaltet wird, bekommt man ein Menü auf den Bildschirm, das alle belegten Steckplätze anzeigt, und zwar durch die Buchstaben A-O. Durch Drücken des entsprechenden Buchstaben kann man dann das gewählte Eprom ein- bzw.



Andrang beim Amstrad PC 1512

ausschalten. Interessant sind noch einige RSX-Befehle, die durch das Steuer-Eprom zur Verfügung gestellt werden. Als wichtigsten möchte ich den Befehl UPLOAD erwähnen. Mit ihm läßt sich ein beliebiges File in ein RAM-Modul laden. Ein weiterer Vorteil dieser Karte ist, daß bis zu 14 16-K-RAM-Module untergebracht werden können, die um die 18 Pfund kosten. Alle Möglichkeiten dieser Karte aufzuzählen, würde hier zu weit führen. Bemerkenswert ist, daß immer mehr Software auf Eproms angeboten wird. Ein gutes Beispiel dazu ist der auch in Deutschland erhältliche MAXAM, ein Assembler der Extraklasse.

Sehr interessant war auch das Multiface II von Romantic Robot. Es handelt sich dabei um eine NMI-Kopierkarte mit 8-K-Eprom und 8-K-RAM. Diese Karte ist auf



Knopfdruck in der Lage, jedes beliebige Programm nach dem Laden zu kopieren. Man kann es oder auch nur den Screen bei jeder beliebigen Spielposition absaven. Das Programm startet nach erneutem Laden genau da, wo es gesichert wurde. Außerdem sind noch einige Tools enthalten, mit denen im Speicher Manipulationen möglich sind. Über die Taste R ist jederzeit ein Rücksprung ins Programm möglich. Ein kleiner Trost für die Softwareanbieter: Die gesicherten Programme lassen sich nur mit Multiface II wieder laden.

Neue Software für den Schneider war eigentlich wenig auszumachen. Das vielgepriesene Spiel "Paperboy" von Elite war noch nicht vorführbereit. Einzige Attraktion auf dem Elite-Stand waren zwei Spielautomaten mit dem Spiel "Spaceharrier", die ununterbrochen von den Spielereaks umlagert wurden. "Spaceharrier" ist übrigens eines der nächsten Elite-Projekte. Man darf gespannt sein, ob es den Programmierern gelingt, ein Spiel von dieser Qualität und dieser atemberaubenden Grafik überhaupt auf den Homecomputer zu übertragen. Es wird da sicher einige Abstriche geben.

Interessanter sah es da schon auf dem Stand von Gargoyle aus. Das Spiel "Light-Force", das unter dem neuen Label FTL (faster than light) herauskommt, sah in einer noch nicht ganz fertigen Version schon sehr vielversprechend aus. Besonders die Geschwindigkeit der Animation ist hervorzuheben. Das saubere Scrolling und die schnelle Animation wurden durch eine neue Programmieretechnik mit dem phantasievollen Namen "Laser-Motion" erreicht. Man sollte dieses neue Label von Gargoyle im Auge behalten.

Sehr interessant war auch ein ausgiebiger Test des "OCP Art Studio" von Rainbird, das es leider wohl nur für den CPC 6128 geben wird. Die Möglichkeiten

dieses Malprogramms sind sehr beeindruckend. Ich habe bis jetzt auf dem Schneider CPC 6128 noch nichts Vergleichbares gesehen. Wir hoffen, in einer der nächsten Ausgaben darüber einen Test bringen zu können. Was mich sehr verblüffte, war die Tatsache, daß ein "Advanced OCP Art Studio" angekündigt wurde. Man darf gespannt sein, was an diesem Programm noch verbessert werden kann.

Ebenfalls von Rainbird gibt es bald das "Advanced Music System" für den CPC. Damit kann man nach Lust und Laune Musikstücke komponieren. Der Clou daran: Es ist möglich, die erstellten Musikstücke als MCode abzuspeichern, und man kann sie dann über RSX-Befehle zu eigenen Programmen im Hintergrund abspielen lassen. Das "Advanced Music System" ist übrigens MIDI-kompatibel.

Markteinführung des PC 1512 in Großbritannien

Am Dienstag, dem 2. September 1986 – zwei Tage früher als in Deutschland – wurden die neuen PCs von Amstrad in London der Presse vorgestellt. Annähernd 1000 Journalisten kamen, um zu sehen, was der Supercomputer nun wirklich kann.

Die Pressekonferenz fand in einem supermodernen Konferenzzentrum im Herzen von London statt. Manch einer der Presseleute erinnerte sich dabei wohl auch daran, daß die neuen Sinclair-Produkte früher nur 50 Meter weiter präsentiert wurden. Damals handelte es sich um einen klapprigen Tisch mit handgemalten Plakaten, heuer läßt sich Amstrad die Multi-Media-Show £ 30.000 kosten – die Zeiten ändern sich eben.

Es spricht für das Showtalent von Amstrad, eine derartige dra-

matistische Vorstellung für ein Produkt abzuliefern, das jedem vorher schon in allen Details bekannt war. Der Herausgeber der Zeitschrift MicroScope gratulierte Alan Sugar zu dieser sagenhaften Leistung, in so kurzer Zeit eine derartige Maschine gebaut zu haben und stellte sie im letzten Heft gleich in allen Einzelheiten vor.

Obwohl viele Insider davon ausgehen, daß der PC-Markt hoffnungslos überflutet ist, rechnet man bei Amstrad mit einer Verkaufsrate von monatlich 70.000 Stück, die sogar auf 100.000 gesteigert werden soll. Fred Köster, Chef der deutschen Schneider-Computer-Division, war da doch um einiges vorsichtiger: "...eine gut 5stellige Zahl..." (bei uns). Bis Ende 1986 will man weltweit 300.000 Stück an den Mann bringen. Nach einem weiteren Jahr sollen es nochmals 800.000 Stück sein. Mehr als die Hälfte davon in Übersee, wo man den US-Markt mit aller Gewalt attackieren möchte.

Alan Sugar: "Der PC-Markt macht jährlich einen weltweiten Umsatz von 34 Milliarden US-Dollar. Da wäre es doch gelacht, wenn wir uns nicht ein großes Stück vom Kuchen abschneiden könnten."

Eines ist auf jeden Fall sicher: Der neue PC 1512 wird eine Low-Cost-Lawine bei der Software



lostreten. Freuen kann sich darüber der Anwender, der jetzt, 5 Jahre nach dem ersten IBM-PC, eine echte Profi-Anlage mit Software für ca. 3000 DM bekommt.

Software-News

Neues hört man von Gargoyle Games. Unter der Bezeichnung FTL (Faster than light) erblickte ein neues Software-Label das Licht der Welt. Das erste Produkt dieses reinen Arcade-Labels nennt sich "Light Force". Die Vorab-Version macht schon einen sehr guten Eindruck. Sobald die endgültige Fassung vorliegt, werden wir das Programm genauestens unter die Lupe nehmen.

Der Name Infocom steht bekanntlich für herausragende Textadventures. Bisher konnten CPC-Benutzer nicht in den Genuss dieser Programme kommen. Das soll sich nun ändern. Die erste Software-Umsetzung auf Amstrad/Schneider bringt gleich einen Klassiker; das Science-Fiction-Ulk-Epos "The Hitchhikers Guide to the Galaxy" ist jetzt erhältlich. Das Monumentalwerk, das unter CP/M läuft, werden wir im nächsten CPC-Magazin ausführlich vorstellen. Wer sich schon mal einstimmen will, sollte sich den Roman "Per Anhalter durch die Galaxis" kaufen. Es lohnt sich!

Gute Nachrichten gibt es für Joyce-Besitzer, die auch mal spielen wollen. Immer mehr englische Firmen schreiben ihre aktuellen Spiele für die Joyce-Modelle um (z.B. Batman). Sobald uns die Testmuster vorliegen, werden wir darüber berichten.

Wer sich für die Geschehnisse um Rommel und Tobruk im Mai 1942 interessiert, kann sich auf das Programm "Tobruk" freuen. PSS-Software hat im Rahmen der Wargame-Serie diese Simulation am 28.8. auf den Markt gebracht. Ein ausführlicher Test folgt.

Ein neuer Trend zeichnet sich ab: Immer mehr Softwarehäuser bringen alte Titel neu auf den Markt. Schneider-Besitzer können davon nur profitieren, da einige gute Programme bisher noch nicht umgesetzt wurden. Im Zuge der Neuauflage erscheinen sie nun aber auch für die CPC-Computer, z.B. "Trashman", die Abenteuer eines Müllmanns bei der Arbeit. Leider liegt uns bisher nur die Spectrum-Version vor, aber wir bleiben am Ball.

Letzte Meldung: Werner kommt. Keine Angst, damit ist nicht etwa der Leiter der technischen Redaktion des CPC-Magazins gemeint. Nein, hier geht es um den Werner aus dem Norden der Republik, den mit der Flasche Bier. Ariolasoft hat in Zusammenarbeit mit Brösel (Werners Geistespapi) ein Computerspiel mit dieser beliebten Comicfigur auf den Markt gebracht. Lagerbericht folgt.

Rolf Knorre

Computertage Saar

Vom 5.-7. September fanden auch dieses Jahr wieder die Computertage Saar statt, bei denen fast 9000 Besucher den Weg in die Kongreßhalle fanden. Zwar gab es auch eine Hacker- und Spielecke, doch eindeutig standen professionelle Anwendungen im Vordergrund. Anbieter und Kunde waren unter sich; Leute, die nur sehen wollten, gab es weniger als bei den üblichen Verbrauchermessen. Wie in der Software-Branche allgemein üblich, war auch hier eine ganze Reihe von Unternehmen vertreten, deren Inhaber noch relativ



Prachtvoll

AMDRIVE heißt ein neues 3"-Zweitlaufwerk für alle Schneider-Computer. Es handelt sich dabei um ein Qualitätslaufwerk mit einem Jahr Vollgarantie, das inklusive Netzteil geliefert wird. Das Gerät befindet sich in einem stabilen Aluminiumgehäuse, das beim CPC 664/6128 in der

Größe genau zum eingebauten Laufwerk paßt. Wie's funktioniert und bedient wird, steht im Handbuch zum Computer. Preis des Laufwerks: 333 Märker.

PR8-Softwaredienst
K. Pracht
Postfach
8702 Eisingen

jung sind. Ebenso war auch das Publikum: 20-40 Jahre, im T-Shirt oder mit Diplomatenkoffer.

Schwerpunktmäßig war es eine 16-Bit-Demonstration. Alle Unternehmen waren vertreten, die sich in der Region mit EDV befassen. Zusätzlich rundeten Fachvorträge die gelungene Präsentation ab.

Neue Spiele im Herbst

Star Strike II

Ein schnelles Weltraum-Actionspiel mit verblüffend guter 3D-Vektorgrafik, die sogar mit ausgefüllten Flächen arbeitet. Gegen diese fantastische Grafik soll sogar der Renner "Elite" arm aussehen. Mal sehen.

Future Games

Etwas Neues für Sportspiel-Freunde: eine Olympiade auf dem Mond! Sie fristen ein trauriges Leben als Sklave, doch wenn Sie die sechs Disziplinen der Mond-Spiele innerhalb eines Zeitlimits schaffen, winkt die Freiheit! Wenn nicht, haben Sie halt Ihr Geld zum Fenster hinausgeworfen.

The Hive

Das jüngste Werk des Programmierer-Teams, das die erfolgreiche Schneider-Version von "Elite" geschrieben hat. Im neuen Arcade-Adventure mit vielen komplexen Puzzles und 3D-Grafik müssen Sie sich gegen eine böse Königin behaupten.

In wenigen Wochen werden weitere Neuerscheinungen folgen: das packende Fantasy-Action-Spiel "Druid" und die Titel "Pandora" und "Cholo".

Diepholzer Computer Versand

G. Frobieter, Fichtenweg 10, 2840 Diepholz ☎ 054 41 / 2983

Joyce 8256
erweitert auf 512 KByte DM **1749.-**

3"-Disketten (10er Pack)	DM 89.-
5 1/4"-2D-Disketten (10er Pack)	DM 12.-
Abdeckhauben CPC 464	DM 12.-
Abdeckhauben CPC 664	DM 12.-
Diskettenbox DX 85	DM 20.-
Diskettenbox SS 50	DM 20.-

Fordern Sie unsere Preisliste an - Händleranfragen erwünscht!

Jetzt auch in Berlin: DCV-Berlin, Waldsassener Str. 48, Telefon 030/7 11 85 92

Neu: Vertrieb Österreich · Telefon (02 22) 97 21 66 Wagner Electronics · Hauptstraße 171 3001 Mauerbach

GSX-Grafik-Treiber BASIC,
(für Striche, Kästchen, Polygonzüge = Kreise, Kuchen, Torten, Männchen). **Endlich kann die Grafik-Fähigkeit der JOYCE genutzt werden!** **64,-**
Standardleistungsbuch (StLB)
für das Bauwesen auf 3"-Disketten!!
für Gartenbau, Hochbau, Elektrotechnik, Gas, Wasser, Abwasser, Wasserwirtschaft Pr. a. Anfr.
Schach, "3-D Clock Chess"
für JOYCE **59,-**

Aufrüstung JOYCE auf JOYCE Plus
- **RAM-Erweiterung** von 256 KB auf **512 KB**, **Orig.-RAM-Baust.**, mit ausführl. Einbauanleitung. **119,-**
- 2. Laufwerk **FD-2**, 1 MB **579,-**

Papierführung JOYCE-Drucker
für exakten Papiereinzug und Wiederholgenauigkeit **37,-**

Bildschirmfilter für
- GT 64/65 bis 12/85 **39,-**
- CTM 644 **49,-**
- JOYCE **59,-**

Begrenztes **Sonderangebot: 3"-Disketten**
100 Pack. mit 10 St. **CF2 89,-**, 50 Pack. mit 5 St. **CF2DD 79,-**

Telefon (0 70 71) 6 60 22
G + W · Hartmeyerstraße 50 · 7400 Tübingen

PROFI PAINTER CPC

Der Profi Painter packt aus:

Leicht hat er es sich nicht gemacht, und viele dachten: „Was der vor hat, das geht gar nicht.“ Thomas Vervost, Autor des Profi Painter CPC, hatte sich ausgerechnet die stärksten Grafikprogramme als Konkurrenz und Maßstab ausgesucht. Sein ehrgeiziges Ziel: Er wollte ein Programm schreiben, das eigentlich unmöglich ist – den Profi Painter CPC. Einfach zu bedienen sollte es sein, schnell sollte es sein, flexibel sollte es sein. Kurz, seinen großen Vorbildern aus der 32Bit-Welt möglichst wenig nachstehen. Nur eben auf einem 8Bit-Rechner. Und das Ergebnis? „Die Geschwindigkeit und die vielen Möglichkeiten, die dem Anwender das Arbeiten erleichtern, machen den Umgang mit Profi Painter CPC zu einem wirklichen Vergnügen.“ Das schrieb CPC International im Test. Der „Aktuelle Softwaremarkt“ schloß mit dem Fazit: „Der Firma DATA BECKER sowie dem Autor des Programmes ist nur zu gratulieren.“ Den einen fasziniert die Einfachheit der Bedienung über Symbole wie Bleistift, Farbeimer, Lineal oder Sprühdose, den anderen die Leistungsfähigkeit, die das Programm bietet:

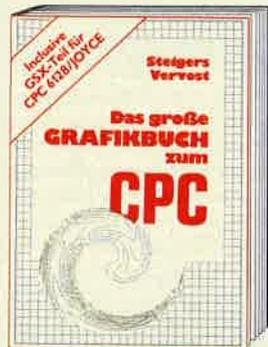
- Pinsel mit verschiedenen Strichstärken
- Sprühdosenfunktion für Farben und Muster
- Ausschneiden und Verschieben beliebiger Bildausschnitte
- Invertieren, Rotieren, horizontales und vertikales Drehen von Objekten
- optionale Rasterunterlegung
- problemloses Zeichnen von Geraden, Rechtecken, Kreisen, Ovalen usw.

PROFI PAINTER CPC

DM 99,-

Dazu kommt das ausführliche deutsche Handbuch, das leichtverständlich in den Umgang mit dem Programm einführt.

Nur eine Frage blieb bei den Profi-Painter-Besitzern dennoch offen: „Wie hat Thomas Vervost das gemacht?“ Die Antwort darauf gibt er selbst: mit dem großen Grafik-Buch zum CPC. Zusammen mit Jürgen Steigers, dem



Co-Autor von 6128 Intern, fand sich ein Top-Team in Sachen Schneider-Grafik: „Wir wollten das Schneider-Grafik-Buch schlechthin machen. Ein Buch, in dem Profis Neues lernen und das aus Einsteigern Profis macht.“ Aus diesem Anspruch ergibt sich schon fast die Gliederung. Der erste Teil widmet sich ausschließlich der Grafikprogrammierung in BASIC. Autoren und Leser entwickeln gemeinsam ein einfaches Malprogramm. Auf diese Weise erhält man nicht nur ein nützliches Utility, sondern es werden wichtige Grundlagen vermittelt. Dann kommt die Spezialitäten-Abteilung: Business- und Vektorgrafik sind hier genauso Themen wie Peripheriegeräte zum

Thema Grafik. Daß Jürgen Steigers zu Recht durch das 6128 Intern als CPC-Kenner bekannt wurde und sich auch durch seine „Bastelgags“ in der DATA WELT als Hardware spezialist etablierte, zeigt sich im nächsten Kapitel. Hier hat er eine gehörige Portion Arbeit geleistet: er erklärt nicht nur, wie ein Lightpen funktioniert, sondern anhand von Schaltplänen und exakten Bauanleitungen auch gleich, wie man so etwas selber bauen kann. Der große Maschinensprache-Teil schließlich ist ein Leckerbissen für Grafik-Freaks und solche, die es werden wollen. Neben allen Einsprungadressen der Grafikroutinen finden Sie auch eine komplette Abhandlung über GSX. Sie wissen nicht, was GSX ist? Keine Sorge, so geht es den meisten, denn bisher wurde diese grafische System-Erweiterung noch nie wirklich dokumentiert. Lassen Sie sich von Thomas Vervost und Jürgen Steigers zeigen, wie Sie unglaubliche Grafiken auf Ihrem CPC zaubern können. Übrigens: Alles, was Sie hier gelesen haben, ist natürlich nur ein kleiner Ausschnitt dessen, was Sie in diesem Buch finden werden. Kunststück, bei fast 500 Seiten ...

**Das große Grafikbuch zum CPC
Hardcover, ca. 500 Seiten, DM 49,-**

BESTELL-COUPON

CM 116

Einsenden an: DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf 1

Bitte senden Sie mir:

Das große Grafikbuch zum CPC DM 49,-

zzgl. 5 DM Versandkosten

PROFI PAINTER CPC DM 99,-

per Nachnahme

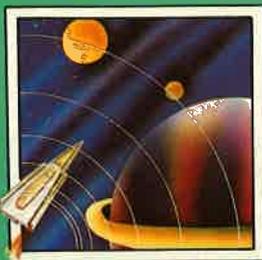
Verrechnungsscheck liegt bei

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010



The PLAYER'S DREAM



Für alle, die Ihren Computer nicht nur mit Anwendungen langweilen wollen, haben wir aus den Heften 12/85 bis 10/86 zehn tolle Spiele herausgesucht und auf Cassette verpackt. Ihre Augen können sich somit voll der Grafik widmen, anstatt gerötet auf Listings zu starren. Damit es aber noch angenehmer wird, bieten wir die Sammlung auch gleich auf Diskette an. In Sekundenschnelle katapultiert man sich mitten hinein ins Spielgeschehen. Auf beiden Datenträgern sind folgende Zeitvertreiber abgespeichert:

Darts (12/85), Senso (12/85), Showdown (1/86), Jump Over (2/86), Pingo (2/86), Zentus (5/86), Steinschlag (6/86), Centibug (7/86), Jolly Jumper (8-9/86) und Pyramide (10/86).



C O D E X



Damit Anwender keine kostbare Zeit mit Abtippen verschwenden müssen, gibt es jetzt 15 praktische Programme aus den Heften 12/85 bis 10/86 auf Diskette. Da Anwendungen auch schnell sein sollen, haben wir auf eine Cassettenversion verzichtet. Vom Dateiprogramm bis zum Speicherkibitzer ist so manches Problem erfaßt.

Hexmonitor (12/85), Sprites mit Editor (12/85), Kalender (12/85), Datenverwaltung (1/86), DIR-Doctor (Directory-Editor) (2/86), Mini-Monitor (3/86), Mathematik (3/86), Statistik (4/86), Baudcopy (4/86), Hidump (Hardcopy) (4/86), Bücherdatei (5/86), Labelassembler ASSO (6/86), Notizblock (Sideklick, 6/86), Basic-Compiler (8-9/86), Disassembler (10/86).



P PASCAL DISC

Damit Sie sich mehr mit der Theorie unseres Pascal-Kurses in den CPC-Magazinen 1/86 bis 11/86 als mit dem Abtippen der Beispiele beschäftigen können, bieten wir die Programme auf einer einzelnen Diskette an. Wenn Sie aber mit der unten aufgeführten Liste nicht viel anfangen können, so schauen Sie sich doch einfach noch einmal alle Kursteile daraufhin durch, ob Pascal nicht doch etwas für Ihren CPC wäre.

Adreßverwaltung (6/86), ASCII-Lister (7/86), WS-Konverter (7/86), File Verwaltung (BDOS-Funktionen) (8-9/86), Grafik-Funktionen (10/86), Firmware-Funktionen (10/86), Grafikgag (10/86), Liste (11/86) und noch ein paar mehr.

Software-Bestellschein

Ich bestelle aus dem CPC-Programmservice folgende Software:

Anzahl	Titel	Einzelpreis	Gesamtpreis
	Player's Dream Cass.	19,90 DM	
	Player's Dream 3" Disk	24,90 DM	
	Player's Dream 5,25" Disk	19,90 DM	
	Codex 3" Disk	24,90 DM	
	Codex 5,25" Disk	19,90 DM	
	Pascal 3" Disk	24,90 DM	

Ich wünsche folgende Bezahlung:

- Nachnahme (zuz. 5,70 DM Versandkosten)
 Vorauskasse (keine Versandkosten)

Bei Vorauskasse bitte Scheck beilegen oder auf Postcheckkonto Karlsruhe 434 23-756 überweisen.

Name des Bestellers

PLZ/Ort

Anschrift

Datum/Unterschrift

Coupon ausschneiden, auf Postkarte kleben und einsenden an:

Verlag Rätz-Eberle/CPC-Software, Postfach 16 40, 7518 Bretten.

Fingerschonend



Auf 3"-Disk
und Cassette

"CPC Fingerschonend" gibt es natürlich auch weiterhin. Neu ist nur das wunderbar gestylte Cover. Aber, und darauf haben sicher viele schon gewartet, jetzt gibt es die tollen Programme – von Lesern für Leser – endlich auch auf Diskette. Abtippfehler und lange Ladezeiten haben keine Chance mehr. Für Einzelbestellung oder Abonnement von "Fingerschonend" benutzen Sie bitte den Bestellschein beim Software-Service.

Die Spezialisten-Ecke

Aus dem Bereich Business-Software kommen immer mehr Programme, die sich an ganz spezielle Anwendergruppen richten. Die Mehrheit der CPC- und Joyce-Benutzer kann damit nichts anfangen. Aus diesem Grund werden wir darauf verzichten, ausführliche Testberichte zu bringen. Ganz übergehen wollen wir diese Programme aber auch nicht, da sich unter unserer Leserschaft sicher auch Spezialisten befinden, die gerade auf solch ein Programm gewartet haben. In der Spezialisten-Ecke werden also in Zukunft diese Programme kurz vorgestellt. Einzelheiten können dann direkt beim Vertrieb erfragt werden.

Wärmebedarfsberechnung nach DIN 4701/83

Joyce- und Joyce-Plus-Besitzer, die sich mit dieser Thematik beruflich beschäftigen, erhalten jetzt softwaremäßige Hilfestellung. Einige Beweise der Leistungsfähigkeit:

- Hausspezifische Daten müssen nur einmal eingegeben werden
- Automatische Vorgabe der Fugenzahl
- Unzulässige Eingaben werden nicht angenommen
- Höhenkorrekturfaktoren werden automatisch ermittelt
- Lüftungswärmebedarf Geschloß-/Schacht-Typ

WÄRME-BEDARFSBERECHNUNG DIN 4701/83

(c)WHS-Soft V1.1

1 - normale Lage Einzelhaus	1sp	I	5 - normale Lage Einzelhaus	1sp	I
2 - normale Lage Reihenhaus	1sp	II	6 - normale Lage Reihenhaus	1sp	II
3 - freie Lage Einzelhaus	1sp	I	7 - freie Lage Einzelhaus	1sp	I
4 - freie Lage Reihenhaus	1sp	II	8 - freie Lage Reihenhaus	1sp	II
Hauskenngröße	N	= 0.72			
Windgeschwindigkeit	v _w	= 0.80			
Konstante	C ₁	= 0.80			
Konstante	C ₂	= 0.10			
Konstante	C ₄	= -0.7334			

- Automatische Abstimmung des Zeta-Wertes
- Berechnung der Raumkennzahl
- Vorgabe des Fugendurchlässigkeitsfaktors
- Berücksichtigung von Rolläden etc.
- Bis zu 60 Räume können pro Diskette bearbeitet werden
- Übersichtlicher Raumausdruck im Format DIN A4

Das ist nur ein kleiner Auszug der Möglichkeiten, die dieses Programm bietet. Ich bin davon überzeugt, daß z.B. Architekten durch dieses Programm erheblich entlastet werden. Auch das Preis-/Leistungsverhältnis ist unschlagbar. Der Verkaufspreis beträgt mit k-Wert-Berechnung 298.- DM, ohne 248.- DM.

WHS Hinderer
Hohenzollernstraße 9
7447 Aichtal-Grötzingen

Konkursverwaltung

Rechtsanwälte und Konkursverwalter, die in ihrer Kanzlei einen Joyce Plus stehen haben, sind die Zielgruppe für das Programm "Konkursverwaltung" aus dem Hause Carat Soft. Das Programm erlaubt die komplette Durchführung eines Konkurses mit Schuldner- und Gläubigerverzeichnis, körperlicher Inventur und Kassenbuch. Dabei ist zur Erweiterung der Speicherkapazität auch der Einsatz einer Festplatte möglich. Weitere Informationen erhalten Sie von:

Carat Soft
Nelkenstraße 27
4600 Dortmund
Friedbert Reisser

data berger

Das Diskettenablage-System der Zukunft

3" MEDIA-BOX für bis zu 100 Disketten	42.90
3.5" MEDIA-BOX für bis zu 150 Disketten	42.90
5.25" MEDIA-BOX für bis zu 70 Disketten	64.90
MEDIA CASSETTEN BOX für 16 Cassetten	20.90

Urteil der Fachpresse:

Die POSSO MEDIA BOX ist eine Diskettenbox, die ihrem Preis-, Leistungs- und Funktionsverhältnis endlich gerecht wird.

CPC Mouse Pack
mit Software **228.-**

JOYSTICKS

Quickshot II	19.90
Quickshot IV - Joy Ball -	39.-
Competition Pro 5000	49.-

SOFTWARE

KASSEMBL 12
Der außergewöhnlich schnelle und komfortable unter den Assemblern **98.-**

SECURANCE
Die Kundenstammmverwaltung für eigene und fremde Versicherungsbestände unter dBase II **149.-**

Würfelspaß
Das Spiel für die ganze Familie **49.-**

TreffMann
Das Wettbewerbsprogramm für Schießsportvereine **98.-**

ElkaMan
Damit Sie keinen Termin mehr vergessen oder versäumen **79.-**

CopyMan
Damit Ihre Hardcopies so aussehen, wie Sie es sich wünschen. An jeden Drucker anpaßbar **98.-**

miniAktien
Damit Sie wissen, wo es am Aktienmarkt lang geht **49.-**

Genaue Beschreibung unserer Artikel finden Sie in unserem neuen 30seitigen Katalog mit den tollen günstigen Preisen und Beilagen vom Heim-Verlag und Markt & Technik. Versand nur gegen 2.- DM Rückporto.

3" Disketten No Name (Bulk) mit Shutter **79.-**

data berger

Im Lichtenfelde 76, 4790 Paderborn, RUF 05251/64852

1 2 3 4 5 6 7 8 Eingabe)

Hauptmenü der Wärmebedarfsrechnung

SUPER ~ WE

Diese Firmen haben folgende Preise gestiftet:

GfA

- 2 Turbo-Adress
- 2 RH-Buero
- 1 Beitragsverwaltung
- 2 RH-DMON
- 2 RH-DAT
- 1 Mica

Profi-Soft

- 5 Tasword für 6128
- 1 Tasword für 464

Carat-Soft

- 1 Adressenverwaltung
- 1 Vereinsverwaltung
- 1 Kassenbuch
- 1 Dateiverwaltung

Micro-Land

- 5 Brief- und Adreßprogramme
- 2 Easy Memo
- 5 Geschäftspakete mit Lagerverwaltung Rechnungsschreibung Einnahme-Überschuß-Rechnung
- 5 Geschäftspakete mit Rechnungsschreibung Deutscher Zeichensatz Brief- und Adreßprogramm Karteiprogramm
- 10 Fluglehrer
- 10 Vokabeltrainer
- 10 Die Zeitmaschine
- 5 Biorhythmus

Van der Zalm

- 1 Textking
- 1 Fibuking

Otten/Fecht

- 1 Oktett-Meßdatensystem

Rethemeier

- 1 Lightpen
- 1 Headline-Programm

Ferdl's Computer Software

- 2 Anwenderprogramme
- 4 Spiele
- 1 5er Pack 3"-Disketten

Deltacom

- 3 Anwenderprogramme

vortex

- 1 Maskgen
- 1 Vector

Weeske

- 5 Adventures
- 1 Vokabel-/Verbentruainer

Ingo Fleckenstein

- 2 Sprit Control

Michael Naujoks

- 1 64K Memory Expansion
- 1 Speech Synthesizer

R. R. Software

- 2 Biorhythmus

Cico'tronics

- 2 Rechnung

Peter West Records

- 1 Laser Basic
- 1 Laser Compiler
- 1 Sack voll Spiele

Schäfer

- 1 Zyklus

Computer Team

- 1 Basic-Compiler

Roth u. Partner

- 1 Turbo-Adress

ZS-Soft

- 10 Spiele
- 1 Grafpad II

PIZie Data

- 1 Grafpad II

Media Plast

- 40 Abdeckhauben

Schechensoft

- 1 Lern-, Diagramm- und Rechnungs-Fix

EDV-Service Probst

- 1 Maxam-ROM-Modul
- 1 Reis-Maus

Micro-Market Worms

- 1 Electric Studio Lightpen
- 1 System X Basic-Erweiterung
- 1 Disk UNI-DAT 3.3 mit 18 Public Domain CP/M-Utilities
- 1 Window-Lager
- 1 Window-Kasse
- 1 Window-Adreß

Compi-Club

- 2 Spiele

SFK-Elektronik

- 1 Lightpen
- 4 Spiele

Verlag Rätz-Eberle

- 10 Schneider-Bücher

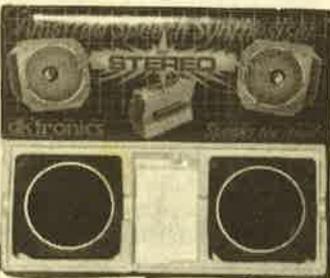
Syber-Verlag

- 3 Assemblerkurs für Schneider CPC
- 3 CPC Systembuch
- 3 Arbeiten mit dem CPC
- 3 Star Texter
- 3 Schneider Grafikbuch

Weit über 1000 Lösungen erhielten wir für unseren Wirtschafts-Spionage-Krimi. Unter allen richtigen Einsendungen – der Täter war ja Dr. Uwe Schindler – wurden inzwischen die glücklichen Gewinner durch Los ermittelt. Zum größten Teil haben sie ihre Gewinne bereits erhalten – ihnen allen einen herzlichen Glückwunsch! Die Gewinnerliste finden Sie nebenstehend. Wer diesmal nichts gewonnen hat, kann sich trösten: unser nächster Wettbewerb kommt bestimmt.

Bedanken möchten wir uns natürlich auch bei all den Firmen, die für diesen Superwettbewerb die vielen wertvollen Preise gestiftet haben. Ohne sie wäre der Erfolg der ganzen Aktion nicht möglich gewesen. Links haben wir deshalb noch einmal alle Firmen mit den entsprechenden Preisen aufgelistet.

ITBEWERB



Andreas Ettel
8748 Bad Bergzabern
Werner Gruner
7121 Ammrigheim
Dirk Moewes
8834 Wallerfangen
Klaus Multerer
8113 Kochal am See
Harry Liedtke
5142 Hückelhoven
Thomas Bucher
7882 Wollegg/Altann
Frits Zeiger
8730 Neustadt 22
Elmar Schuler
6846 Wahlen
Andreas Thiemann
4437 Schöppingen
Thomas Krauss
8900 Heidelberg
Jürgen Meier
8490 Cham-Chammünster
Josef Niedermeyer
8371 Langdorf
Thomas Fuhrmann
8825 Sonnsfeld
Lothar Mettel
8624 Großrosseln 6
Maria Schöneberg
5415 Bendorf
Martin Kohring
2000 Hamburg 76
Jürgen Haas
7000 Stuttgart 50
Sven Looschelders
2300 Kiel 1
Ralf Wegener
3410 Norheim
Mario Vranken
5100 Aachen
Thomas Wodtke
8072 Dreieich
Klaus-Jürgen Clausen
5000 Köln 1
Thomas Danner
8908 Gerethofen
Thomas Ritter
7015 Kornal 2
Torsten Lochmann
3460 Holzminden 1
Matthias Schmid
A-6460 Imst
Volker Glumm
4840 Rheda-WD
Peter Köhler
7760 Konstanz 16
Klaus Wombacher
8780 Aschaffenburg 19
Karl-Heinz Martens
2351 Neumünster
Ralf Gausmann
4500 Osnabrück
Jochen Kurdelbaum
4380 Recklinghausen
Kal Jürgen Gerstner
5450 Neuwied 23
Thomas Gauking
4420 Coesfeld
George Lorsche
L-3473 Dudelange
Dieter Kothe
8270 Idstein/Ts
Uli Vetter
7080 Aalen-Ebnat
Isolde Beck
3577 Neustadt
Lothar Ide
3688 Homberg-Wernsberg
Bernd Krüger
4500 Osnabrück
Marco Konrad
8580 Bayreuth
Dietmar Kleinhaus
7800 Freiburg
P. Podjed
8000 München 19
Jörg Bauer
7441 Unterensingen
M. Fritschle
7890 Ravensburg
F. Teufel
7122 Beilheim
B. Urban
7880 Ravensburg
Martin Bandelow
7121 Mundelshelm
Brigitte Migge
2800 Bremen 21

Gerd Steinmüller
8000 Frankfurt 70
Edi Schaller
CH-3110 Münsingen
Reinhard Hillmann
1000 Berlin 30
Carsten Westdörp
2901 Wiefelstede
Carmen Löbl
5500 Trier
Karl-Heinz Harves
2117 Wistedt
Dieter Wermuth
CH-8830 Rütli
Wolfgang Kaiser
2300 Kiel 1
Andreas Metzger
7809 Tomsdingen
Ulrich Bildmann
7142 Marbach/W.
Renate Sandelac
7121 Mundelshelm
Olaf Bartsch
8107 Ettal
Hans-Jürgen Krieg
8011 Höhenkirchen
Lars Eklert
3208 Giesen 1
Gerhard Glück
8050 Freising
Horst Friedrich
7630 Pforzheim 14
Oliver Hoffmann
7030 Böblingen
Klaus Biesel
6810 Lebach
Josef Füglein
8721 Michelau
Andreas Schütz
3601 Fuldaal 2
Helge Petersen
2390 Flensburg
Georg Geenen
2800 Oldenburg
Ralf Zebisch
7185 Fichtenberg
Ulrich Haunschild
8071 Denkendorf
Dr. W. R. Wirantprawira
8900 Heidelberg 1
K. Eberle
7631 Kieselbronn
Thomas Susalek
7080 Aalen-Ebnat
Michael Korber
A-8951 Ainat
Gudrun Stigenbauer
8781 Ruppertsweiler
Frank Maletz
3000 Hannover 81
David Brungs
5400 Koblenz
Michael Koderisch
8750 Kaiserslautern
Lorena Heckel
7803 Laichingen
Carsten Meier
2400 Lübeck
Thomas Rotter
8874 Oberstaufen
Peter Klar
7300 Esslingen
Andreas Porada
4055 Elmpt
Barbara Sagitski
8500 Nürnberg
Thomas Knorr
5110 Aisdorf
Thomas Schlagenhauer
8070 Ingolstadt
Eva Dosslerer
8218 Unterwössen
Patrick Bingen
L-6451 Echternach
Mario Mülter
4780 Scharding
Hubert Kamlage
5946 Schmallenberg 1
Stein Vetter
7080 Aalen-Ebnat
Lars Meier
2400 Lübeck
Lore Bauer
2351 Badenstedt
Bertram
8457 Maintal 2
Stoppel
4400 Münster

Harald Koppay
2000 Hamburg 76
Joachim Werner
8207 Ruppichterth 4
Christian Trepl
8370 Regen
Pari Garvano
5276 Bergneustadt
Thomas Tegethoff
7622 Philippsburg
Friedhelm Schang
4370 Marl
Lars Larsen
2808 Wardsberg
Per Schönfeld
2391 Wielenberg
Thorsten Drumhardt
4800 Bielefeld 1
Dr. med. Uve Barmwater
2369 Lentföhrden
Dr. Klaus Perschmann
6752 Winnweiler
Ralf Crumme
5863 Kierspe
Mathias Janke
3100 Celle
Thorsten Beck
3577 Neustadt
Otto Trummer
7750 Konstanz
Michael Hölzel
8000 München 70
Frank Meier
5400 Koblenz
Volker Padéke
4830 Detmold
Emanuel Florakis
1000 Berlin 42
Thomas Seitz
4353 Öt-Erkenschwick
Dirk Rutsatz
2432 Harmsdorf
Thomas Hellinghausen
5245 Brachbach
Thomas Böttge
8236 Eschborn 2
Walter Schlagenhauer
8070 Ingolstadt
Max Rutli
7512 Rheinstetten 2
Klaus Pommerenning
8607 Ingelheim
Jörg Coltermann
8560 Lauf
Rainer Wahl
6250 Limburg 1
R. Bruckhaus
3392 Clausthal-Zellerfeld
Christian Harroth
4158 Wüllich 1
Martin Bauer
8113 Babenhausen 5
Lars Kastrop
3004 Isernhagen 2
Wolfgang Schnurr
7570 B.-Baden
Michael Maltan
8240 Schönaun
Günter Veith
8509 Undenheim
Ellen Streckel
7550 Raastat
Dirk Steitz
8900 Augsburg 22
Jens Steingrübner
8239 Krieth
Frank Goller
7057 Leutenbach 3
Achim Singh
8580 Bayreuth
Michael Schlüter
2380 Schleswig
Thomas Schmidbauer
8000 München 82
Helko Murtfeld
3008 Garbsen 9
Norbert Hochheim
4220 Dinslaken
Georg Matzku
8901 Wiesenbach
Traudi Wahl
6250 Limburg 1
Peter Elsmann
8560 Forchheim
Chr. Pavelskyk-Seyer
5000 Köln 90
Heinke Assig
4150 Krefeld

Gerd-H. Gott
1000 Berlin 20
Michael Clesen
2387 Klappholz
Gregor Mülter
A-4780 Scharding
M. Krauss
3252 Bad Münder 1
Joachim Pils
8300 Landshut
Herbert Spreen
2840 Diepholz
W. Nickel
4020 Mettmann
Oliver Haas
7000 Stuttgart 80
Pia Loch
8769 Cronenberg
Andreas Reller
6110 Aisdorf
Christoph Knobloch
5601 Lorscheid
Markus Gretschmann
8121 Böbling
Dirk Vlöcker
8780 Arnberg 1
Jutta Sackmann
4800 Dortmund 50
Monika Saehar
8580 Lauf/Peg.
Peter Meyer
3111 Wriedel
Sven Matneck
7000 Stuttgart 1
Kleindienst
7000 Stuttgart 1
Thomas Sonntag
5400 Koblenz
B. Korupp
2000 Hamburg 66
Sven Bruns
2190 Cuxhafen
Margareta Ott
7463 Burladingen 6
Axel Dahl
7015 Kornal-Münchingen
Gerd Leitz
7107 Neckarau
Roland Aigner
A-5082 Grödig
Thomas Humberg
2908 Friesoythe
Torsten Buller
1000 Berlin 20
U. Schwarz
4100 Duisburg 11
Dirk Schmitt
8823 Neulüfheim
Michael Ehrreich
A-5541 Altmarkt
Thomas Zimmermann
7000 Stuttgart 80
Gernot Grimm
2100 Hamburg 80
Stephan Wilken
5850 Solingen 19
Linus Neff
7500 Karlsruhe 1
Michael Sürter
4100 Duisburg 1
Dirk Jung
8700 Ludwigshafen
Reinhard Hühn
8090 Rüsselsheim
Barbara Samuel
8700 Ludwigshafen
Monika Scheiner
7144 Asperg
B. Glumm
4600 Dortmund 1
Hans Gauer
6759 Wolfstein/Pfalz
Edda Hirsch
7900 Ulm-Wiblingen
Petra Pfeiffer
2071 Ammersbek 1
Hannelore Heckel
7903 Laichingen 1
Helke Thiel
7150 Becknang

Aktuelle Spiele
heute bestellt - morgen geliefert!

3-D Grand Prix	39.90/49.90
Airwolf	39.90/49.90
Alien 8	34.90
Alien Highway	49.90
Battle of Planets	34.90
Bruce Lee	39.90/49.90
Batman	34.90/49.90
Beach Head II	39.90/49.90
Biggles	39.90/59.90
Bomb Jack	29.90/49.90
Colossus Chess 4.0	39.90
Combat Lynx	29.90
Crafton & Xunk	59.90
Cyrus II Chess	39.90/59.90
Critical Mass	34.90
D. Th. Superast	34.90
D. Th. Decathlon	29.90
Elite (Deutsch)	79.90
Equinox	34.90/49.90
Eden Blues	49.90/59.90
Exploding Fist & Fighting Warrior	59.90
Fighter Pilot (Deutsch)	49.90
Frankie goes to Hollywood	39.90
Friday the 13th	59.90
Frank Bruno's Boxing	49.90
Green Beret	34.90/49.90
They sold a Mill. II	49.90
Gyroscope	34.90
Get Dexter	39.90
Ghost's Goblins	34.90/49.90
The Way of the Tiger	39.90/49.90
Highway Encounter	39.90
Hypersports	39.90/49.90
Hexenküche II	29.90
Harrier Attack	59.90
Hunchback	34.90
Impossible Mission	39.90/49.90
Kung Fu Master	39.90
Knight Rider	39.90
Knight Games	39.90/49.90
Lord of the Rings	69.90
Lord of Middle	39.90/49.90
Movie	49.90
Marsport	34.90/49.90
Match Day	34.90
Match Point	34.90
Misson Elevator	39.90/49.90
Mindshadow	34.90/49.90
Monty on the Run	34.90
N.O.M.A.D.	34.90
Neverending Story	39.90
Panzadrome	34.90
Ping Pong	34.90/49.90
Psychedelia	34.90
Raid	39.90
Rambo	29.90
Saboteur	34.90
ditto & Combat Lynx	59.90
Sky Fox	34.90
Shogun	34.90/49.90
Samantha Fox	34.90/49.90
Strange Loop	34.90/49.90
Space Invasion	29.90
Spiffire 40	39.90/49.90
Spindizzy	49.90
Sweevos World	29.90/49.90
Super Pipeline 2	39.90/49.90
Str. Force Harrier	39.90/49.90
Spy vs Spy	34.90/49.90
The Hacker	49.90
Tau Ceti	34.90
The Goonies	39.90
They sold a Mill.	39.90/49.90
Tomado Low Level	29.90/49.90
Tomahawk	39.90/49.90
Turbo Esprit	34.90/49.90
The Rocky Horror Show	49.90
V+	34.90/49.90
Winter Games	34.90/49.90
World Cup Cam	39.90/49.90
Wanted Gunfight	39.90
Yie ar Kung Fu	34.90/49.90
Zoids	34.90/49.90
Zorro	34.90/49.90

Jedes dieser Spiele ein echter Prellhit

Caves of Doom	9.90	Into Oblivion	9.90
Chiller	9.90	Nonterraquous	9.90
Finders Keepers	9.90	Soul of a Robot	9.90
Formula 1	9.90	Sub-Sunk	9.90
Locomotion	9.90	Wild Bunch	9.90
One man and his Droid	9.90	Trolly Wally	9.90
Nuclear Heist	9.90	Shorts Fuse	9.90
Molecule Man	9.90	Storm	9.90
		Kane	9.90

Achtung!

Eprommer für den Schneider CPC 464/664	Fertigergerät	298.50
ditto in gleicher Ausführung für den CPC 6128		319.50
Bausätze auf Anfrage		
Software auf 3"-Diskette	Aufpreis	15.00
Software auf 5.25"-Diskette	Aufpreis	5.00
Eprom-Karte 264 für den Schneider CPC 464/664	Fertigergerät	249.50
Eprom-Karte 264 für den Schneider CPC 6128	Fertigergerät	259.50
Bausätze auf Anfrage		
Software auf 3"-Diskette	Aufpreis	15.00
Software auf 5.25"-Diskette	Aufpreis	5.00

Bachstraße 52
D-7980 Ravensburg
Telefon 0751/26138
oder 26497



Kleiner Sprachführer Basic Logo Pascal

Von Prof. Fritz Nestle und Eberhard Ostertag
Verlag Markt & Technik
200 Seiten, 39.- DM
ISBN 3-89090-160-3

Das Problem ist immer das Gleiche: Wer in eine Computersprache einsteigt, ist zunächst einmal festgelegt. Bequemlichkeit und Gewohnheit hindern daran, sich vielleicht einmal in die Strukturen einer neuen, besseren Sprache einzuarbeiten. Vor allem die Homecomputerbesitzer, die Basic zum Hausgebrauch gelernt haben, sehen keine Veranlassung, sich in die Tiefen von Logo oder gar Pascal zu begeben, bis sie dann schließlich einen besseren Computer bekommen, der in Pascal oder einer Pascal-ähnlichen Sprache programmiert wird.

Da hat man es doch gut, wenn vorher der "kleine Sprachführer Basic Logo Pascal" durchgearbeitet wurde. Er ist für alle gedacht, die in die Programmierung einsteigen und sich gleich den Horizont für mehrere Sprachen offen halten wollen oder auch für diejenigen, die eine Sprache zur Genüge ausgereizt haben und nach Neuem streben.

Das Buch beginnt mit einer kleinen Vorstellung der Sprachen, um die es im folgenden gehen soll: Basic Pascal Logo. Dann folgt nicht etwa ein seitenlanges Werk mit Tausenden von Vergleichen und trockenen Listings. Nein, es ist ja nur ein kleiner Sprachführer. Er vermittelt gleich Praxiswissen. Das Buch besteht ausschließlich aus elf kleinen Beispielen, die in allen drei Sprachen entwickelt werden.



Zunächst gehen die Autoren auf die Problemstellung ein, legen fest, welche Funktionen ausgeführt werden sollen, zeigen eine Bildschirmmaske, die als Vorbild dient und nennen Formeln, die man benötigt. Bei der Entwicklung in Basic entsteht zunächst das Kernprogramm mit dem reinen Algorithmus, der dann zu einem komfortablen Programm ausgeweitet wird. Es folgen die Entwicklungen in Logo und Pascal, wobei am Ende eines jeden Abschnitts noch einmal die neuen Sprachelemente zusammengefaßt werden.

Die Komplexität der Probleme nimmt im Lauf des Buches immer mehr zu, und am Ende hat man sich schon mit stattlichen Listings zu beschäftigen. Die genaue Erklärung jedes Entwicklungsschrittes wird aber nie vernachlässigt, wobei betont werden muß, daß auf jeden Fall Referenzbücher der einzelnen Sprachen griffbereit sein sollten, da verständlicherweise nicht auch noch die Funktion jedes einzelnen Befehls erläutert werden kann. "Der kleine Sprachführer" ist wohl das nützlichste Einsteigerbuch, das in der letzten Zeit den Buchmarkt betreten hat. Es ist didaktisch gut aufgebaut, gut geschrieben, will nicht zu viel und erreicht sein Ziel mit Bravour. Am Ende kann sich der Leser in die Strukturen von Logo, Basic und Pascal hineindenken, kennt die wesentlichen Unterschiede der Sprachen und kann in der Praxis die richtige Sprache für seine individuelle Lösung herausuchen. Prädikat: Uneingeschränkt empfehlenswert.

Thomas Tai

Sybex-Ratgeber dBase II

Von Gerhard Renner
Sybex-Verlag
360 Seiten, 55.- DM
ISBN 3-88745-305-0

Seien es nun schlichte Anwendungs-Programmierer oder solche, die Unterstutzung bei individuellen Datenverwaltungsaufgaben erwarten: Autor Renner hat mit seinem auf der Grundlage der deutschen 2.41-Version aufgearbeiteten dBase II-Ratgeber für beide Zielgruppen etwas parat. Als Nachschlagewerk konzipiert, garantieren die folgenden Features raschen Zugriff auf Detail-Informationen: übersichtliche Strukturierung des Sprachumfangs nach anwen-



dungsorientierten Teilaufgaben, Querverweise und ein ausführliches Sachwortregister u.v.m. Mit vollständigen Informationen über Dateien, Felder, Variablen, Syntax, Datentypen, Ausdrücke, Operatoren, reservierte Wörter, Funktionen und Befehle sowie deren detaillierte Darstellung will der Ratgeber dazu beitragen, anstehende Datenverwaltungsaufgaben mit dBase II effektiver zu lösen.

Gut gemischt!

Im letzten CPC-Magazin hatten wir bei den Buchvorstellungen das Schneider-CPC-Grafikbuch aus dem Sybex-Verlag und das Buch "Programmmentwicklung unter CP/M 2.2 auf dem CPC 464/664" von Markt und Technik etwas durcheinandergebracht. Die beiden letzten Abschnitte dieser Buchbesprechungen müssen jeweils vertauscht werden. Wir bitten das Versehen zu entschuldigen.

Cracker, Hacker, Datensammler

Von Thomas Tai
Dr. Alfred Hüthig Verlag,
92 Seiten, 24.- DM
ISBN 3-7785-1236-6

Eine völlig neue Richtung schlägt der Hüthig Verlag aus Heidelberg mit dem kürzlich erschienenen Band "Cracker, Hacker, Datensammler" von Thomas Tai ein. Es ist kein - wie sonst beim Hüthig Verlag üblich - wissenschaftliches Fachbuch zum Thema Programmierung und Computeranwendung, sondern ein Sachbuch über Raubkopierer. Dieses Buch versucht, dem



Leser einen Einblick in das Kopierwesen zu geben und wichtige Fragen zu beantworten: Warum wird kopiert? Was unternehmen die Softwarehersteller dagegen? Wie soll man sich verhalten, wenn man "erwischt" wird?

Zu Beginn widmet sich der Autor der "Kopierszene" und erläutert die Unterschiede zwischen "Hacker" und "Cracker". Auch wird hier die Frage geklärt, warum soviel kopiert wird. Danach erfolgt eine Fallbeschreibung anhand einer erdachten Person, um das gesamte Thema noch zu verdeutlichen. Im 2. Kapitel bekommt der Leser einen Einblick in die Entstehung eines Programms und damit verbundene Kosten für die Herstellerfirma. Besonders interessant wird es in Kapitel 3, wo sich der Autor den Folgeerscheinungen zuwendet, mit denen jeder erwischte Kopierer zu rechnen hat. In vielen Beispielen werden die Prinzipien der Abmahnung und die rechtliche Situation eindringlich dargestellt. Der konstruierte Fall (er trägt m. E. unverkennbar autobiografische Züge des Autors) wird an dieser Stelle fortgesetzt. In den beiden letzten Kapiteln gibt Thomas Tai Hilfen für erwischte Kopierer sowie einen Einblick in die derzeitige rechtliche Situation.

Das Buch bleibt bei den behandelten Themen leider zu sehr an der Oberfläche, da es an konkreten und greifbaren Stellungnahmen fehlt. Es würde sicher einen "besseren Beitrag zu einer besseren Verständigung zwischen Kopierer und Softwarefirmen" leisten, wenn der Autor die Meinungen beider Seiten nicht global, sondern subjektiv an einzelnen authentischen Persönlichkeiten herausgestellt hätte. Bemerkenswert ist es hingegen, daß sich endlich ein Autor und ein Verlag mit diesem sehr umstrittenen Thema auseinander-

setzen und ein derartiges Buch veröffentlicht haben. Hier ist sicher Neuland betreten worden. Das Buch ist trotz aller erwähnten Einwände hochinteressant und besonders dem noch nicht erwischten Kopierer sowie dem Leser, der sich über das Thema Softwarepiraterie informieren möchte, zu empfehlen.

Monika Ohlfest

Das Joyce Praxisbuch

Von Joachim Seidler
Signum Medien Verlag
160 Seiten, 35.- DM
ISBN 3-924767-11-4

Nach den üblichen Data-Beker-Erstlingswerken liegt nun ein Buch über den Joyce vor, das nicht aus Düsseldorf kommt. Der Signum-Verlag hatte mit dem vorliegenden Werk wohl den richtigen Riecher, denn auf Unterlagen für den Joyce warten die Besitzer dieses Textsystems schon seit einem dreiviertel Jahr.

Joachim Seidler möchte mit seinem Buch sowohl Neulinge als auch Profis auf dem Computer ansprechen und hat dieses Ziel mit bewundernswerter Hartnäckigkeit verfolgt. Er beginnt mit der obligatorischen Einführung über Schreibsysteme und ihre Rolle im Leben des Menschen. Nachdem noch einige Worte über den Joyce und die Systemphilosophie verloren wurden, folgt der Sprung mitten hinein in die Praxis. Der Joyce wird aufgebaut und die elementarsten Bedienungsschritte besprochen.



Mit nahezu allen Optionen und Menüs von Locoscript wird der Leser in den folgenden Kapiteln

vertraut gemacht. Teilweise werden dabei für den Laien Schritt-für-Schritt-Anleitungen, oder übersichtliche Tastenbelegungstabellen für eingearbeitete Joyce-User abgedruckt. Ebenso interessant für die Profis sind kleine Tips und Tricks, die zu vielen Optionen verraten werden. Beispielsweise nennt der Autor viele nützliche Anwendungen für die Suchen/Ersetzen-Option, auf die man erst einmal kommen muß.

Nachdem Locoscript als solches abgehandelt ist, kommt die Sprache auf die Kommunikationsschnittstelle und den Anschluß von Fremddruckern. Im letzten Kapitel schließlich geht es um den Joyce als CP/M-Maschine. Nach einer ultrakurzen Beschreibung der Dienstprogramme gibt Joachim Seidler einen Überblick über die derzeit lieferbaren Joyce-Programme mit Angabe der Bezugsquelle.

Insgesamt gesehen ist das ein ansehnlicher Informationsgehalt. Nachdenklich stimmt jedoch die Tatsache, daß das alles auf nur 140 Seiten untergebracht wurde und sowohl Profis wie Anfänger bedient werden sollen. So entstand ein Gemisch aus Einsteigeranleitung, die außer einer besseren und leichteren Ausdrucksweise keine Neuerungen gegenüber den Handbüchern bringt. Zusätzlich enthalten sind zwar mehrere kleine Tips für den Routinier, diese sind aber versteckt und nur vereinzelt zu finden. Dadurch konnten die Tips nicht so ausgearbeitet werden, wie es in einem reinen Buch für Profis möglich gewesen wäre. Daß dann auch noch der Bogen zur Softwareübersicht genommen wird, grenzt an Zauberei: drei Bücher in einem. Dagegen wäre wirklich nichts einzuwenden, doch wenn schon, dann bitte auf mindestens 300 Seiten oder durch eine sinnvolle Aufteilung in zwei Bände für Einsteiger und Profis.

Das Buch kann dem Joyce-Besitzer nur dann empfohlen werden, wenn er bereit ist, den Erwerb von vielen unnützen Seiten in Kauf zu nehmen. Am ehesten dürfte der Einsteiger an dem Buch interessiert sein, da es ihn in das System einführt, und nach einiger Zeit wird er auch einige der an sich guten Tips in Anspruch nehmen, soweit er sie in ihrer Kürze versteht.

Thomas Tai

Jetzt endlich!

1. 8-Bit-Druckerkabel inkl. Treibersoftware für 464/664/6128 für jeden Centronics-Drucker (umschaltbar auf 7 Bit). Kein Eingriff am Computer! **98.-**
2. Für DMP 2000 und 664/6128 **Designer**
Eigene Zeichen für den Drucker entwerfen und im Drucker speichern. Zusätzlich Hardcopies in 2 Größen.
Bedingung: 8 Bit und Treiber **49.-**
3. DMP 2000 mit Anschlußkabel **698.-**
4. CPC 6128 Grün-Monitor **998.-**

Pos. 1+2 Paketpreis DM 129.-
Pos. 1+3 Paketpreis DM 729.-
Pos. 1+2+3 Paketpreis DM 759.-
Pos. 1+2+3+4 Superpaket DM 1729.-

Händleranfragen erwünscht!

ton & technik

Visbecker Straße 2
2878 Wildeshausen
Tel. 0 44 31 / 38 88

Philosoft®

TEXTVERARBEITUNG

PAGE=02 LIN=03 COL=45
fett, unterstrichen, kursiv,
hoch und hergestellt auf dem
Bildschirm wie auf dem Drucker!

96 Kommandos: Blockbefehle, Absatz/Seitenumbruch, Suchen/Ersetzen, Druckeranpassung, perfekt, superschnell!

KOMMUNIKATION

Mailboxbetrieb, Textspeicher, Senden und Empfangen mit und ohne Prüfprotokoll (MODEM7 und XMODEM kompatibel)

ASSEMBLER

sehr schneller Assembler für Z80, 8080, 8085 und 8048 (!) Opcodes, 26 Pseudo-Opcodes, if/else, dseg/cseg, rept...

Z80-TESTER

symbolischer Tester mit 26 Funktionen incl. Multi-BP, Datentransfer, EPROM progr.!

AUF EINER KARTE:

32k Software im EPROM (nur 24 Byte RAM weniger!), Programmiergerät (bis 27256) und RS232-Schnittstelle:

Grundausbau **278.- DM**
Option EPROM-Progr. **119.- DM**
Option RS 232 **119.- DM**

Philosoft

Pariser Platz 2
8000 München 80

Über 63000 Besucher bei der 9. PCW-Show in London

Absoluter Magnet war natürlich der Amstrad PC 1512

Diese 9. PCW-Show war mit über 300 Ausstellern in 2 Hallen mit ca. 14000 m² die größte, die das denkwürdige Olympia in London je erlebt hat. Die Ausstellung war in zwei getrennte Bereiche aufgliedert: zum einen die Businessabteilung (z.B. Bürokommunikation) und zum anderen die Personalcomputer. Diese Trennung brachte jedoch einige Doppelanbieter in arge Bedrängnis (das gleiche Problem hatten ja die CeBit-Aussteller in Hannover), deren Produkte zu beiden Lagern gehören. Commodore und Amstrad lösten das Problem dadurch, daß sie sich zwar in der PC-Ebene niederließen, jedoch so nahe wie möglich am Durchgang zur Businessabteilung ihren Platz behaupteten.

Angekündigt als "Großbritanniens größtes Homecomputer-Ereignis", eröffnete die Show am 3. September ihre Pforten, einen Tag, nachdem der Amstrad PC 1512 der Presse vorgestellt wurde. Dieser sorgte allein schon dafür, daß der Besucherrekord vom Vorjahr – 63000 – bei weitem überboten wurde. Sieht man einmal vom Amstrad PC ab, so war Atari die absolute Dominanz (warum sollte es in England anders sein als bei uns). Auf 1000 m² konnte der interessierte Besucher die STs bewundern. Peripherie und Programme wurden von Gastausstellern im "Ataridorf" präsentiert. So tummelten sich über 40 Händler und Dienstleistungsunternehmen unter der Atari-Flagge.

Früher war er einmal der König des Dschungels: der Sinclair-Computer. Heute gibt es zwar noch einen Sinclair-Stand, dieser zeigte sich aber in den Amstrad-Farben und firmierte unter Sinclair Computer Division of Amstrad. Die Show war Anlaß, den neuen alten Sinclair 128 K in einem außergewöhnlichen Gewand zu präsentieren. Der 128 K+2 (bei uns noch nicht erhältlich) hat ein eingebautes Cassettendeck, eine neue Tastatur und 2 Joystickports. Die absolute Spielhalle für das Kinderzimmer. Obendrein bekommen alle lauffähigen Programme von Amstrad ein "Qualitätssiegel" aufgedrückt: SQC – Quality Control System. Die Frage bleibt, ob alles, was auf dem neuen Spectrum läuft, auch qualitativ hochwertig ist. Wie auch immer, dem Amstrad/Sinclair werden in England Bestseller-Chancen eingeräumt. Der einstige Sinclair-Boss, Sir Clive Sinclair, hätte in Anbetracht dieser "enormen Weiterentwicklung" nicht unbedingt für £ 5 Millionen verkaufen müssen. Denn der alte Computerhase hätte mehr getan, als nur ein Tape-Deck angehängt und eine neue Tastatur integriert. Daß diese Transaktion für Alan Sugar nur Taschengeldformat hatte, zeigt sich an dem Umstand, daß Amstrad allein in diesem Herbst £ 2.432.000 an Werbegeld ausgeben will, um dem 128 K+2 Marktchancen zu ermöglichen.

Aber Atari hin, Commodore und Sinclair her, einer stahl allen die Show: der



SAGE-soft Manager David Goldman

PC 1512. Der Amstrad-Stand erwies sich als absoluter Magnet und brachte zeitweise die ganze Ausstellung aus der Bahn. Jeder versuchte, einmal das Wunderding mit eigenen Händen zu fühlen. Frei nach dem Motto: "Und es geschah zu einer Zeit, als Alan Sugar Statthalter von Amstrad war und den Erdenbürgern ein neues Kind vorstellte..."

Ähnlich wie in unseren Ländern, wird auch in England die etablierte Software einen wahren Preisrutsch erleben. SAGEsoft Managing Director David Goldman kündigte drastische Einschnitte in der Preisgestaltung an. Robin Oliver, Chef bei MicroPro, erklärte: "Dieses sagenhafte Preis-/Leistungsverhältnis wird in der ganzen Branche, bei Software und Hardware, Spuren hinterlassen. Die Firma New Star Software hat in dieser Sache gleich ein neues Wort kreiert: Workalike. Billige Clonesoftware für billige PCs. So gibt es in U. K. bereits Low-Price-Versionen von WordStar 2/3, Lotus 1-2-3 und dBase III, um nur einige zu nennen.

Alles auf der Messe vorgestellte Material für den PC aufzuzählen, würde wohl den Rahmen sprengen. Aber wir werden die Bonbons, sobald sie auf dem "Festland" erhältlich sind, ausführlich vorstellen.

Auf dieser PCW-Show sah man einige Highlights, die den Eintritt in ein neues Computer-Zeitalter markieren. SAGEsoft Manager David Goldman faßte alles zusammen, wenn er Alan Sugar zitiert: "The micro business will never be the same!"

Peter Cornforth



Über 63000 Besucher wollten das Angebot der mehr als 300 Aussteller in den Hallen des Londoner Olympia begutachten. Absoluter Magnet war der Amstrad PC 1512.

Telefonische Bestellung

Vortex

VERSANDSERVICE

Tel. 07131
52065

Kompetente Hard- und Software sowie Zubehör aus einer Hand!
Alle Produkte sind von uns getestet und entsprechen unserem hohen Qualitätsstandard.

SUPER - GÜNSTIG - SUPER - GÜNSTIG - SUPER - GÜNSTIG

5 1/4" VORTEX-DISK DS/DD 96 tpi 10 Stk. 59,-- DM
5 1/4" VORTEX-DISK DS/DD 135 tpi 10 Stk. 79,-- DM
3" DISK CF-2 5 Stk./10 Stk. 49,50/89,-- DM
3" DISK CF-2DD (Joyce 8512) 5 Stk. 99,-- DM
Wir führen nur erstklassige, geprüfte Disketten, Deshalb können Sie die Disketten innerhalb von 14 Tagen ohne Angabe von Gründen umtauschen.

DISKETTENBOXEN

3" Diskbox für 10 Disketten 14,80 DM
3" Diskbox für 40 Disketten 38,90 DM
do, abschließbar 53,90 DM
5 1/4" Diskbox für 50 Disketten 36,90 DM
5 1/4" Diskbox für 85 Disketten abschließbar 39,90 DM
3 1/2" Diskbox für 40 Disketten 38,90 DM
3 1/2" Diskbox für 80 Disketten 44,90 DM

PFLEGEMITTEL

ORIGINAL VORTEX-ABDECKHAUBEN:
Schneider Floppy DDI-1 16,80 DM
VORTEX Floppy F1-S o. F1-D 19,80 DM
Schneider Konsole für 464 u. 664 19,80 DM
Schneider Konsole für 6128 19,80 DM
VORTEX Floppy F1-X u. M1-X 19,80 DM
Schneider Monitor grün 24,80 DM
Schneider Monitor color 26,80 DM
Schneider NLQ 401 19,80 DM
Schneider DMP 2000 22,80 DM
VORTEX Winchesterlaufwerk WD-20 19,80 DM

Reinigen Sie Ihr 5 1/4"-Laufwerk mit unserem speziellen Reinigungsset
5 1/4" Kopfreinigungsdiskette 13,90 DM

Kassettenreinigungsset bestehend aus:
1 Prüfkassette, 1 Reinigungsmittel für Kapstan- und Gummi-antrieb, 1 Tonkopfreinigungslösung und mehrere Reinigungsstäbchen. Einführungspreis: 10,90 DM

Azimuth: Ein Programm zur genauen Einstellung Ihres Kassettenrecorders am CPC 464. Kein Read Error mehr durch einen verstellten Tonkopf. Mit Schraubenzieher und einem Testspiel. Preis: 39,90 DM

ANDRUM:

DIGITALES SCHLAGZEUG FÜR CPC 464/664/6128
- 8 digital aufgenommene Schlagzeugsounds
- einfach zu bedienen
- ein echter digitaler Drumcomputer
- Echtzeitprogrammierung am Bildschirm
- komplett mit Software
- andere Sounds können von Kassette geladen werden
- für die Kreativität, zum Lernen und zur Freude
- eine der besten Peripherien für die CPC's
- bis zu 1000 Rhythmen speicherbar
- Lieder können auf Kassette/Diskette gesichert werden
- Polyphonie
- ausführliche Anleitung
- einfach an einen HI-FI- oder PA-Verstärker anschließen

VERBINDUNGSKABEL:

Druckerkabel:
für CPC 464, 664 (1m Länge Flachbandkabel) 39,-- DM
für CPC 464, 664 (2m Länge Flachbandkabel) 44,-- DM
für CPC 6128 (1m Länge Flachbandkabel) 39,-- DM
für CPC 6128 (2m Länge Flachbandkabel) 44,-- DM
für CPC 6128 (1,5m Länge Rundkabel) 49,-- DM
Akkustikkopplerkabel (zw. RS 232 u. Modem) 1,5m 49,50 DM
Anschlusskabel: 2 Floppy an CPC 664: 44,50 DM
Anschlusskabel: 2 Floppy an CPC 6128: 44,50 DM
Floppyverlängerungskabel für alle CPCs (2m): 59,-- DM
Monitorverlängerung für CPC 464: 22,90 DM
Monitorverlängerung für CPC 464 und 6128: 28,90 DM
Joystickverlängerung für 1 Joystick (2m Länge) 14,90 DM
Recorderanschluß (CPC an 5pol. DIN Buchse) 17,90 DM
Recorderanschluß (CPC an Klinkenbuchse) 17,90 DM
CPC-Stereokabel zum Anschluß an HI-FI-Anlage 15,90 DM
Multi-Link-Kabel:
Programmierbares RS-232 Kabel (s. Seite 2) 69,90 DM

DFÜ (Datenfernübertragung)

VORTEX-VAK-300:
Übertragungsgeschwindigkeit: 300 Baud
Originale-/Antwortmodus
Stromversorgung: 9V Blockbatterie / externes Netzteil
Preis: 198,-- DM

VORTEX-RS-232 Schnittstelle:
Ins ROM eingebundene Software mit RSX-Befehlen, unter anderem: Einstellung der Parameter, Übertragung (senden und empfangen) von: einzelnen Zeichen und Zeichenketten (Strings), ASCII-Files mit und ohne Softwareprotokoll, serielle Druckersteuerung (Plotter) und integriertes Terminalprogramm. Unter CP/M: sofortige Einbindung mittels der im AMSDOS und VDOS 2.1 enthaltenen Treiberrouitinen. Mit ausführlichster deutscher Dokumentation (ca. 60 Seiten) und Anschlußkabel. Zeigen Sie uns eine RS-232 Schnittstelle, die bei diesem Preis mehr bietet: 298,-- DM

VORTEX-Phono-Set bestehend aus:
Akkustikkoppler VORTEX-VAK-300, Schnittstelle VORTEX-RS-232, Zeitteil zur Stromversorgung, Diskettensoftware und Verbindungskabel. Ihr Vorteil: Alles aus einer Hand, d. h. keine Kompatibilitätsprobleme. Nur auspacken, anschließen und „datenfernübertragen“.
SONDERPREIS: 498,-- DM

Adapter Null-Modem RS-232:
Bei Datenübertragungen ohne Modem wird das Null-Modem eingesetzt um die Verbindung zwischen Endgerät und Computer herzustellen. Dieses wird möglich durch Kreuzung von drei Leitungen. Einsatzbegrenzung bis 50m Länge.
Preis: 69,-- DM

NEU - NEU
- SOFTWARE - SOFTWARE - SOFTWARE - SOFTWARE -

SPIELE

PAPERBOY C/D 36,--/54,-- DM
1942 C/D 36,--/54,-- DM
LIGHT FORCE C/D 36,--/54,-- DM
BOMB JACK C/D 32,--/45,-- DM
GHOST'N GOBLINS C/D 29,90/39,90 DM
LEADER BOARD C/D 39,--/49,-- DM
GREEN BERET C/D 35,--/45,-- DM
BAT MAN C/D 35,--/45,-- DM
EQUINOX C/D 35,--/45,-- DM
KNIGHT GAMES C/D 34,90/49,90 DM
HEXENKÜCHE Teil II C/D 29,--/45,-- DM
IMPOSSIBLE MISSION C/D 39,--/59,-- DM
SHOGUN C/D 39,--/59,-- DM
MISSION ELEVATOR C/D 39,--/59,-- DM
GOONIES C/D 39,--/59,-- DM
REBEL PLANET C/D 39,--/59,-- DM
QUESTPROBE C/D 39,--/59,-- DM
INTERNATIONAL KARATE C/D 29,90/39,90 DM
STREET HAWK C/D 39,--/59,-- DM
EDEN BLUES C/D 19,90/39,90 DM
CRAFTON & XUNK (GET DEXTER) C/D 19,90/39,90 DM
MELTDOWN C/D 27,90/34,90 DM
WANTED GUNFRIIGHT C/D 39,--/59,-- DM
JACK THE NIPPER C/D 24,90/34,90 DM
SUPERBOWL C 34,90 DM
GLADIATOR C/D 39,90/54,90 DM
MANDRAGORE C/D 39,90/49,90 DM
NEXUS C/D 39,90/49,90 DM
NIGHT GUNNER C/D 29,90/39,90 DM
PACIFIC C/D 39,90/49,90 DM
RESCUE ON FRACTALUS C/D 39,90/49,90 DM
ROOM TEN C/D 39,90/49,90 DM
SAI COMBAT C/D 39,90/49,90 DM
SPACE SHUTTLE C/D 29,90/49,90 DM
DAN DARE C/D 29,90/42,90 DM
GLASS C/D 19,90/39,90 DM

Suchen Sie einen guten Drucker oder andere Hardware?
Rufen Sie uns an. Wir senden Ihnen nähere Informationen und machen Ihnen zudem einen günstigen Preis!

Perforiertes Druckerpapier:
300 Blatt EDV-Endlospapier im Tragekarton 14,90 DM
1000 Blatt 240mm x 12" endlos, holzfrei 60g 29,90 DM

Druckerständer:
Papierzuführung wahlweise von unten oder hinten, sehr stabile Ausführung, Nutzfläche 40 x 32 cm 55,-- DM

Traktorführung:
Zum problemlosen Verarbeiten von Endlos-EDV-Papier.
Für Schneider NLQ 401: 69,90 DM
Für Okidata ML 182/192: 79,-- DM

Midi-Schnittstelle:
Endlich erhältlich! Das Midi-Interface für Ihren CPC. Mit diesem Interface von EMR können Sie nun Ihren Schneider an einen Synthesizer anschließen.
Beachten Sie den Testbericht in CPC International (Oktoberausgabe).
Preis (incl. Software): 399,-- DM

NÜTZLICHES ZUBEHÖR:

Plastikschablone CPC 464: Legen Sie die Schablone auf Ihre Tastatur und die ewige Blätterei im Handbuch entfällt. Folgende Anweisungen befinden sich auf der Schablone: Basic Anweisungen, Ein-/Ausgabenweisungen, logische Operationen, mathematische Operationen, Syntax Notizen, Basic Befehle, Basic Funktionen, numerische Funktionen, Zeichenketten, Farbtabelle und AMSDOS-Befehle.
Preis: 27,90 DM

Kabelschellen: Schluß mit dem Kabelsalat unter Ihrem Computertisch. Benutzen Sie unsere selbstklebenden Kabelschellen. Kein Beschädigen der Tischoberfläche. In den Farben weiß, schwarz und braun lieferbar.
Verpackungsinhalt: 16 Stück 8,90 DM

VORTEX-Monitorständer: Dreh- und schwenkbar in allen Richtungen. Für alle 12"-Monitore. Solide Ausführung aus bruchfestem Kunststoff.
Preis: 49,90 DM

Monitor-Entspiegelungsspray: Beseitigen Sie lästige Reflexionen sofort und dauerhaft. Die Augen werden spürbar entlastet. Auch für andere Anwendungen geeignet, wie z.B.: Bilderglas, Autoarmaturen, Fernsehgeräte und ähnliche optische Anwendungen. Siehe Sonderprospekt.
Preis: 29,90 DM

Bildschirmfilter: Vorsatz-Filter, reduziert störende Spiegelungen und Flimmern, wird mit einem Klebeband-Klettverschluss am Gehäuse befestigt. Filter aus Gewebe auf Kunststoffrahmen.
Für Farbmonitor CTM 640/644 79,-- DM
Für Grünmonitor GT 64/65 (bis 12/85) 59,-- DM

Datenrecorder: Zum Laden und Speichern von Kassettensoftware auf dem CPC 664 und CPC 6128. Im Preis ist das Datenübertragungs- und das Netzkabel enthalten. Auch für Batteriebetrieb geeignet und als normaler Musikrecorder verwendbar.
Preis: 89,-- DM

Micro-T-Schalter: Ein Schnittstellenumschalter mit dem Sie 2 Drucker an 1 Computer (oder umgekehrt) anschließen können. Einfache Drucktastenumschaltung, auch für alle anderen Peripheriegeräte. Optional mit RS 232/V24 oder Centronics-Schnittstelle.
Preis: 169,-- DM

Selbstverständlich erhalten Sie über den Vortex-Versand-Service auch alle Vortex-Produkte

Händleranfragen erwünscht!
Achtung Programmierer: Haben Sie gute Spiele (keine Adventures) oder Utilities, setzen Sie sich mit uns in Verbindung. Wir sind ein fairer Partner.

Telefonische Bestellung von 8-12 Uhr und von 13-17 Uhr.
Außerhalb der Geschäftszeiten nimmt unser Anrufbeantworter Ihre Bestellung entgegen.

vortex Computersysteme GmbH
Versandservice
Falterstraße 51-53 · 7101 Flein
Telefon 07131/52065
Telex 728 015 vortex

Senden Sie mir Ihren Katalog (Schutzgebühr DM 3,-, bei Bestellung ab DM 100,- frei)

Senden Sie mir umgehend folgende Artikel aus Ihrem Angebot:

per Nachnahme per Euro-Scheck

_____ DM

_____ DM

_____ DM

_____ DM

bei Aufträgen bis DM 200,- Versandkostenpauschale DM 5,90 _____ DM

Absender: _____ Gesamtsumme _____ DM

Telefon-Nr. _____ Unterschrift _____

Grafikkünstler aufgepaßt!

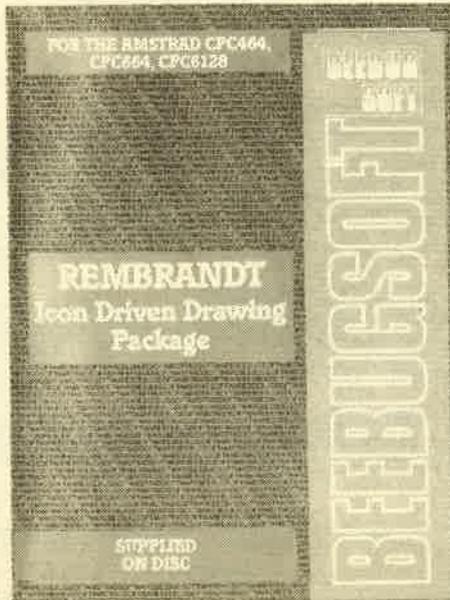
"Rembrandt" ist wohl das beste Grafikprogramm, das es zur Zeit auf dem Markt gibt

Der Hersteller muß schon an sein Produkt glauben, wenn er ein Programm mit einem solchen Namen ausstattet. Das tun natürlich alle Software-Anbieter, zumindest in der Werbung. Hier steckt aber doch mehr dahinter. Wer bisher mit seinem Computer zeichnen wollte, hatte schon arge Probleme. Da gab es den "Profi Painter", der anfangs für 198.- DM viel zu teuer angeboten wurde und meines Erachtens nicht so toll in der Leistung war. Gute Konkurrenzprodukte wurden auch nicht angeboten. Daher konnte man nur abwarten oder ein eigenes Programm erstellen.

Wer bis heute gewartet hat, wird nun belohnt. Um es vorwegzunehmen, ich halte "Rembrandt" für das zur Zeit beste Grafikprogramm. Im Detail gibt es sicher noch einiges zu verbessern. Aber wo gibt es das nicht? Das Preis/Leistungs-Verhältnis, die Bedienerfreundlichkeit und die Möglichkeiten, die dieses Programm bietet, sind enorm. Jetzt aber genug des Lobes, es folgen die Tatsachen.

"Rembrandt" läuft laut Herstellerangabe auf den CPC-Modellen 464/664/6128; getestet wurde es auf dem 464. Zum Lieferumfang gehört eine Diskette nebst englischem Handbuch (laut deutschem Vertrieb soll es auch eine deutsche Kurzanleitung geben), alles verpackt in einer stabilen Plastikbox. Wahlweise kann man auch eine Cassettenversion erhalten. "Rembrandt" kostet auf Diskette 69.90 DM, auf Cassette 59.90 DM.

Das englische Handbuch umfaßt 17 Seiten, auf denen alle Möglichkeiten des Programms kurz und klar erläutert werden. Da es "icon driven", also symbolgesteuert ist, kann man aber auch ohne Anleitung vieles herausfinden. Nach Programmstart erscheint auf dem Monitor die Arbeitsfläche, die bis auf einen kleinen Streifen am oberen Bildschirmrand leer ist. Dieser Streifen enthält 8 Symbole, die einige "Rembrandt"-Optionen darstellen, wovon insgesamt 24 vorhanden sind. (Die Symbolleiste kann mit der TAB-Taste umgeschaltet werden.) Die einzelnen Symbole sind grafisch relativ gut dargestellt. So kann auch der Anfänger die Bedeutung der meisten Zeichen schnell erraten bzw. erkennen (z.B. Mülleimer = Clear Screen; Diskette = Load, Save, Cat; Stift = Zeich-



nen etc.). Die Symbolleiste kann vom oberen auf den unteren Bildschirmrand gesetzt oder ganz ausgeschaltet werden.

Zum Zeichnen läßt sich also die gesamte Bildschirmfläche nutzen, was dem normalen CPC-Screen-Format entspricht. Zeichnen und Wählen der Symbole erfolgt über die Cursor-Tasten oder einen Joystick, wobei man sich nicht festlegen muß; beides ist jederzeit möglich. Das halte ich für eine gute Lösung, da man z.B. Feinarbeiten lieber über die Tastatur ausführt, für andere Dinge aber den Joystick benötigt. Im Grunde gibt es bei "Rembrandt" einen Modus zum Zeichnen und einen für die Symbolsteuerung. Umgeschaltet wird durch einmaligen Druck auf die ESC-Taste. Ein normaler Arbeitsgang sieht wie folgt aus.

1. Zeichnen mit dem Stift auf dem Bildschirm.
2. ESC-Taste drücken.
3. Mit der TAB-Taste eine der drei gewünschten Symbolleisten aufrufen.
4. Einen kleinen Pfeil auf das gewünschte Symbol bewegen.
5. Mit der SPACE-Taste bestätigen.
6. Mit der nun gewählten Option weiterarbeiten.

Das klingt hier etwas umständlich, geht aber in der Praxis nach kurzer Einarbeitungszeit schnell über die Bühne.

Kommen wir jetzt zu den einzelnen Symbolen und ihrer Bedeutung.

1. "Punkt und Linie" nennt sich die erste Option, die freihändiges Zeichnen ermöglicht und damit den wichtigsten Teil des Programms darstellt. Gezeichnet wird bei Druck auf die Feuer- oder Space-Taste; ohne Betätigung dieser Tasten kann der Cursor frei über den Bildschirm bewegt werden.
2. "Linie" stellt eine flexible Linie dar, deren Anfangspunkt markiert werden muß. Dann erscheint hinter dem Cursor eine blinkende Linie, die man beliebig verlängern und verschieben kann, bis sie paßt und der Endpunkt fixiert werden kann.
3. "Rechteck" spricht für sich. Rechtecke beliebiger Größe sind damit möglich.
4. "Gefülltes Rechteck" heißt die nächste Option, die weitgehend Punkt 3 entspricht. Das Rechteck wird aber mit der aktuellen Zeichenfarbe ausgefüllt.
5. "Circle" sowie "Solid Circle" dienen dem Erstellen von Kreisen analog den Punkten 3 und 4.
6. "Triangle" ist die nächste Option im Bereich der geometrischen Zeichen, zu denen auch noch Punkt 7, "Polygon" (Vieleck), zählt. Angewendet werden sie wie "Rechteck" und "Kreis", eine Füllfunktion ist hier aber nicht vorgesehen.

Damit wäre das Ende der ersten Symbolleiste erreicht, und nun zur nächsten.

1. Hinter "Fill Shape" verbirgt sich eine Fill-Routine, mit der beliebige Formen (auch Triangle und Polygon) mit einer Farbe nach Wunsch ausgefüllt werden können. Wenn das Resultat nicht gefällt, kann der Vorgang wieder aufgehoben werden.
2. "Text" ermöglicht das Einsetzen von Textstellen in die Grafik. Auch selbstdefinierte Zeichen sind möglich. Nachdem der Text auf den Bildschirm geschrieben wurde, kann er noch beliebig positioniert werden, was ich sehr gut finde.
3. "Air Brush", die Sprühpistole, ermöglicht schöne Effekte. Um den Cursor

werden Pixel gesetzt, so daß man je nach Sprühdauer feine Sprenkel bis volle Flächen erreicht. So werden Stoppelbärte produziert.

4. "Paint Brush" ist ein Pinsel, der im Gegensatz zum normalen Schreibstift in seiner Breite verändert werden kann.
5. "Grid" erlaubt das Einstellen der Cursorschrittweite von 1 bis maximal 32 Pixel.
6. "Magnify" stellt die Lupe dar. Nach Aufruf dieses Symbols erscheint auf dem Bildschirm ein Rechteck, das über Cursor-Tasten oder Joystick an die zu vergrößerte Stelle bewegt wird. Nach Druck auf die Space-Taste wechselt der Bildschirm, und die Ausschnittsvergrößerung (8-fach) erscheint. Jetzt können Feinarbeiten vorgenommen werden. Leider kann man das Rechteck selbst nicht variieren und ist deshalb auf die vorgegebene Form angewiesen.

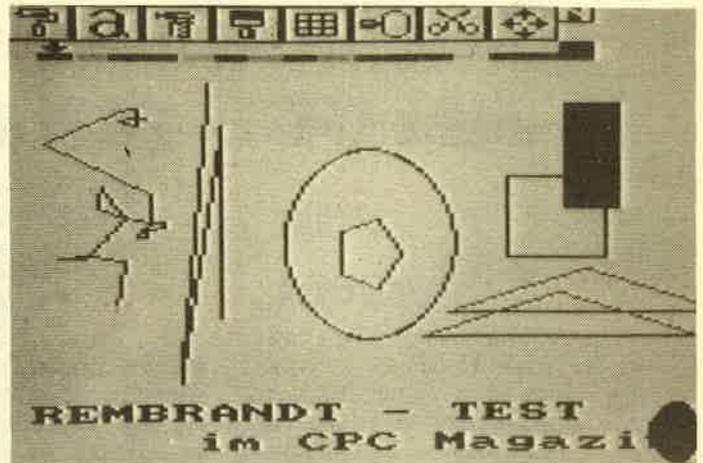
7. "Out and Paste" dient dem Bewegen und Kopieren eines Bildausschnitts. Wie bei der Lupe kann auch hier wieder ein Rechteck über das Bild geführt werden, um den Ausschnitt festzulegen. Danach ist zu entscheiden, ob dieser gelöscht und an anderer Stelle wiedergegeben oder an eine beliebige Stelle kopiert werden soll. Besonders die Kopier-Funktion kann die Arbeit unter Umständen erheblich erleichtern, wenn z.B. ein mühevoll erstelltes Detail mehrfach benötigt wird.

8. "Scroll", die letzte Option in der zweiten Symbolleiste, ist auch wieder vom Feinsten! Wird dieses Symbol gewählt, verschwinden die Leiste und der Cursor, so daß der Blick auf das ganze Bild frei wird. Über Joystick oder Cursorblock kann es nun pixelweise verschoben werden. Am Bildschirmrand verschwindet das Bild, um an der anderen Seite wieder aufzutauchen. Der Scrollvorgang zerstört also nichts.

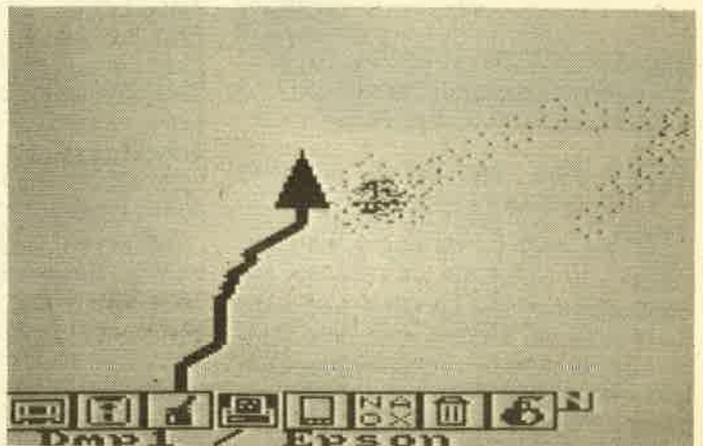


Dies ist ein kleiner Ausschnitt aus dem Bild rechts oben mit Hilfe von Magnify

Hier kann man deutlich die 2. Symbolleiste mit den Symbolen Fill Shape, Text etc. erkennen



Hier wurde in Symbolleiste 3 die Hardcopyfunktion angewählt. Epsonkompatible Drucker sollten für diese Funktion genügen. Allerdings konnte ein positiver Test nur mit einem CPA 80 GS durchgeführt werden.



Kommen wir nun zur dritten Symbolleiste.

1. "Cassette" ermöglicht die Wahl zwischen LOAD, SAVE und CAT.
2. "Diskette" entspricht Punkt 1, dient aber für Diskettenoperationen.
3. "Palette" ermöglicht das Austauschen der Zeichenfarben. Alle 27 CPC-Farben können gewählt werden.
4. "Screen Dump" verrät ebenso wie das Symbol (ein Drucker), daß "Rembrandt" damit eine Hardcopy-Funktion zur Verfügung stellt, die mit den Schneider-Druckern und Epson-kompatiblen Geräten zusammenarbeitet. Diese Option halte ich für die schwächste des Programms. Der Ausdruck mit einem Epson-kompatiblen Drucker (CPA 80 GS) funktioniert zwar einwandfrei, doch erscheint er invertiert, d.h., was auf dem Schirm weiß war, wird schwarz und umgekehrt. Das ausgedruckte Bild hat bei mir ein Format (Breite 15,2 cm, Höhe 13,4 cm), das leider nicht den CPC-Proportionen entspricht. Das Bild wird in die Breite gequetscht, wodurch unschöne Eierköpfe entstehen. Mir stand kein Schneider-Drucker zur Verfügung, so daß ich nicht genau weiß, ob es an meinem Gerät gelegen hat.

5. "Mode Change" ist eine Option, die ich bei anderen Grafikprogrammen immer vermißt habe. Hier kann der Anwender wählen, in welchem der drei verfügbaren Modi er arbeiten möchte. Danach richtet sich natürlich dann auch die Anzahl der möglichen Farben.
6. "Ink Modes" stellt ebenfalls eine Besonderheit dar. Vier Möglichkeiten werden geboten: N = Normalmodus, A = And-Modus, O = Or-Modus, X = Exclusive or-Modus.
7. "Clear Screen" steht für den schon zu Anfang genannten Mülleimer. Bei Aufruf wird der ganze Bildschirm gelöscht.
8. "Quit Rembrandt" hat sinnigerweise das Symbol einer kleinen Bombe erhalten, womit "Rembrandt" gelöscht werden kann.

Damit sind alle Symbole aufgeführt. Durch die Art der Programmgestaltung und durch die Anleitung läßt sich's mit "Rembrandt" schon nach wenigen Minuten arbeiten. Abgesehen von der Drucker-Routine funktioniert alles hervorragend. Da auch der Preis stimmt, gibt es für Interessenten keinen Grund, dieses Programm nicht zu erwerben.

System: CPC 464/664/6128
 Hersteller: Beebugsoft
 Bezugsquelle: PiZie-Data
 Rolf Knorre



Nach dem "Star-Texter" erscheint nun im Sybex-Verlag ein neues Programm für den CPC, die Datenverwaltung "Star-Datei". Da sich auf diesem Gebiet seit der Einführung des Schneider-Computers einiges getan hat, steht das Programm beileibe nicht ohne Konkurrenz da. Ob es zur Spitzengruppe gehört und mit der Konkurrenz mithalten kann, soll dieser Test zeigen.

Gleich nach dem Öffnen der Packung kann das Programm schon den ersten Pluspunkt verbuchen, denn es liegt ein deutsches Handbuch bei, das sogar gebunden und recht umfangreich ist. Dieser Eindruck erweist sich aber bald als trügerisch, denn das ganze Büchlein ist in einer relativ großen Schrift gesetzt und mit Abbildungen, Tabellen und großzügigen Absätzen gespickt. Diese Platzschinderei kann man aber noch entschuldigen, da der Inhalt wirklich stimmt. Es wird eine hervorragende Einführung in "Star-Datei" gegeben, wobei alle Features ausführlich zur Sprache kommen, nur sollte sich der Autor die unzähligen Verweise auf noch folgende Kapitel abgewöhnen. Das irritiert den Leser, und man muß erst etliche Seiten durcharbeiten, bis man mit dem geladenen Programm experimentieren kann.

Wenden wir uns nun endlich "Star-Datei" zu. Zunächst einmal sitzen wir vor einem leeren Bildschirm, auf dem nur die Menüzeile sichtbar ist. Das Programm ist voll menügesteuert, und wer nun endlich loslegen will, wählt "Neue Datei" an. Wie bei Dateiprogrammen üblich, beginnt es auch hier mit der Definition der Maske. Zu diesem Zweck steht dem Anwender fast der ganze Bildschirm zur Verfügung. Er kann den Monitor nach Belieben mit Zeichen, Kommentaren und sonstigen Verweisen füllen; lediglich die Kleiner-, Größer-Zeichen sowie der Punkt dürfen nur zur

Star-Datei auf dem Prüfstand

Nach dem Star-Texter gibt es vom Sybex-Verlag jetzt auch ein Dateiverwaltungsprogramm

Felddefinition benutzt werden. Weitere Angaben zu den Feldern sind nicht notwendig. Das Programm will beispielsweise nicht wissen, ob es sich um ein numerisches oder alphanumerisches Feld handelt. Das ist allerdings auch nicht nötig, denn Rechnen oder sonstige Manipulationen mit den Feldern sind nicht möglich. Für den privaten Gebrauch mag das keine Rolle spielen, doch stellt es den Einsatz im professionellen Bereich stark in Frage.

Kehren wir zur Erstellung der Maske zurück. Insgesamt kann "Star-Datei" 1000 Datensätze pro Datei verwalten, wobei pro Datensatz höchstens 512 Zeichen und pro Datenfeld 254 Zeichen erlaubt sind. Das sind wohl Werte, die im Normalfall nicht überschritten werden dürften. Ist die Erstellung der Maske abgeschlossen, errechnet "Star-Datei" die Felder und eröffnet die Datei auf der Diskette. Hierbei muß noch als weitere Besonderheit darauf hingewiesen werden, daß die Daten in einer Random-Access-Datei abgelegt werden. Diese Dateiarart, die beim CPC gar nicht vorgesehen ist, ermöglicht einen direkten und damit schnelleren Zugriff. Namen, Adressen usw. werden also nicht sequentiell abgelegt und müssen deshalb nacheinander gelesen werden, sondern das Programm kann auf jeden beliebigen Datensatz – ob am Anfang oder Ende der Datei – einzeln zugreifen.

◆ 1000 Datensätze

Nun können wir mit der Dateneingabe beginnen. Wir müssen nur noch bedenken, daß wir zuvor das Index-Feld festgelegt haben, nach dem sich die Sortierung des Feldes richtet. Sollte sich hinterher erweisen, daß die Wahl nicht richtig war, kann das Index-Feld nachträglich (!) verändert werden. Die Dateneingabe gestaltet sich recht simpel; die Felder werden automatisch vom Cursor angesprungen, und man muß die Daten nur noch eintippen.

Die Leistungsfähigkeit einer Dateiverwaltung offenbart sich in den Auswertungsmöglichkeiten. Auch "Star-Datei" bietet hier einiges. Neben auf- und ab-

steigendem Sortieren nach dem Index ist die Suche nach bestimmten Kriterien möglich. Es können bis zu 5 Bedingungen angegeben werden, die das gesuchte Feld erfüllen bzw. nicht erfüllen soll. Zwischen diesen Bedingungen ist eine der Verknüpfungen möglich, die in einer sog. Verknüpfungstabelle angegeben werden. So kann man beispielsweise Datensätze herausuchen, die bestimmte Kriterien nicht erfüllen, dafür aber bis zu drei andere Bedingungen erfüllen.

Wer seine Daten schwarz auf weiß sehen will, wird ebenfalls nicht im Stich gelassen; ein Ausdruck der Indexfelder, beliebiger Datensätze und Mail-Merge mit "Star-Texter" sind möglich. Besonders erwähnenswert ist die Möglichkeit, ein eigenes Formular zur Ausgabe der Datensätze zu definieren. Wer sich also nicht unbedingt auf die Wurstelei mit "Star-Texter" einlassen will, kann hier beispielsweise ein Rechnungsfeld erstellen und mit selektierten Datensätzen füllen.

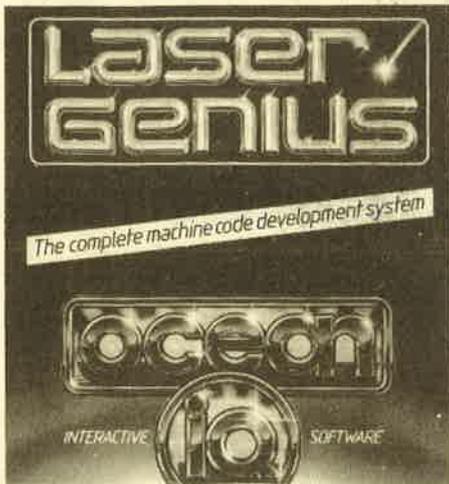
Insgesamt kann man sagen, daß "Star-Datei" eine Menge an Leistung für einen sehr realen Preis bietet. Es stellt sich aber die Frage nach Einsatzgebieten für dieses Programm. Hier muß ganz klar gesagt werden, daß der professionelle Einsatz einige Probleme mit sich bringen wird, da keine Rechenfunktion vorhanden ist, die Ausgabemöglichkeiten zu begrenzt sind, die Maske sich nur über eine Bildschirmseite erstrecken kann und ein Datum-Feld fehlt. Schallplattenarchiv, Bücherverwaltung und Videodatei (diese Masken sind übrigens schon vorher eingestellt) sind hervorragende Anwendungen. Bei einer richtigen Lagerdatei oder Kundenverwaltung wird es dann aber schon kritisch. Den Erfordernissen eines kleinen bis mittelgroßen Betriebes kann das Programm nicht gerecht werden.

Dem privaten Anwender ist "Star-Datei" also ohne Gewissensbisse, dem professionellen Anwender mit einigen Vorbehalten zu empfehlen.

System:	CPC 464/664/6128
Hersteller:	Sybex-Verlag
Preis:	85.- DM
Bezugsquelle:	Sybex-Verlag
Thomas Tai	

Laser Genius – das geniale Assemblerpaket

„Laser Genius“ bringt eine erhebliche Arbeitserleichterung



Die Programmierung in Maschinensprache ist und bleibt eine mühselige Sache, denn Systemabstürze und eine anschließende nervenaufreibende Fehlersuche treiben den Entwickler nicht selten an den Rand des Wahnsinns. Aber was soll man machen? Nur Programme in Maschinensprache weisen die Geschwindigkeiten auf, die man oft benötigt. Das Ziel der Entwickler von Assemblern war und ist die Verbesserung und Erleichterung der Codeeingabe und vor allem der Fehlersuche. Die Eingabe des Sourcecodes ist mittlerweile so komfortabel wie bei einer richtigen Textverarbeitung, doch eine effiziente Fehlersuche war bisher noch nicht möglich.

Mit "Laser Genius" hat sich die Situation nun grundlegend geändert. Das

Programmpaket versteht sich nicht als Assembler mit halbherzig beigefügtem Monitor, sondern als Assembler, Editor, Monitor und Debugger-Paket. Alle Teile sind gleichberechtigt und wurden daher auch mit derselben Sorgfalt programmiert.

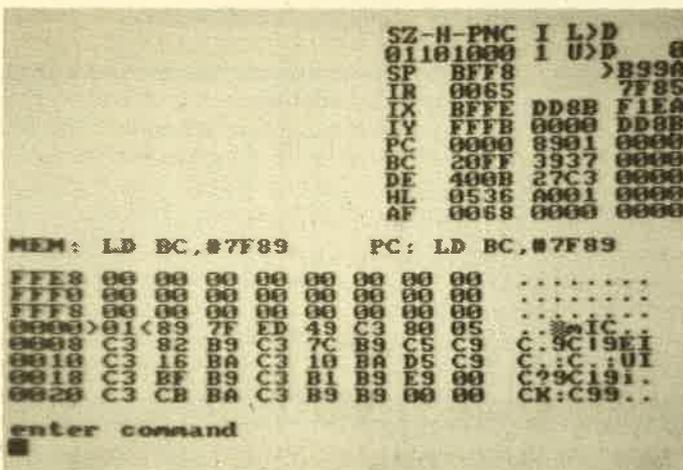
Wie die normale Entwicklung eines Programms soll auch unsere Betrachtung der Features von "Laser Genius" mit dem Editor beginnen, der wirklich hervorragend ist. Der Code wird wie ein Basic-Programm mit Zeilennummern geschrieben und auch so editiert. Umfangreiche Befehle für die Cursorbewegung und sogar eine Find/Replace-Funktion stehen zur Verfügung. Besonders auffällig ist aber der Speicherplatz. Mehrere tausend Zeilen Assemblercode finden im CPC Platz, denn "Laser Genius" wandelt den Code in sog. Tokens um. Für jeden Assemblerbefehl steht nun ein Code, der nur ein Byte Speicherplatz benötigt. Das wirkt sich natürlich auf die Geschwindigkeit des Assemblierungsvorgangs aus. Der Hersteller gibt an, daß 20000 Zeilen Code in der Minute umgesetzt werden können. Wird von Diskette assembliert, sind es immerhin noch 8500 Zeilen pro Minute. Nun brauchen die Besitzer anderer Assembler aber nicht enttäuscht zu sein, denn der Editor erlaubt auch das Laden von Sourcecode, der mit einem anderen Assemblerpaket erstellt wurde.

Der Assembler befindet sich gleichzeitig mit dem Editor im Speicher. Er bietet eine Fülle von Fähigkeiten, die an

dieser Stelle gar nicht vollständig aufgezählt werden können. Auf die wichtigsten wollen wir jetzt aber eingehen. "Laser Genius" erlaubt die Definition von sog. Makros, das sind Programmteile, denen ein Name gegeben wird, den man dann an der Stelle des Codes einsetzt, wo die definierte Routine stehen soll. Man kann auch eine kleine Bibliothek mit Assembler-routinen anlegen, die nach Belieben in den Code eingesetzt werden können. Weiterhin stellt "Laser Genius" sog. Direktiven zur Verfügung. Das sind Befehle, die Anweisungen an den Assembler geben und den Assemblierungsvorgang beeinflussen, jedoch nicht das spätere Programm. Sie steuern beispielsweise die Druckerausgabe oder das Ziel des Assemblierungsvorgangs.

Der Knüller des Assemblers ist aber die eingebaute Programmiersprache Phoenix, die leichter und schneller zu erlernen und zu handhaben ist als Assembler. Durch die Kombination beider Sprachen hat man ein nahezu unschlagbares Werkzeug für den Programmierereinsatz. Es ist beispielsweise denkbar, die Hauptteile eines Programms weiterhin in Assembler zu schreiben, schwierige Stellen dagegen in Phoenix (z.B. Funktionsdefinitionen, Repeat-Until-Schleifen oder ein IF...THEN..ELSE). Der Assembler akzeptiert selbst den größten Mischmasch dieser Sprachen.

Der Monitor muß noch nachgeladen werden, denn er ist ebenfalls ziemlich lang, weil er den Debugger und Disassembler beinhaltet. Der Disassembler ist bei "Laser Genius" ebenfalls etwas Besonderes; er erzeugt einen Sourcecode, der keine Fantasielabel enthält, sondern rekonstruiert genau die Label, die der Anwender vor dem Assemblieren gesetzt hat. Der Monitor bietet eine sehr übersichtliche Bildschirmaufteilung, wobei immer ein Speicherbereich, die Status-Flags und der zuletzt abgearbeitete Befehl sichtbar sind. Seine Stärke liegt im Debugging. Er stellt einen Einzelschrittmodus und acht verschiedene Geschwindigkeiten zur Abarbeitung des Programms zur Verfügung. Die 16 zur Verfügung stehenden Breakpoints bewirken unter anderem den Wechsel in verschiedene Geschwindigkeiten oder



Oben rechts die Registerinhalte, Register A bitweise aufgeschlüsselt. In der Mitte des Bildes der Speicherzeiger und der PC sowie der disassemblierte Speicherinhalt, darunter Hex- und ASCII-Dump.

den Abbruch, sobald eine Zeile zum n-ten Mal durchlaufen wurde.

Das beste Hilfsmittel bei der Fehlersuche ist der eingebaute Analyser. Er wird in einer Forth-ähnlichen Sprache programmiert und kommt selbst den hartnäckigsten Fehlern auf die Spur. Die Hauptanwendung liegt in der Definition von Breakpoints unter bestimmten Bedingungen. Das Programm kann praktisch unter jeder beliebigen Bedingung abgebrochen werden. Dies ist z.B. auch möglich, wenn es in einen bestimmten Speicherbereich schreibt, ein Stackwert

verändert wird oder Speicherstellen beliebige Werte annehmen. Sind diese Bedingungen erfüllt, können Verzweigungen ausgeführt, beliebige Stacks und Speicherstellen beeinflußt und Unter-routinen angesprungen werden.

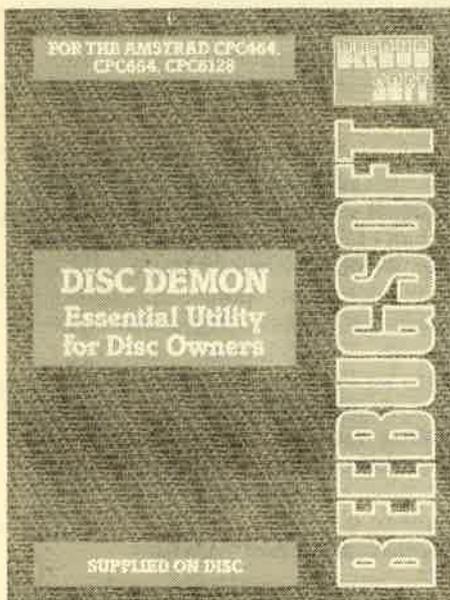
"Laser Genius" ist wirklich ein Genius unter den Assemblerprogrammen. Das etwas unübersichtlich und schlecht strukturierte englische Handbuch erleichtert die Arbeit mit dem Programm aber nicht gerade. Neulinge der Programmierkunst in Assembler seien also noch ein wenig vertröstet. Die Aussicht

auf ein schnelles Auffinden von Fehlern ist zwar verlockend, doch bis man sich in das Programm selbst eingearbeitet hat, dürfte die Lust an der Programmierung vergangen sein. Profis und solche, die sich dafür halten, sollten aber sofort in den nächsten Computer-Shop stürmen und "Laser Genius" verlangen.

System: CPC 464/664/6128
 Hersteller: Ocean Software
 Preis: Cass. 58.-, Disk 76.-
 Bezugsquelle: Peter West Records
 Thomas Tai

Wichtig für Floppy-Besitzer!

"Disc Demon" ist ein hervorragendes Utility für den Einsatz im Diskettenbetrieb



Hinter "Disc Demon" von Beebugsoft verbirgt sich kein fetziges Actionspiel, in dem der Spieler einen Dämonen jagen muß, sondern ein hervorragendes Werkzeug für alle ernsthaften Programmierer. Voraussetzung ist natürlich der Besitz eines oder mehrerer Diskettenlaufwerke. Zum Test lag mir die Diskettenversion des "Disc Demon" vor. Der englischen Anleitung konnte ich entnehmen, daß es auch eine ROM-Version gibt. Der Vorteil einer solchen Erweiterung liegt auf der Hand: Im CPC wird kein Speicherplatz verbraucht, das Utility ist immer arbeitsbereit, der Ladevorgang entfällt. Da man für den Einsatz eines ROM-Moduls aber noch eine Erweiterungskarte mit entsprechendem Steckplatz anschaffen muß, werden die

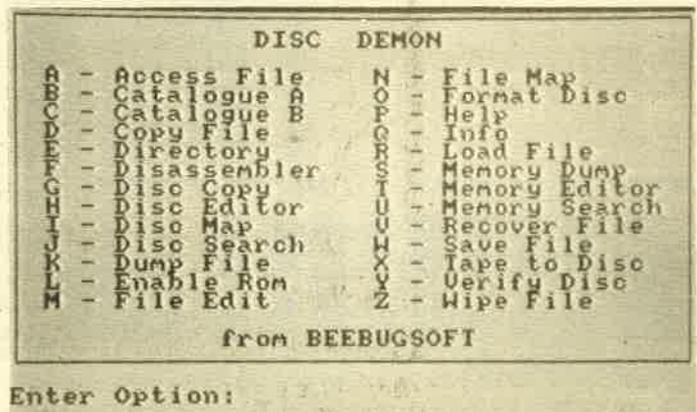
se Vorteile durch den höheren Preis fast wieder aufgehoben.

Nach Laden der Diskettenversion verbleiben dem Anwender beim CPC 464 rund 27 KByte an freiem Speicherplatz. "Disc Demon" meldet sich nach dem Laden mit dem Hauptmenü, was aber nicht bedeutet, daß wir es hier mit einem menügesteuerten Programm zu tun haben. Die Beebug-Programmierer haben bei der Realisierung ihrer Ideen einen optimalen Mittelweg gefunden, der mich voll überzeugt. Normalerweise werden die neuen Befehle mit einem vorangestellten Balken direkt vom Basic aus aufgerufen; es handelt sich also um RSX-Kommandos. Auf Wunsch kann man aber über solch einen RSX-Befehl auch das Hauptmenü als Befehlsüberblick abrufen oder über das Menü Kommandos wählen. Das Hauptmenü ist demnach ständig verfügbar, man muß aber nicht damit arbeiten. Bevor ich näher auf die einzelnen Befehle des Menüs eingehe, will ich noch eine Besonderheit

vorstellen. Die 10 Tasten des separaten Zahlenblocks wurden mit verschiedenen Funktionen belegt, die nicht im Menü enthalten sind, da sie dort nur begrenzt nutzbar wären, dem Programmierer aber dennoch während der Arbeit wertvolle Hilfestellung leisten. Nachstehend also die Belegung der Funktionstasten.

- f0 - setzt Screen-Modus 0.
- f1 - setzt Screen-Modus 1.
- f2 - setzt Screen-Modus 2.
- f3 - ruft Katalog von Laufwerk A auf.
- f4 - ruft eine Help-Seite auf. Hier werden die neuen Befehle kurz erläutert. Nach der Einarbeitungsphase kann man die Anleitung also abheften.
- f5 - ruft einen erweiterten Diskettenkatalog auf (z.B. mit Startadresse etc.).
- f6 - schaltet das ROM um.

Jede Menge Funktionen bietet das Hauptmenü von Disc Demon.



- f7 - schaltet den Drucker "On Line".
- f8 - schaltet den Drucker "Off Line".
- f9 - ruft das Hauptmenü auf.

Nach Druck auf eine Funktionstaste wird die entsprechende Funktion sofort ausgeführt. Natürlich kann man diese Befehle auch per Hand eingeben. Jetzt aber zu der Auflistung aller weiteren Befehle des "Disc Demon".

- DEMONOFF löscht "Disc Demon" aus dem Speicher.
- ACCESS sperrt oder öffnet Files auf Diskette. Kann als Zugriffssicherung eingesetzt werden.
- CATA ruft den Katalog von Laufwerk A auf.
- CATB ruft den Katalog von Laufwerk B auf.
- COPY kopiert einzelne oder mehrere Files von Diskette auf Diskette.
- DEDIT listet einen bestimmten Diskettensektor auf und ermöglicht das Editieren.
- DIRECT bearbeitet analog zu DEDIT den Diskettenkatalog.
- DIS ruft einen Disassembler auf.

- DISCCOPY kopiert den Disketteninhalt auf eine andere Diskette.
- DMAP stellt die Sektorbelegung der Diskette grafisch dar.
- DSEARCH sucht auf der Diskette einen bestimmten String.
- DUMP listet den kompletten Inhalt eines Diskettenfiles auf dem Bildschirm auf.
- FEDIT listet und untersucht ein File Sektor für Sektor.
- FMAP gibt den Diskettenstatus aus (z.B. Usernummer).
- FORM formatiert eine Diskette oder einen bestimmten Bereich.
- LOAD lädt jede Fileart an jede Adresse.
- MDUMP listet einen bestimmten Speicherbereich auf dem Drucker.
- MEDIT listet und untersucht den Speicherinhalt.
- MSEARCH sucht im Speicher einen bestimmten String.
- RECOVER holt ein gelöscht Programm zurück.
- REN dient dem Umbenennen eines Files.
- SAVE speichert einen Bereich mit

- unterschiedlicher Ladeadresse ab.
- TDISC kopiert Files von Cassette auf Diskette.
- VERIFY untersucht, ob eine Diskette korrekt formatiert wurde.
- WIPE löscht Files von der Diskette.

Damit wären die meisten neuen Funktionen aufgeführt. Die Anleitung erklärt alle Befehle ausführlich, in der Regel mit Angabe der Syntax und einem Beispiel. Anfänger können dieser Anleitung auch die Grundlagen der Arbeit eines Diskettensystems entnehmen.

Fazit: "Disc Demon" ist ein wirklich gutes Werkzeug für diskettenorientierte Programmierer. Schon nach kurzer Bearbeitungszeit will man darauf nicht mehr verzichten. Nach dem hervorragenden Grafikprogramm "Rembrandt" hat Beebugsoft mit "Disc Demon" wieder einmal bewiesen, daß diese Firma schon in Kürze zu den ganz großen der Branche gehören kann. Ich bin gespannt, was die Zukunft noch alles bringt.

System: CPC 464/664/6128
 Hersteller: Beebugsoft
 Bezugsquelle: PiZie-Data
 Rolf Knorre

Schneider CPC Software-Schnell-Versand

CLONE

CLONE ist ein neuartiges Diskettenkopierprogramm welches Sicherheitskopien von geschützter Diskettensoftware herstellt. Welcher Kopierschutz, ist dabei völlig nebensächlich. CLONE läuft auf jedem CPC und kopiert fast alle (99,9%) auf dem Markt befindliche Software.

Diskette 464/664/6128 **68,90**

DISC-WIZARD

DISC-WIZARD ist eine Rom-Erweiterung, die auf den Expansionsport gesteckt wird. Mit DISC-WIZARD können Sicherheitskopien einfach und bequem durch Knopfdruck erstellt werden. DISC-WIZARD läuft auf allen CPC's. Beim 6128 wird ein zusätzlicher Adapter benötigt.

DM 149,- Adap DM 49,50

Sind Sie mit Ihrem
LATEIN AM ENDE

Dann brauchen Sie den
MIRAGE IMAGER

Der Mirage - Imager wird hinten auf den CPC (464,664,6128) gesteckt und kopiert auf Knopfdruck alle Programme, die nicht nachladen müssen, wahlweise auf Kassette oder Diskette

- ACHTUNG -
 Der Mirage Imager darf nur für persönliche Sicherheitskopien benutzt werden !!

DM 219,-
 Adapter für CPC 6128
 DM 49,50



OAX kopiert die Programme, die mit dem Kopierschutz "SPEEDLOCK" geschützt sind wahlweise auf Kassette oder Diskette. SPEEDLOCK ist ein Kopierschutz, den viele Softwarehäuser bei Ihren Spielen verwenden. OAX liest diese Programme und erzeugt ungeschützte Duplikate. ACHTUNG - OAX darf nur zur Erzeugung von persönlichen Sicherheitskopien benutzt werden.

Kass 47,90 Disc 68,90

Händleranfragen erwünscht!

Neue Waldeck-Software | 2870 Delmenhorst | Ladenverkauf:
 Anschrift: Tulpenstraße 30 | Tel.: 04221/1 64 64 | täglich 15.00 - 18.00

Auf Postkarte füllen oder in Briefmarken-Steckbrief und abgeben bei Post
 Ja, schicken Sie mir umgehend folgende Artikel aus Ihrem Angebot:

Stück DM Bargeld anbei
 Stück DM per Nachnahme
 Stück DM per Überweisung
 Versandkosten DM 1,50 € (Vorkasse)
 Gesamtsumme DM

Name: _____
 Adresse: _____

Spielend Schreibmaschine lernen mit TASTSET?

Fingerübungen mit dem Schneider-Computer

Wenn man sich selbst beim Tippen mit zwei Fingern auf seinem CPC beobachtet, kommt oft der Wunsch auf, endlich einmal richtig mit allen zehn Fingern schreiben zu können (wozu hat man sie denn?). Genau hier setzt das Programm "TASTSET" an. Es versucht mit spielerischen Mitteln, den Benutzer zum sogenannten "Tastschreiben" (im Volksmund 10-Finger-Blindschreiben) zu führen. Ziel dieses Programms ist es, auf spielerische Weise die Sicherheit aller Griffe und damit die Grundlage für ein schnelles und fließendes Schreiben mit zehn Fingern zu erhalten.

Das Grundprinzip dieses in mehreren Stufen organisierten Programms ist das Betätigen von Tasten, die auf dem Bildschirm in einer "leeren" Tastaturgrafik angezeigt werden. Die Grafik selbst ist sehr einfach gehalten und lenkt daher nicht von der eigentlichen Aufgabe ab. Je nach Schwierigkeitsgrad ist die Menge der ausgewählten Tasten unterschiedlich gestaffelt. Die angezeigten Buchstaben in der Tastatur befinden sich entweder auf der richtigen Taste, dann ist diese Taste zu betätigen, oder aber auf einer falschen Taste. In diesem Fall muß dieses mit der Betätigung der Leertaste erkannt werden. Wurde ein Fehler gemacht, so wird zunächst die richtige Taste blinkend angezeigt, bevor es weitergeht.

Interessant ist dabei, daß dieses "Trainings-Spiel" die gemachten Fehler in einer "Fehlerwiederholungs-Warteschlange" registriert und auf diese Fehler im weiteren Verlauf besonders ein-

Zeit-Bremse hat die Aufgabe, dem Benutzer bei großer Fehlerhäufigkeit eine Besinnungspause zu geben. Mit der Leerung der Warteschleife normalisiert sich auch die Zeit-Bremse wieder. Da es beim Tastschreiben auch um die Schreibgeschwindigkeit geht, ermittelt "TASTSET" laufend die durchschnittliche Reaktionszeit und wiederholt gezielt Zeichen, die richtig, aber deutlich langsamer gegriffen wurden. Da das Ganze als Spiel ausgelegt ist, erfolgt zusätzlich eine Punktbewertung, an der man seinen Leistungslevel messen kann.

Nach dem Start von "TASTSET" erscheint ein Menü, in dem eine bestimmte Übung ausgewählt werden kann. Im wesentlichen sind dies die Standard-Spielstufen (unterschiedliche Zeichenmengen, die dem geübteren Benutzer vorbehalten sind), ein Übungsbereich, um gezielt bestimmte neue Tasten zu üben (neue Griffe), und eine Sonderzeichenfolge, in der der Benutzer ganz spezielle Folgen üben kann.

Für den Anfänger ist das Üben neuer Griffe von besonderer Wichtigkeit. Solange er im Bereich der "Grundstellung" (asdf jklö) übt, fängt das Spiel mit 2 Zeichen an; nach jeweils 36 aufeinanderfolgenden fehlerfreien Anschlägen kommt ein weiteres Zeichen hinzu. Werden die ersten 8 Zeichen dagegen schon beherrscht, so kann der Benutzer eine Teilmenge der verfügbaren Zeichen selbst auswählen und gezielt in 5 Phasen üben, deren Schwierigkeitsgrad allmählich zunimmt.

Die drei Standard-Spielstufen sind in alle Kleinbuchstaben (1), Groß- und Kleinbuchstaben (2) und alle Zeichen der Tastatur (3) eingeteilt. Hier findet eine Punktbewertung statt, die vom System registriert und gespeichert wird. Wurde eine neue Höchstpunktzahl erreicht, ertönt eine Melodie zur Belohnung. Diese Melodie, besser als Gepiepse, wirkt aber eher abschreckend als belohnend und sollte in dieser Form lieber aus dem Programm herausgenommen werden. Nach Beendigung der Übung (ein Abbruch ist übrigens jederzeit mit der DEL-Taste möglich), besteht noch die Möglichkeit, sie zu vertiefen, indem

auf die gemachten Fehler gesondert eingegangen wird.

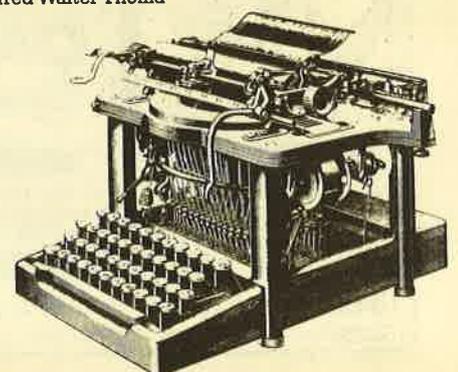
Das Programm "TASTSET" ist eine gute Spielidee, die Unterhaltung und Lernen kombiniert. Hier unterscheidet sich das Programm lobenswert von den meisten anderen Spielen. Trotz der Übungsvielfalt und der gut gelösten Fehlerbehandlung fehlen spezielle

◆ Verbesserungen sind in Arbeit

Übungsteile, die den Benutzer schrittweise zum gekonnten und selbstverständlichen Tastschreiben führen. Die spielerischen Momente stehen m.E. zu sehr im Vordergrund. So fehlen zusätzlich Text-Übungsteile, die das Schreiben von sinnvollen Wörtern und Sätzen gestatten. Diese sollten dann im Zeit- und Fehlverhalten analysiert werden. Laut Hersteller ist aber ein solches weiterführendes Text-Übungsprogramm bereits in Arbeit.

Mit "TASTSET" ist sicherlich der Einstieg in das Tastschreiben möglich, und das auf eine gelungene und spielerische Art und Weise. Zur konsequenten Weiterführung des Erlernten und Geübten sollten aber dann spezielle Übungstexte auf einem Textverarbeitungssystem folgen.

System: CPC 464/664/6128
Preis: Cass. 98.- DM, Disk. 108.- DM
Bezugsquelle:
Hanno E. Zimmerle
Am Eckbusch 39/110
5600 Wuppertal 1
Manfred Walter Thoma



◆ Mit Fehlerwiederholungs-Warteschlangen

geht. Falls in der Warteschlange Fehler gespeichert sind, so wird mit 50%iger Wahrscheinlichkeit das nächste Zeichen aus ihr entnommen. Dadurch ist eine gezielte Wiederholung möglich. Falls mehrere Fehler hintereinander eingetreten sind, tritt eine sogenannte "Zeit-Bremse" in Kraft, und es dauert länger, bis neue Zeichen angezeigt werden. Die

DAS SUPER BALLERSPIEL

FORCE



LIGHT

24,-

Bestellschein ausschneiden, ausfüllen und ab die Post zum Verlag Ritz-Eberle, Postfach 1640, 75118 Bretten.

Ja, ich will Lightforce haben! Schicken Sie mir sofort
..... Cassette(n) à DM 24,- ins Haus.

Name _____

Anschrift _____

Datum/Unterschrift _____

Ich habe einen Scheck über DM _____ beigelegt und muß daher keine zusätzlichen Portokosten bezahlen.

Ich wünsche den Versand per Nachnahme (zuz. DM 5,70 Porto und Versand)

Comal – der neue Favorit bei den Programmiersprachen

Comal enthält das Beste aus Basic und Pascal

Sie werden mir nicht glauben, aber Sie Stäten gut daran, es zu tun: Mit Comal 80 erleben Sie ein ganz neues Schneider-Gefühl! Das müssen Sie gesehen haben. Sie flicken Ihr Programm, weil Sie mit-tendrin etwas vergessen haben, z.B. die schließende Klammer bei H=VAL (A\$). Plötzlich verschwindet die nachfolgende Zeile unter dem Text: "Fehler Nr. 14:)" erwartet". Gleichzeitig legt der Interpreter seinen Finger auf die Wunde oder besser seinen Cursor auf die fehlerhafte Stelle, nämlich hinter das Dollar-Zeichen. Tun Sie ihm nun den Gefallen und geben die Klammer ein, so löst sich der Fehler in Luft auf, und wie von Zauberhand ist die ursprüngliche Zeile wieder da! Wirklich, beim ersten Erscheinen solcher wundersamer Dinge bekommt man große Augen. Dabei ist das noch das geringste unter seinen Kunststücken. Der Interpreter ist hinter den Fehlern her wie die Grünen hinter den Umweltsündern. Beide kann man, wenn man will, natürlich abwählen. Später mehr davon.

Da liegt es nun, das verheißungsvolle Päckchen für 69.- DM. Eine schlichte 3"-Diskette und ein maiengrünes Heftchen verbreiten Hoffnung. Also nichts wie "run". Dies ist der erste Erfahrungsbericht nach unserer Übersicht in Heft 7, Seite 40. Aber vor dem run "COMAL" sollten wir uns überlegen, ob es sich lohnt, in diese noch junge Sprache einzusteigen. Wie schon früher geschildert, besteht Comal aus den besten Teilen von Pascal und Basic, gut vermischt mit eigenen Zutaten. Feste Strukturen

Komfortabler Editor

verhindern den gefürchteten Basic-Salat, ohne einem jedoch durch übertriebene Pingeligkeit den Spaß am Programmieren zu verderben. Für Pascal wird empfohlen, seine Struktur durch Einrückungen der Zeilen beim Tippen zu verdeutlichen. Der Comal-Editor macht das von alleine. Und wenn es am Ende mal nicht aufgeht, so stimmt auch etwas nicht, aber Sie brauchen nicht nachzusehen, wo Sie sich vielleicht "verrrückt" haben könnten. Überlassen Sie die Suche dem

eingebauten Scanner (to scan = prüfen), der Ihnen sagt, an welcher Stelle Ihre Logik Sie verlassen hat. Daß Comal als Interpreter-Sprache langsamer sein muß als Pascal, bedarf keiner Erwähnung. Beim Rechnen ist es (z. Zt. noch) eine Winzigkeit langsamer als das Schneider-Basic, nicht jedoch bei der Textverarbeitung. Das liegt an der Verwendung der "Substring" genannten Technik, die auch die Befehle LEFT\$, MID\$ und RIGHT\$ nicht nur ersetzt, sondern sie übertrifft. Darüber mehr in einem gesonderten Beitrag.

Die Comal-Gruppe Deutschland schätzt, daß momentan ca. 50 000 Installationen laufen, die wegen der gleichartigen Benutzeroberflächen auch zwischen unterschiedlichen Rechnern weitgehend übertragbar sind. Das Landesinstitut IPN in Soest prüft im Augenblick im Auftrag des Landes Nordrhein-Westfalen in vielen Pilot-Projekten und Unterrichtseinheiten die Möglichkeiten des Schuleinsatzes. Eine Empfehlung für den Einsatz in den Sekundarstufen I und II liegt hier bereits vor, sofern nicht im Informatikunterricht Pascal zur Pflicht gemacht wurde. Ähnliches gilt für Niedersachsen (Hildesheim) und Schleswig-Holstein (Kiel). Bekannt sind auch Anwendungen in Hamburg, in Bremen, im Saarland und in Baden-Württemberg. An den Universitäten Paderborn, Würzburg, Heidelberg und der PH Freiburg ist Comal Favorit für das Informatik-Grundstudium. Sicherlich wird es auch an vielen mir unbekanntem Orten eingesetzt. Der Grund dafür liegt auf der Hand, denn Comal ist der universale Umsteigebahnhof mit Anschlüssen in alle Richtungen. Wer Comal beherrscht, hat keine Schwierigkeiten, sich auf andere Sprachen zu spezialisieren, was man von Basic nicht immer sagen kann. Also, wenn Sie mich fragen, ich stimme für Comal.

run "COMAL"

Kurze Zeit schnurrt die Diskette, und Comal meldet sich im 40-Zeichen-Modus mit seinem Starttext. Nun sind Sie automatisch im Editor. An diesen freundlichen Herren müssen Sie sich erst etwas gewöhnen, was aber so schnell geht, daß es Ihnen schon nach

zwei Stunden leid tut, wenn Sie wieder zum "gewöhnlichen" Basic-Editor zurückkehren. Die vier Cursor-Tasten sowie CLR und DEL gestatten Ihnen, völlig frei und ohne störendes Gepiepse auf dem Bildschirm herumzumarschieren und Zeichen vor und unter dem Cursor zu löschen. Die anderen Tastenwirkungen wollen wir wegen ihrer Nützlichkeit kurz unter die Lupe nehmen.

TAB

Tabulator. Wird mit ZONE eingestellt; Startwert 0.

COPY

Cursor nach links oben, Home-Funktion. Wenn Sie ein Window 0 (Meckerfenster für das System) gesetzt haben, wird dessen linke obere Ecke angesteuert.

Gute Mischung aus Basic + Pascal

SHFT CLR

Löscht alles ab Cursor bis Zeilenende. Wenn die Zeilennummer mitgelöscht wird, bleibt der Speicher unverändert!

SHFT COPY

Löscht den Bildschirm (evtl. nur das Meckerfenster).

SHFT CURSOR RECHTS

Entspricht dem INSERT anderer Rechner. Der Comal-Editor befindet sich stets im Modus "Überschreiben". Hiermit schaffen Sie Platz zum Einsetzen. Nehmen Sie ruhig reichlich davon, denn überflüssige Blancs wirft der Editor von selbst raus. Eine notwendige Fortsetzungszeile richtet er automatisch ein.

SHFT CURSOR LINKS

Wie CLR, d.h., das Zeichen unter dem Cursor wird gelöscht, der Zeilenrest nachgezogen.

SHFT CURSOR OBEN

Die aktuelle Zeile wird gelöscht, der Rest des Bildschirms hochgezogen.

SHFT CURSOR UNTEN

Die aktuelle Zeile wird um einen Platz nach unten geschoben und eine Leerzeile eingefügt. Sehr übersichtlich beim Einsetzen ganzer Zeilen. (Muß aber nicht sein! Sie können die neue Zeile auch auf eine beliebige freie Stelle pflanzen.)

CTRL CURSOR RECHTS

Cursor an rechtes Zeilenende (Rand).

CTRL CURSOR LINKS

Cursor an linken Zeilenanfang (Rand).

CTRL CURSOR OBEN

Cursor auf Zeile 1 bei gleicher Spaltenposition.

CTRL CURSOR UNTEN

Cursor auf Zeile 25 bei gleicher Spaltenposition.

CTRL COPY

Schaltet zwischen Mode 1 und Mode 2 hin und her.

Sie erkennen, wie vielseitig die Steuertasten sind. Insbesondere werden Sie anfangs die COPY-Taste "ins Herz schließen", weil Sie sie nicht mehr brauchen, aber trotzdem aus Gewohnheit draufdrücken und den Cursor heim-schicken. Zur Programmtext-Reparatur fahren Sie einfach in den Bildschirm und tippen nach der Arbeit auf die ENTER-Taste. Der Editor übernimmt dann die ganze Zeile einschließlich der evtl. geänderten Zeilennummer. Die alte Zeile bleibt in diesem Falle erhalten, was zum Kopieren von Zeilen ideal ist. Aber Achtung! Auch Restquatsch, der noch in der Zeile steht, wird übernommen und führt zur Fehlermeldung, weil der Editor jede Zeile unmittelbar nach ENTER analysiert und mit seinen Randbemerkungen nicht sparsam ist; immerhin existieren 72 Fehlermeldungen auf Deutsch (!). Dies gilt wie bei jedem echten Bildschirm-Editor auch dann, wenn Sie ein Direktkommando in den Programmtext einbauen, ist aber unschädlich. Mit der ESC-Taste können sie jede unerwünschte Aktion abstellen.

HELLO.SAV

Unser erstes Beispielprogramm beschäftigt sich mit der Umbelegung des Zehnerblocks. Von sich aus belegt Comal die Funktionstasten wie folgt: F0 SCAN+r, F1 LIST, F2 SAVE", F3 RE-NUM+r, F4 EDIT, F5 LOAD", F6 CAT+r, F7 AUTO+r, F8 FIND", F9 SIZE+r, F. RUN, ENTER bleibt

Das "+r" bedeutet sofortiges RUN. Stören Sie sich nicht an einigen noch unbekanntem Befehlen; sie sind ungeheuer praktisch. Diese Belegung erfolgt aber auf allen drei Ebenen, was mich störte. Zur Umbelegung verwende ich die normale Funktion des Schneider-Systems KM SET TRANSLATE (Nr. 13 des Firmware-Buches) mit der Einsprungsadresse &BB27, was unter Comal \$BB27 heißt. (Überhaupt können Sie alle Schneider-Funktionen einschließlich der Steuersequenzen (Kapitel 9, Seiten 1 u. 2 des Manuals) in Ihren Programmen benutzen.) Diese Funktion fordert das Laden des Akkus und des B-Registers vor ihrem Aufruf. Unter Basic bedeutet das,

daß Sie ein Maschinenprogramm schreiben müssen. Nicht so unter Comal. Die Befehle AF, BC, DE und HL versorgen Hilfszellen mit Ihren Zahlenwerten, die beim Erkennen des Aufrufs CALL adresse vor dem Ansprung der eigentlichen Routine in die gleichnamigen CPU-Register geladen werden. Um also in der obersten Ebene (ohne SHFT und CTRL) wieder Zahlen zu bekommen, genügt es, die Tastennummern und zugehörigen ASCII-Codes in DATA-Zeilen abzulegen und sie mit der WHILE-Schleife zu verarbeiten (Zeilen 60 bis 150 des Pro-

tiger Comal-Programmierer merken sollten. Der Befehl NEW ist nur von Hand zulässig. Was tun, wenn ein Programm sich "auf Französisch" verabschieden soll, ohne Spuren zu hinterlassen, also der "perfekte Mord"? Erzeugen Sie auf der Diskette eine Leerdatei mit beliebigen Namen, hier "nix.sav", indem Sie bei leerem Speicher den Befehl SAVE "nix" eingeben. Durch den CHAIN-Befehl in Zeile 240 wird "nix" geladen, aber der Speicher gelöscht. Die Zeichen // bedeuten übrigens das gleiche wie REM in Basic.

◆ Mit allen Schneider-Funktionen

gramms). Das END in Zeile 150 erzeugt beim Lesen das Signal EOD (end of data). Nun ist endlich Schluß mit dem Abzählen der Daten.

Im Anschluß daran wird eine Namensmeldung in Mode 2 ausgegeben. Die Variable a\$ habe ich eigentlich nur eingeführt, um die Wirksamkeit der Steuersequenzen (hier Wagenrücklauf/Zeilenvorschub) zu demonstrieren, und empty\$ bedeutet dasselbe wie "", also leer. Man kann beides benutzen, auch beim logischen Vergleich: 0 oder false, +1 oder true. Aber Vorsicht: Unter Basic ist true gleich -1! Die Zeilen 200/210 erzeugen dann ein CAT-Bild, wie wir es gewöhnt sind. Das Comal-CAT bringt eine unsortierte Kurzübersicht über den Disketteninhalt ohne Längenangaben. Dafür dürfen Sie Wildcards angeben, um Auszüge zu erhalten. In den Zeilen 220/230 wird der evtl. angeschlossene Drucker überprüft. BITAND deshalb, weil Comal zwischen der logischen und der bitweisen Verknüpfung unterscheidet.

Das Programm endet dann mit einem Trick, den Sie sich als (hoffentlich) künf-

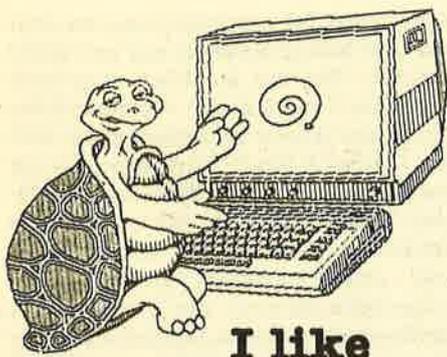
KILLER.SAV

Auch das zweite Programmbeispiel, das ich Ihnen zeigen möchte, ist ein Nutzprogramm, nämlich das Comal-Gegenstück zu dem Programm "KILLER.BAS", das in Heft 12/85 veröffentlicht wurde und ein bequemes Löschen und Umbenennen von Disk-Programmen möglich macht. Bevor wir in Einzelheiten gehen, sollten Sie sich das Listing im groben Überblick ansehen. Es besteht aus fünf völlig getrennten Teilen, von denen vier mit PROC anfangen und mit ENDPROC aufhören. Wenn Sie noch nicht mit Pascal in Berührung gekommen sind, muß ich Ihnen sagen, daß das die sagenhaften "Prozeduren" sind, von denen soviel geredet wird und die nichts anderes darstellen als gewöhnliche Unterprogramme, die nur mit ihrem Namen aufgerufen werden statt mit der Zeilennummer. Außerdem können sie sich selbst aufrufen (Rekursion) und eigenständige (lokale) Variablen haben. Was Sie wissen müssen, teilt man Ihnen mit. Die Prozeduren können in beliebiger Reihenfolge stehen; der Scanner notiert die Adressen.

Die ersten acht Zeilen (40-110) sind der Steuerkopf. Beginnen wir dort und betrachten dann die Prozeduren in der Reihenfolge, in der sie im Programm ste-



Comal 80 als ROM-Cartridge soll es schon Ende des Jahres für die CPCs geben. Das spart viele RAM-Speicherplätze und das Ganze wird auch etwas umfangreicher als die Disk-Version.



I like

COMAL

hen. In Zeile 40 vereinbaren wir, daß i\$ den Steuercode für die Invers-Darstellung enthalten soll. Dann wird mit PAGE (Seite) der Schirm gelöscht. Auch der PRINT-Befehl sieht genau wie in Basic aus, allerdings erzwingt hier das Semikolon jeweils ein Blanc. Die Zeilen 70 und 90 haben gemeinsam, daß sie nur aus einem Namen bestehen. In Zeile 80 trägt der Name noch einen "Index". Weiter unten sehen Sie, daß es sich um Prozedurnamen handeln muß, denn sie tauchen jeweils einmal hinter dem Wort PROC auf. Eine solche Namensnennung bezeichnen wir in Zukunft als "Aufruf". In Basic würde in Zeile 70 stehen: GOSUB 610; aber Comal kennt keine Zeilennummern, nicht mal beim GOTO! Comal kennt zwar diesen Befehl, aber seine Anwendung gilt außer in zwingenden Notfällen als Sünde. Die Zeilennummern dienen nur als bequeme Orientierungshilfe beim Programmieren und vor allem beim Entwanzen (debugging). Das "(i\$)" hinter dem Aufruf "main" bedeutet hier, daß sich die Prozedur "main" der Variablen i\$ bedienen möge. Ob sie es tut, steht nicht zur Debatte; nur übernehmen muß sie sie. Nach diesen drei Aufrufen folgt nur noch eine normale Meldung und die END-Anweisung, die aber auch fehlen könnte. Sie unterdrückt nur die in Basic obligatorische Ready-Meldung. Fehlt sie dagegen in Basic, so würde das Programm in das nächste Unterprogramm hineinlaufen

Keine Zeilennummern

und mit dem Schreckensschrei "return without gosub" in Panik enden. Prozeduren dagegen verhalten sich wie die meisten Schüler. Sie arbeiten nur, wenn sie dazu aufgerufen werden.

PROC cata CLOSED

Das CLOSED hinter dem Namen bedeutet "geschlossen". Diese Prozedur weiß nichts von ihrer Umwelt und kann deshalb auch keinen Umweltschaden anrichten, es sei denn, sie greift selbst wild in den Speicher und macht Unsinn. Das aber muß man dem Programmierer in die Schuhe schieben. Was "cata" tut, wissen wir schon vom Hello-Programm; es löscht den Schirm und rührt einen Schneider-CAT ein. Nach getaner Arbeit gibt jede Prozedur das Kommando wieder brav an den zurück, der sie aufgerufen hat.

PROC neuname CLOSED

Bezüglich des CLOSED gilt das vorher Gesagte, aber hier werden die zwei Variablen a\$ und n\$ eingerichtet. Sie sollen den alten und den neuen Namen eines Diskettenfiles aufnehmen, die dann mit RENAME ausgetauscht werden. Die Namen müssen den AMSDOS-Regeln entsprechen. Wenn es geklappt hat, wird die Prozedur wieder verlassen. Wenn nicht, geschieht das auch, aber das AMSDOS schiebt vorher eine Meldung ein, die Comal nicht abfangen kann, weil

Neue Schleifenstruktur

es nicht an der Reihe ist. Comal kann sie höchstens im Meckerkasten ablegen, wenn einer eingerichtet ist. Und was ist nun mit den Variablen? Die hält die Prozedur geheim. Der Speicherplatz wird wieder freigegeben. Eine geschlossene Prozedur ist gewissermaßen undurchlässig.

PROC main(i\$) CLOSED

Hier haben wir jetzt erstmalig eine geschlossene Prozedur vor uns, die nicht ganz "dumm" sein soll. Schon beim Aufruf hatten wir ja in Klammern einen Variablennamen anhängt. Für diese Variable (es können auch mehrere sein) stellt jetzt die Prozedur Speicherplatz bereit und füllt ihn mit dem, was im Aufruf steht. Der Prozedur-Kopf, also das PROC main(i\$) CLOSED, entscheidet erst jetzt darüber, ob mit dem Wert der Variablen oder mit ihrer Adresse gearbeitet werden soll. Hier brauchen wir nur den Wert, sonst hätte es main(REF i\$) heißen müssen. Da der neue Platz nur der Prozedur selbst bekannt ist, hätte ich statt i\$ auch nasowas\$ in den Kopf schreiben können. Dann hätte eben nasowas\$ den ASCII-Wert 24, aber nur innerhalb der Prozedur! Die beiden Plätze sind bei der Übergabe von Werten völlig unabhängig. In unserem Fall kann ich

mit dem i\$ innerhalb der Prozedur machen, was ich will; das i\$ außerhalb bleibt unverändert. Das möge momentan zur Parameter-Übergabe genug sein.

Unsere Haupt-Arbeitsprozedur "main" soll auf Wunsch auch eine neue Katalogausgabe machen und Files neu benennen; deshalb muß sie wissen, wo das geschieht. In Zeile 290 erfährt sie die Adressen durch IMPORT. Weiter werden drei Textmeldungen vereinbart, und in Zeile 330 wird festgelegt, daß h\$ auf keinen Fall länger sein darf als 12 Bytes (8 für den Filenamen, 3 für die Extension und 1 Byte für den Punkt). Zeile 340 kennen wir schon; die Prozedur "cata" wird aufgerufen und erzeugt das Inhaltsverzeichnis auf dem Bildschirm.

Zwischen den Zeilen 350 und 570 liegt eine in Basic unbekanntes Schleifenstruktur: REPEAT/UNTIL. Sie hat Ähnlichkeit mit WHILE/WEND, aber im Gegensatz dazu erfolgt die Abfrage erst am Schleifenende bei UNTIL, und sie wird verlassen, wenn die Abfrage wahr ist. Als Eselsbrücke kann man sich "Wiederhole.../es sei denn, daß ... wahr ist!" merken. Auf die vielen Zuweisungen brauchen wir nicht weiter einzugehen, denn wir wollen uns ja nur das herauspicken, was in Comal anders ist; höchstens sei noch erwähnt, daß Sie den Doppelpunkt bei "!=" nicht mitschreiben müssen. Solche Kleinigkeiten macht der Editor ohne Rückfrage. Interessant ist jedoch das INPUT in Zeile 370. Unter Comal kann INPUT viele Formen annehmen; hier die vielseitigste. Allgemein geschrieben sieht die Anweisung so aus:

INPUT AT zeile, spalte, länge: "Meldung": Variable,...

Hierdurch wird ein geschütztes Eingabefeld erzeugt, das genau bei "zeile/spalte" beginnt und "länge" Zeichen aufnehmen kann; weiter läuft der Cursor nicht. Innerhalb dieses Feldes können Sie bis zum ENTER schreiben, löschen, einsetzen usw., was Sie wollen. Raus kommen Sie nicht! Danach ist es nicht mehr definiert.

Die nächste, Basic-Programmierern unbekanntes Konstruktion ist CASE ... OF / WHEN / OTHERWISE / ENDCASE. Dieser "Fallvergleich" ist so vielseitig und nützlich, daß ich mich nur kurz fassen kann. Zwischen CASE und OF kann praktisch alles stehen, von der einfachen Variablen bis zum komplizierten Rechenausdruck. Auf jeden Fall wird das "Ergebnis" mit dem verglichen, was jeweils hinter den einzelnen WHENs steht. Trifft ein Fall zu, so macht der Rechner das, was in den nächsten Zeilen steht. Ist überhaupt nichts richtig, kommt nach OTHERWISE, was in Zeile

500 "Null" heißt und was ich auch gern täte, nämlich nichts.

Der IF-Befehl unterscheidet sich nicht sehr von seinem Basic-Partner. Wenn hinter THEN mehr als eine ausführbare Anweisung stehen soll, so muß er mit ENDF abgeschlossen werden. Allerdings hat man dann die Möglichkeit, mit ELIF = ELSE IF den IF-Befehl über beliebig viele Zeilen fortzuführen. Gleich noch den letzten Hinweis zu dieser Prozedur. ORD ist die Abkürzung von "Ordnungszahl", was in Basic ASC heißt.

PROC laufwerk CLOSED

Wenn Sie den Artikel bis hierher verfolgt haben, so sind Sie fast schon ein perfekter Comal-Programmierer. Aber Sie werden in Zeile 660 das WEND vermissen. Wenn wir davon absehen, daß es in Comal als ENDWHILE bekannt ist, so gilt das gleiche wie bei IF: Wenn nach dem obligatorischen DO nur ein Befehl folgt, kann es entfallen.

Nach diesen notwendigen Erklärungen unbekannter Befehle und Strukturen können Sie das Programm so leicht lesen, als hätten Sie es selbst geschrieben. Und das sogar ohne Kommentare, die man durch // gekennzeichnet reichlich über sein Programm streut. (Sie kosten keine Laufzeit! Der Scanner sorgt dafür, daß sie dem Programm unbekannt bleiben.) Die Lesevorschrift ist ganz einfach. Sie fangen mit dem Steuerprogramm an und folgen den Prozedur-Aufrufen. Wenn Sie daran denken, daß jeder Aufruf nichts anderes als der Einsprung in ein Unterprogramm ist, so landen Sie todsicher beim END in Zeile 110.

Dies und das und sonst noch was

Ich hoffe, daß ich mit den zwei kleinen Anwenderprogrammen und den allgemeinen Erörterungen zumindest Ihre Neugier geweckt habe. Ganz wollte ich Sie nicht mit den phantastischen Möglichkeiten überfallen, die in den Packages stecken, denn das wäre für den An-

natürlich. Das geht nicht, meinen Sie? Doch, das Package "Graphics" macht's möglich! Sogar mit wenigen Befehlen und sekundenschnell. Wenn Sie möchten, dann lassen Sie noch Ihren Namen drumherumschreiben, aufwärts, abwärts und über Kopf von links nach rechts. Hätten Sie lieber Turtle-Grafik? Auch dafür gibt es ein Package. Alles im Preis inbegriffen! Weitere Ergänzungen erlauben das Ein- und Ausschalten von Hintergrund-ROMs, das Einrichten von Funktionstasten und den Eigenbau von Zeichen.

Nach dem "Dies" kommt nun das "Das". Comal läuft zur Zeit nur unter AMSDOS oder von Cassette. (Von Cassette ist es aber nicht das Wahre. Wird wohl auch nicht ausgeliefert.) Auch Speichererweiterungen können zur Zeit noch nicht benutzt werden, sind aber sonst unschädlich. Damit haben Sie augenblicklich nur einen freien Speicher von 11512 Bytes zur Verfügung. Das reicht aber für die meisten Anwendungen im Heimcomputerbereich völlig aus. Comal ist eine noch junge Sprache, an deren Ausbau und Weiterentwicklung ständig gearbeitet wird. Das sollte Sie nicht davon abhalten, die Comal-Diskette jetzt schon zu kaufen, um erst einmal Erfahrungen im kleinen zu sammeln. Wenn sich vielleicht in etwa einem Jahr wesentliche Änderungen und Ergänzungen ergeben haben, wird Ihre Diskette für 15.- DM von der Comal-Gruppe Deutschland aufgemöbelt. Auch neue Packages sind im Preis mit drin. (Wenn Sie Maschinensprache können und großartige Ideen für neue, allgemein verwendbare Packages haben, so bieten Sie sie doch der Comal-Gruppe an. Wie man Maschinenprogramme zum Package macht, steht im Handbuch.)

Und "sonst noch was". Durch interne Straffung des Kernprogramms wird im Augenblick versucht, 8 KByte Speicherplatz einzusparen. Davon sollen ca. 2 KByte für die Arbeit mit relativen Dateien wieder geopfert werden, so daß dann effektiv über 16 KByte frei sind, was sich schon hören läßt. Weiterhin arbeiten vortex und die Comal-Gruppe seit jüngster Zeit zusammen, um auch den Leuten zu helfen, die mit VDOS 2.0 oder VDOS 2.1 arbeiten. Ob damit auch das Schaffen mit den Erweiterungen SP 64-512 möglich wird, weiß ich noch nicht. Wir halten Sie darüber aber auf dem laufenden. Wenn Sie etwas mehr Geld anlegen wollen, so können Sie spätestens bis zum Weihnachtsfest das Comal auch als ROM haben. Diese Version ist ungleich mächtiger und enthält auch noch weitere Befehle. Auch Packages sitzen dann im ROM, brauchen nicht mehr geladen werden und verlangen keinen Speicherplatz. Etwa 35 KByte bleiben

frei für Ihre Programme. (Man munkelt sogar etwas von einer RAM-Floppy!) Der Preis steht noch nicht fest, er wird aber sicher niedriger sein als beim IBM-Comal (auch für alle MS-DOS-Rechner), das mit 950.- DM (!) zu Buche schlägt. Auch Versionen für andere MS-DOS-Systeme und für CP/M sind in Vorbereitung.

Zum Schluß noch ein wichtiger Hinweis. Die Bezeichnung Comal 80 ist für eine Bestellung zu allgemein. Für die Schneider-Rechner 464, 664 und 6128 müssen Sie die Version Comal 1.83 verlangen. Für alle kaufmännischen Dinge, Bestellungen, Preisfragen usw. ist die Comal-Gruppe Deutschland, Herr D.

Wo bekommt man Comal?

Belz, 2270 Utersum auf Föhr, zuständig (Tel. 0 46 83 / 5 00). Wenn Sie technische Probleme haben, können Sie sich - natürlich unter Beifügung von ausreichendem Rückporto - direkt an Herrn Siegfried Bauer, Meiereiweg 7, 2381 Börm, wenden.

Ich hoffe, auch Sie sprechen bald "comalsch". Wenn Sie an weiteren Berichten, Tricks und Programmen in oder über Comal Interesse haben, so lassen Sie es bitte unsere Redaktion wissen; Postkarte genügt.

Prof. Walter Tosberg

BASIC BITES THE DUST



Graphics macht's möglich

fang zu viel gewesen. Stellen Sie sich einmal vor, Sie wollten das Grinsgesicht (Zeichen 224) bis auf den vollen Bildschirm vergrößern; die linke Augenhöhle wird zur runden Zielscheibe, rechts erscheint eine elliptische Katzenpupille, und im Munde flitscht das Gesicht die Zeichen 212 bis 215 als Zähne, in bunt

KILLER.SAV

```

0010 // KILLER.SAV
0020
0030
0040 i$:=CHR$(24)
0050 PAGE
0060 PRINT i$;"KILLER (COMAL)";i$
0070 laufwerk
0080 main(i$)
0090 cata
0100 PRINT "KILLER-Ende"
0110 END
0120
0130
0140 PROC cata CLOSED
0150 PAGE
0160 DE 42625
0170 CALL $BC9B
0180 ENDPROC cata
0190
0200
0210 PROC neuname CLOSED
0220 INPUT "Alter Name: ": a$
0230 INPUT "Neuer Name: ": n$
0240 RENAME a$,n$
0250 ENDPROC neuname
0260
0270
0280 PROC main(i$) CLOSED
0290 IMPORT cata,neuname
0300 b$:"Name.ext / R)ename / C)at / <
ENTER> : "
0310 c$:=i$+" Nur einzeln loeschen! "+i
$
0320 d$:=i$+" Extension vergessen ! "+i
$
0330 DIM h$ OF 12
0340 cata
0350 REPEAT
0360 PRINT b$;
0370 INPUT AT CURLIN,LEN(b$)+1,12: "":
h$
0380 IF h$="" THEN h$:"!!"
0390 h$:=UPPER$(h$)
0400 CASE h$ OF
0410 WHEN "R."
0420 neuname
0430 h$:"#.#"
0440 WHEN "C"
0450 cata
0460 h$:"#.#"
0470 WHEN ".*.*","*.BAS","*.SAV","*.LST
","*.PCK"
0480 h$:"#.#"
0490 PRINT AT CURLIN-1,50: c$
0500 OTHERWISE NULL
0510 ENDCASE
0520 IF "." IN h$=0 THEN
0530 PRINT AT CURLIN-1,50: d$
0540 h$:" "
0550 ENDIF

```

```

0560 IF ORD(h$)>41 THEN DELETE h$
0570 UNTIL h$="!!"
0580 ENDPROC main
0590
0600
0610 PROC laufwerk CLOSED
0620 h$:=EMPTY$
0630 PRINT AT 3,1: "Laufwerk: A / B / <
ENTER>: ";
0640 WHILE h$="" DO h$:=LOWER$(KEY$)
0650 CASE h$ OF
0660 WHEN "a","b"
0670 UNIT h$+": "
0680 OTHERWISE
0690 NULL
0700 ENDCASE
0710 ENDPROC laufwerk
0720
0730 // Prof. Walter Tosberg 15.08.86
0740 // Letzte Aenderung: 21.08.86

```

HELLO.SAV

```

0010 // HELLO.SAV
0020 MODE 2
0030 //Normzustand der Funktionstasten h
erstellen
0040 // d.h. ohne Shift--> Ziffern im Ze
hnerblock
0050
0060 WHILE NOT EOD DO
0070 READ a
0080 READ b
0090 AF a*256
0100 BC b*256
0110 CALL $BB27
0120 ENDWHILE
0130 DATA $0F,$30,$0D,$31,$0E,$32,$05,$3
3,$14,$34
0140 DATA $0C,$35,$04,$36,$0A,$37,$0B,$3
8,$03,$39
0150 DATA $07,$2E,END
0160
0170 a$:=CHR$(13)+CHR$(10); b$:=EMPTY$
0180 PRINT "COMAL 1.83";a$
0200 DE 42625
0210 CALL $BC9B
0220 IF INP($F500) BITAND $40<>0 THEN b$
:="un"
0230 PRINT "Drucker "+b$+"klar."+a$+"HEL
LO.SAV-Ende."
0240 CHAIN "nix.sav"
0250
0260 // NIX = Mit SAVE abgespeicherte Le
er-Datei
0270 // Prof. Walter Tosberg 15.08.86
0280 // Letzte Aenderung: 25.08.86

```

Heiße Rhythmen

Das Digital Drum System im Test

Wenn Ihr Nachbar einen CPC besitzt und Sie eines Tages meinen, er hätte seinen Computer gegen ein Schlagzeug eingetauscht, dann hat er nicht erst die Empfehlung dieses Testberichtes abgewartet, sondern "Amdrum" bereits gekauft. Doch wir wollen nicht vorgehen. Bevor wir mit dem Test beginnen, nur noch eines: Sollte dieser Artikel manchmal zu euphorisch wirken, nehmen Sie uns das bitte nicht übel, denn man kann hier wirklich ins Schwärmen kommen.

Wer als stolzer "Amdrum"-Besitzer die Packung geöffnet hat, sieht vor sich ein kleines Modul mit einem langen Kabel, eine sog. Anleitung im Miniformat und eine Cassette mit der Treibersoftware. Stecken wir nun das Modul in den Expansionsport, schließen das Kabel am Verstärker an und kommen zur Lektüre der Anleitung. Hier merken wir dann, daß etwas ganz Wesentliches fehlt, nämlich die Lupe, um die Anleitung lesen zu können. Das Heftchen ist so klein geraten, daß es gute Chancen auf den ersten Preis im Wettbewerb um die kleinste Anleitung der Welt hätte. Hat man aber das Kunststück vollbracht, die Schrift zu entziffern, kann man damit beginnen, die Software zu laden. Hier liegt nun ein weiterer, allerdings auch der letzte Nachteil von "Amdrum". Bevor man wirklich loslegen kann, ist jedesmal das Laden des Programms zur Erstellung der Rhythmen, der digitalisierten Sounds und eventuell fertiger Rhyth-

men notwendig. Alles in allem muß man mit 10-15 Minuten rechnen, bevor man die Erweiterungen in Betrieb nehmen kann.

Das Faszinierende an "Amdrum" ist die Tatsache, daß es mit digitalisierten Sounds arbeitet. Was einem da aus den Lautsprechern entgegenläßt, ist von einem wirklichen Schlagzeug kaum noch zu unterscheiden. Mitgeliefert werden 8 Drumsounds; allerdings ist noch mit dem Erscheinen diverser Sounderweiterungen (weitere Trommeln sowie futuristische Rhythmustöne) auf einer einfachen Cassette zu rechnen.

Von den Drumsounds zu unterscheiden sind die Songs, von denen "Amdrum" 16 (!) gleichzeitig im Speicher verwalten kann. Die Voraussetzungen für eine Live-Aufführung sind damit erfüllt. Man macht "Amdrum" startklar und lädt die 16 Songs. Für das Abspielen müssen Sie dann nur noch zwei Tasten drücken. Werden mehr als 16 Songs vorgespielt, ist nach einiger Zeit eine Pause notwendig, was aber nicht das größte Problem darstellen dürfte.

Die Programmierung der Rhythmen ist nun eine denkbar einfache Sache. Man gibt sie entweder über den Editor ein oder klopft sie mit dem Finger auf der Tastatur. Im Editor muß man nur die Nummer der gewünschten Trommel in eine der drei möglichen Stimmen schreiben. Sie können sich nun Ihr Werk jederzeit anhören und verbessern, Wiederholungen setzen und auch Teile wieder lö-



schen. Nun ist es aber auch denkbar, nur den Grundrhythmus über den Editor einzugeben und dann in den "Tipp-Modus" zu gehen, indem Sie den Grundrhythmus abspielen und mit der CTRL-Taste einen zusätzlichen Sound tippen.

Die Möglichkeiten von "Amdrum" entsprechen denen einer programmierbaren Rhythmusmaschine; nur ist der Anschaffungspreis bedeutend niedriger, und der Computer läßt sich zusätzlich noch auf anderen Gebieten einsetzen. Hobbybands und Soundtüftler sind mit "Amdrum" hervorragend bedient. Preislich und leistungsmäßig bildet die Erweiterung eine echte Alternative zu herkömmlichen Rhythmusmaschinen. Man kann sich nur noch dem gutgemeinten Ratschlag des Handbuchs anschließen: Haben Sie Erbarmen mit Ihren Nachbarn (wenigstens nach zwölf Uhr nachts)!

System: CPC 464/664/6128
 Hersteller: Cheetah Marketing
 Preis: ca. 150.-DM
 Bezugsquelle: Profisoft + Peter West Records
 Thomas Tai

VORTEX

F1-S	5.25" Einzelstation mit Controller	919,-
F1-D	5.25" Doppelstation mit Controller	1398,-
F1-X	5.25" Systemzweilaufwerk, 1 MB Kapazität	709,-
F1-XRS	F1-X Laufwerk mit RS 232 Schnittstelle	848,-

M1-S	3.5" Einzelstation mit Controller	909,-
M1-D	3.5" Doppelstation mit Controller	1419,-
M1-X	3.5" Systemzweilaufwerk, 1 MB Kapazität	729,-

NEU - NEU - PREISSENKUNG - PREISSENKUNG - NEU - NEU

SP-256	Speicherkarte mit 256 KB RAM (bisher: 450,-)	279,-
SP-512	Speicherkarte mit 512 KB RAM (bisher: 535,-)	379,-

WD-20	5.25" Winchesterstation mit Floppy	3198,-
-------	------------------------------------	--------

Alle weiteren vortex-Produkte zu den bekannten Superpreisen!!
 Bitte geben Sie bei der Bestellung von vortex-Produkten unbedingt den Computertyp an!

DRUCKER - DRUCKER - DRUCKER - DRUCKER - DRUCKER

Panasonic KX-P 1080	100 cps EDV.20 cps NLQ	669,-
Panasonic KX-P 1092	180 cps EDV.33 cps NLQ + 1000 Bl. Endlospapier	979,-
Panasonic KX-P 1592	wie KX-P 1092, jedoch 15" breit	1399,-
Seikosha MP 1300	AI 300 cps, Centronics	1699,-

Druckerkabel zum Anschluß der genannten Drucker an die CPC's 75,-

TG-Soft

FD-2 3"	Zweilaufwerk (1 MB) für Joyce	589,-
	256 KB RAM Erweiterung für Joyce	129,-
	kompl. Aufrüstsatz für Joyce auf Joyce plus	709,-
	Bildschirmfilter	Joyce 89,- GT 64/65 59,- CTM 640/644 79,-

PC Emulator von Kersten+Partner ab 1250,-

10 Stück 3"	CF2 Disketten	99,-
	3" CF2-DD Disketten (1 MB) für Joyce	17,90
10 Stück 3.5"	Disketten 135 Ipi von Maxell	66,90
10 Stück 5.25"	DS-DD Disketten von Verbatim	49,90
	Joystick 'Speedking'	44,90
	Dk' Tronics Lightpen	79,-
	CPC MousePack	185,-
	Monitordreifuß stufenlos einstellbar	34,90
	SFT-401 Traktoraufsatz für NLQ-401	65,-

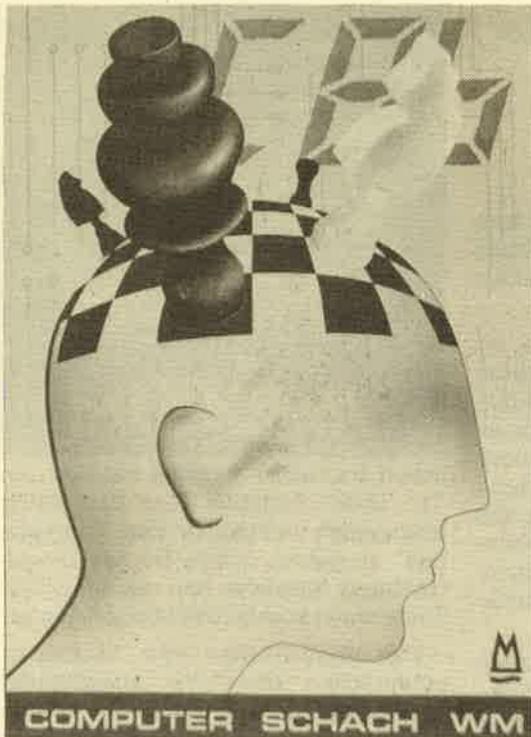
SOFTWARE

Verbrainer ENGLISCH I	CPC	39,-/49,-
(Testbericht in "CPC Int." 4/86)	Joyce	59,-
Vokabeltrainer für Englisch und Latein	CPC	45,-/55,-
	Joyce	59,-
Faktum 30 umfangreiches Bürosoftwarepaket	Joyce	298,-
Faktum 10 Fakturierungsprogramm	CPC	139,-/149,-
	Joyce	149,-
vortex PARA Diskettenmanager	CPC	57,-
Platinenkil Die(!) Leiterplattenentflechtung I. d.	CPC	144,-/149,-
DR-DRAW (Digital Research) Ergänzung zu GSX	CPC 6128	179,-
	Joyce	179,-
DR GRAPH (Digital Research)	CPC 6128	179,-
	Joyce	179,-
3D Clock Chess Schachprogramm	Joyce	64,90
Wordstar I dBase II / Multiplan	(CPC/Joyce)	je 179,-
Paket Wordstar + dBase II + Multiplan		499,-
Turbo Pascal	(CPC/Joyce)	215,-
CBasic Compiler	(CPC/Joyce)	169,-
Pascal.MT+	(CPC/Joyce)	169,-

Wir haben die neuesten Spiele zu Superpreisen!

Der Versand erfolgt per Nachnahme oder Vorauskasse (2 % Skonto).
 Alle Preise zzgl. Versandkostenanteil.

TG-Soft, Offersdorf 5, 8491 Rimbach, Tel. 09941/3765



Geschichte des Computerschach – Bericht über die Weltmeisterschaft in Köln

In unserer neuen kleinen Reihe wollen wir auch das Prinzip der Schachprogramme näher erläutern

Auf diesen Seiten beginnen wir mit einer kleinen Reihe, die Ihnen das Thema "Computerschach" näherbringen soll. Obwohl inzwischen Millionen der kleinen spielstarken Schachcomputer – wie z. B. Mephisto 3, Super Constellation oder Fidelity Elite, um nur einige Namen zu nennen – die deutschen Wohn- und Kinderzimmer überflutet haben und fast jeder Homecomputer-Besitzer über mindestens ein Schachprogramm für seinen Rechner verfügt, wissen doch erstaunlich wenige Schach- und/oder Computerliebhaber, was im Inneren ihrer elektronischen Spielpartner vorgeht. Deswegen möchten wir in diesem Heft und in den nächsten Ausgaben des CPC-Magazins jeweils auf einigen Seiten versuchen, Ihnen zu erklären, wie es möglich ist, daß ein "toter Gegenstand" wie eine Rechenanlage eine Aufgabe bewältigen kann, die zweifelsohne Intelligenz erfordert. Dies mittlerweile so gut, daß viele Menschen im Zweikampf mit der Maschine den kürzeren ziehen. Doch bevor wir uns damit beschäftigen, wie ein Schachcomputer oder ein universell einsetzbarer Homecomputer wie Ihr CPC-Rechner programmiert werden muß, damit er das Schachspiel beherrscht, wollen wir ein wenig in der Geschichte des Computerschach blättern und anschließend aus aktuellem Anlaß – in diesem Jahr fand die 5. Computerschach-Weltmeisterschaft in Deutschland statt – einen Blick auf die Leistungsstärke der heutigen Spitzenprogramme werfen. Abgerundet wird dies durch eine kleine Übersicht, die dem interessierten Leser einige Hinweise auf lesenswerte Zeitschriften und

Bücher gibt, mit deren Hilfe er sich tiefer in diese Materie einarbeiten kann.

Wenn wir nach den Wurzeln des Computerschach forschen wollen, müssen wir uns bis in die Mitte dieses Jahrhunderts zurückbewegen. Über 35 Jahre ist es nämlich schon her, daß der berühmte amerikanische Mathematiker Claude E. Shannon und der nicht minder bekannte englische Mathematiker und Informationstheoretiker Alan Turing unabhängig voneinander in Artikeln, die in Fachzeitschriften erschienen, Überlegungen anstellten, wie man eine (Rechen-) Maschine dazu bringen könne, Schach zu spielen. Schon vor diesen Herren spekulierten große Wissenschaftler wie Charles Babbage, der im 19. Jahrhundert, seiner Zeit weit voraus, an einer mechanischen Rechenanlage bastelte, und auch Norbert Wiener, der 1948 ein Buch über Kybernetik veröffentlichte, über die Spielstärke, die eine Maschine im

me gestaltete sich entsprechend schwierig. Daher verwundert es auch nicht, daß die Vorstellungen von Shannon und Turing niemals auf einem Rechner realisiert wurden. Immerhin ist aus jenen Tagen eine Partie bekannt, die Turing gegen einen menschlichen Gegner spielte (und verlor), indem er die Berechnungen seines Algorithmus mühsam per Hand (!) nachvollzog. Nichtsdestotrotz hat insbesondere der Artikel Shannons später große Bedeutung für die Entwicklung von Schachprogrammen erlangt, finden sich doch seine Ideen in den Grundzügen selbst der heutigen Schachprogramme wieder.

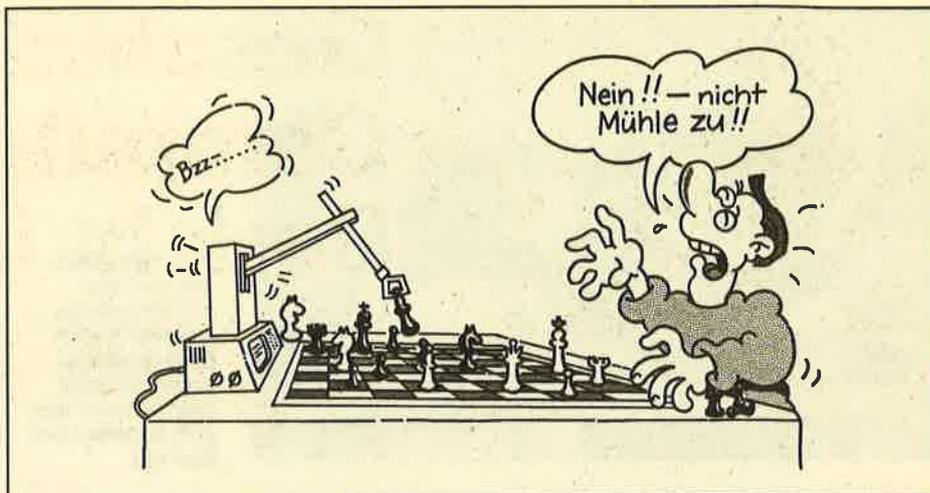
Bis zum Jahre 1958 dauerte es, ehe das erste voll funktionsfähige Schachprogramm der Öffentlichkeit in einem Bericht vorgestellt wurde, der neben einer Beschreibung der verwendeten Algorithmen auch die Züge einer gespielten Partie enthielt. Dies war der Erfolg der Arbeit von Alex Bernstein und anderen amerikanischen Wissenschaftlern, die hierfür einen mit Vakuumröhren ausgestatteten Rechner benutzten. Diese Maschine konnte ca. 42000 Befehle pro Sekunde ausführen. Heutige Homecomputer sind um den Faktor 20 und mehr schneller. Menschen, die gegen das Programm antraten, wurden auf eine harte Geduldsprobe gestellt: Ungefähr acht Minuten benötigte der Rechner, um sich jeweils für einen Zug zu entscheiden. Die Spielstärke dieses "Pioniers des Computerschach" war naturgemäß noch recht dürftig. Die oben genannte Partie gegen einen menschlichen Gegner mußte nach 22 Zügen in hoffnungsloser Stellung aufgegeben werden.

◆ Geschichtliche Grundlagen

Schachspiel erreichen könne. Doch dies waren alles nur vage Gedanken; die ersten konkreten Vorschläge zur Programmierung stammen von Shannon und Turing. Die technischen Möglichkeiten waren damals allerdings noch sehr begrenzt: Die erste programmgesteuerte Rechenanlage war erst ca. 10 Jahre zuvor von Konrad Zuse entwickelt worden. Die Programmierung wurde ausschließlich direkt in Binärcode vorgenommen. Das Erstellen umfangreicher Program-

Nur einige Jahre später begannen die Programmierer, ihre "Zöglinge" im direkten Vergleich zu testen. Das erste bekannte Aufeinandertreffen dieser Art war der Zweikampf zwischen den Programmen von Alan Kotok und John McCarthy (Universität von Stanford, USA) und George M. Adelson-Belsky, Vladimir L. Arlazarov u.a. (Institut für Theoretische und Experimentelle Physik, Moskau). Am 22. November 1966 begann dieses Turnier, in dem beide Programme vier Partien gegeneinander spielten. Beide Seiten benutzten ihre institutseigenen Rechner und übermittelten sich die Züge per Telegraph. Da das Ganze ohne Bedenkzeitlimit ablief, wurde die erste Partie erst am 10. März 1967 mit einem Sieg für das sowjetische Programm abgeschlossen. Der Endstand lautete schließlich 3:1 für Adelson-Belsky & Co. (Zwei Partien wurden nicht zu Ende gespielt und unentschieden gewertet.) Nun könnte man meinen, es gäbe nichts Langweiligeres, als zwei Maschinen beim Schachspiel zuzusehen, insbesondere dann, wenn sie sich, zumindest gemessen am Großmeisterniveau der Menschen, gegenseitig an fehlerhaften Zügen überbieten. Dies ist in der Tat ein Standpunkt, den viele Leute vertreten. Näher besehen sind solche Partien jedoch sehr nützlich, um die Fortschritte der Programme bzgl. ihrer Spielstärke zu überprüfen und die Ergebnisse dieser Analysen als Verbesserungen wieder ins Programm einfließen zu lassen. Auch heute ist es gerade bei Programmen auf Mikrorechnern gang und gäbe, Hunderte von Testpartien zu absolvieren, bevor sie ins harte Turniergefecht geschickt oder als kommerzielles Produkt auf den Markt gebracht werden.

Doch blicken wir noch einmal in die 60er Jahre zurück. Ungefähr zur gleichen Zeit, als das Programmduell USA – UdSSR stattfand, gelang es Richard Greenblatt, einem Studenten am Massachusetts Institute of Technology in Boston (USA), als erstem, mit seinem Programm, das er "Mac Hack Six" nannte, Erfolge im Kampf gegen menschliche Gegner zu erzielen. Im Frühjahr ließ er "Mac Hack Six" in einigen Turnieren antreten. Dies bedeutete, daß das Programm nach den turnierüblichen Regeln durchschnittlich nicht mehr als drei Minuten Bedenkzeit pro Zug in Anspruch nehmen durfte. In diesem Rahmen kam es auch zur ersten Partie, die ein Schachprogramm unter Wettkampfbedingungen gegen einen Menschen nicht verlor. "Mac Hack Six" brachte es insgesamt auf einige hundert Partien und konnte dabei eine Spielstärke nachweisen, die der fortgeschrittener Amateurspieler entsprach. Als Anerkennung dieser Ver-



dienste um das Computerschach trug der US Schachverband dem Greenblatt-Programm später sogar die Ehrenmitgliedschaft an!

Die ACM (Association for Computing Machinery) ist die älteste professionelle Vereinigung von Computerspezialisten in den USA. Sie hält in jährlichem Abstand Konferenzen ab, auf denen neue Forschungsergebnisse vorgestellt und diskutiert werden. 1970 wurde im Rahmenprogramm einer ACM-Veranstaltung in New York das erste Computerschachturnier ausgetragen. Einer der Initiatoren dieses Ereignisses war Tony Marsland, der Ende der 60er Jahre mit

◆ Aus den 60er Jahren

seiner Arbeit auf dem Gebiet des Computerschach begann und auch heute noch, inzwischen als Professor an der Universität von Alberta (Kanada), mit seinem Programm "Awit" aktiv ist. Mit seiner Unterstützung konnten fünf weitere Schachprogrammierer zur Teilnahme bewegt werden, so daß sich insgesamt sechs Bewerber um den Titel eines "1. US-Meisters im Computerschach" stritten. Dabei tauchten mit David Slate, Larry Atkin und Hans Berliner erstmals Namen auf, die in den folgenden Jahren die Szene entscheidend prägten und z.T. auch heute noch zu den führenden Leuten zu zählen sind.

Dieses erste Turnier fand bei den Beteiligten und bei den Zuschauern so großen Anklang, daß beschlossen wurde, diese Idee fortzuführen. So wurden die alljährlichen ACM-Turniere ein fester Bestandteil des Wettkampfkalenders. Auch heute noch kommen hier viele Programmierer zusammen. Nur der Name der Veranstaltung hat sich geändert. Inzwischen geht es um die "Offenen Nordamerikanischen Meisterschaften", an denen regelmäßig auch europäische

Programme teilnehmen. Als Austragungsmodus wurde das sogenannte "Schweizer System" gewählt, bei dem in jeder Runde möglichst Teilnehmer gegeneinander spielen, die in den vorangegangenen Partien die (annähernd) gleiche Punktzahl gewonnen haben. Auf diese Weise ist gewährleistet, daß einerseits keiner nach einer Niederlage vorzeitig ausscheiden muß (wie etwa beim K.o.-System) und daß andererseits das stärkste Programm ermittelt wird, ohne sämtliche Partien eines Rundensystems "Jeder gegen jeden" austragen zu müssen. Im Auftaktturnier 1970 belegte "Chess 3.0", das Programm von Larry Atkin, Keith Gornen und David Slate, Studenten an der Northwestern University (USA), unangefochten mit drei Punkten den ersten Platz. Dies war der Beginn einer Siegesserie, die "Chess 3.0" und seinen Nachfolgeversionen in den folgenden Jahren eine dominierende Rolle im Computerschach zuwies. In den ersten zehn US-Meisterschaften belegte "Chess" achtmal den ersten und zweimal den zweiten Platz! Dazu kommen ein zweiter Platz bei der Weltmeisterschaft in Stockholm (1974) und die Krone des Computerschach, der WM-Titel 1977 in Toronto.

Damit sind wir schon bei einer neuen Epoche angelangt. Seit 1974 werden im dreijährigen Rhythmus Weltmeisterschaftsturniere ausgetragen. Die Schachprogrammierung hat seit Beginn der 70er Jahre auch außerhalb der USA immer mehr Wissenschaftler interessiert. An erster Stelle müssen dabei die Bemühungen um "Kaisa" genannt werden. Hinter diesem Namen verbirgt sich die Weiterentwicklung des Moskauer Programms, das seinerzeit das Duell gegen die Entwicklung von Kotok und McCarthy gewonnen hatte. Von dem damaligen Team wirkte noch Arlazarov mit, als 1971 die Arbeit wieder aufgenommen wurde. Auch England mauserte sich zur Hochburg des Computerschach; mit drei Programmen war die-



Prof. Hans Berliner, Ex-Fernschachweltmeister und "Vater" der Schachmaschine HITECH (rechts neben ihm sein Assistent Carl Ebeling)



Harry Nelson (mit Brille) vom Team des Weltmeisterprogramms "Cray Blitz" (neben ihm seine Frau)



Richard Lang (links), der Programmierer des amtierenden Weltmeisters der Mephisto-Mikros. Neben ihm, ebenfalls aus dem Mephisto-Team, Ossi Weiner.



Großer Publikumszuspruch in der Kölner Messehalle.

ses Land in der WM 74 nach den USA am stärksten vertreten. Von 13 Teilnehmern konnte nur "Kaissa" in diesem Turnier alle vier Partien gewinnen und wurde erster Computerschachweltmeister der Geschichte. Das "Schweizer System" und eine Niederlage von "Chess 4.0" in der zweiten Runde führten allerdings dazu, daß "Kaissa" nicht gegen seinen härtesten Konkurrenten antreten mußte. Dieser direkte Vergleich der zwei vermutlich stärksten Programme dieser Zeit wurde jedoch in einem Demonstrationsspiel im Anschluß an das Turnier nachgeholt und endete unentschieden. Drei Jahre später endete die Weltmeisterschaft in Toronto genau umgekehrt: "Chess 4.6" siegte mit 4:0 Punkten, ohne gegen "Kaissa" spielen zu müssen, weil der russische Titelverteidiger gleich zum Auftakt von dem amerikanischen Programm "Duchess" geschlagen wurde. Wieder kam es zum anschließenden Schaukampf, den "Chess 4.6" diesmal nach einer wahren Abtauschorgie im Endspiel souverän gewann.

Eine bedeutende Rolle spielten die Chess-Programme auch in einer Wette. 1968 kam es während einer Diskussion zwischen dem Internationalen Schachmeister David Levy und zwei Professoren, die auf dem Gebiet der Künstlichen Intelligenz forschten, zu Meinungsverschiedenheiten über die Zukunftsperspektiven der Schachprogramme. Levy meinte, es könne in den nächsten zehn Jahren keinem Programm gelingen, ihn in einem Zweikampf über mehrere Runden zu besiegen. Seine Gesprächspartner hielten dagegen, und so kam es zur Wette mit einem Einsatz von 500 US-Dollar. Dieser Auseinandersetzung schlossen sich später weitere Wissenschaftler an, so daß schließlich Levys Ansicht gegen die von McCarthy, Michie, Papert und Koźdrowicki stand und die Wettsumme auf 2500 Dollar angestiegen war. Unter diesen Voraussetzungen kam es in Toronto vom 26. August bis 4. September 1978 zum auf sechs Runden angesetzten Duell zwischen David Levy und "Chess 4.7".

Die erste Partie begann mit einem Paukenschlag, als das Programm kurz nach der Eröffnung einen Springer opferte und einen sehr bedrohlichen Angriff auf den gegnerischen König startete. Aber selbst drei Bauern, die "Chess 4.7" zwischenzeitlich mehr hatte, reichten nicht zum Gewinn, und die Partie endete remis. Die beiden folgenden Begegnungen sahen Levy als überzeugenden Sieger, der getreu dem Motto "Tue nichts, aber tue es gut!" auf positionelle Fehler seines Gegners wartete und diese dann rigoros ausnutzte. Die vierte Runde ging er dann mit zwei Punkten Vorsprung etwas sorglos an und über-

sah prompt eine taktische Möglichkeit, die ihn die Partie kostete. Doch schon im darauffolgenden Spiel machte er alles klar, als er "Chess 4.7" ein weiteres Mal positionell an die Wand drückte und das Gesamtmatch mit 3.5:1.5 für sich entschied.

Diese Wette ist zwar der spektakulärste, aber bei weitem nicht der einzige Beitrag Levys zum Computerschach. In vielen Turnieren der 70er und frühen 80er Jahre fungierte er als Schiedsrich-

Levy contra Chess 4.7

ter und Kommentator. Inzwischen ist er sogar ins andere Lager übergewechselt und beteiligt sich mit einem eigenen Programm, das er zusammen mit Mitarbeitern seiner Software-Firma entwickelt hat, am Turniergehen. Seine aktive Rolle wird noch dadurch unterstrichen, daß er im Sommer diesen Jahres zum Präsidenten des Computerschach-Weltverbandes ICCA gewählt wurde. Diese Vereinigung existiert seit Mai 1978 und hat sich zum Ziel gesetzt, die Entwicklung des Computerschach und seine Verbreitung zu fördern. Fast alle auf diesem Gebiet aktiven Programmierer und/oder Organisatoren besitzen die Mitgliedschaft, aber auch unbeteiligte, lediglich am Geschehen interessierte Personen können der ICCA beitreten.

Doch kehren wir zu den Turnierereignissen zurück. Die Chess-Programme hatten eine Entwicklung in Gang gesetzt, die dazu führte, daß immer größere und schnellere Rechner und bald auch speziell für das Schachspiel konstruierte Maschinen zum Einsatz kamen. Die nächsten Programme, die sich anschickten, weitere Meilensteine auf dem Weg zum Großmeisterniveau zu setzen, bestätigten diesen Trend. Auf dem WM-Turnier 1980 in Linz (Österreich) ging der Titel zum ersten Mal an ein Programm namens "Belle", das auf einer solchen Spezialhardware lief. Eigens für das Schachspiel entwickelte Chips, die von einem PDP-11-Rechner gesteuert wurden, sorgten für eine Rechenleistung, mit der ca. 160 000 Stellungen pro Sekunde untersucht werden konnten. Gebaut und programmiert wurde dieses Kraftpaket von Ken Thompson und Joe Condon, die bei den Bell Telephone Laboratorien in New Jersey, USA, beschäftigt sind. Trotzdem garantierte dies nicht automatisch den Turniererfolg. Erstmals bei einer Weltmeisterschaft gingen zwei Programme punktgleich ins Ziel (mit 3.5 Punkten aus 4 Partien), "Belle" und das amerikanische Pro-

gramm "Chaos". So mußte eine Entscheidungspartie um die Vergabe des Titels durchgeführt werden. Diese Chance ließ sich "Belle" dann nicht entgehen. Den Exweltmeistern "Kaissa" bzw. "Chess 4.9" erging es nicht besonders gut: Mit nur 2 bzw. 2.5 Punkten aus 4 Partien landeten beide im Mittelfeld.

Aber auch "Belle" gelang es nicht, seinen WM-Titel zu verteidigen. Drei Jahre später mußte es ihn in New York im vierten Turnier abgeben. Robert Hyatt, Albert Gower und Harry Nelson hatten sich die Unterstützung eines Rechnerherstellers gesichert, der daran interessiert war, seine superschnellen Anlagen, die sogar mehrere Prozessoren parallel beschäftigten, durch komplexe Programme testen zu lassen. Dieser Glücksfall bescherte den Amerikanern einen CRAY I-XMP, der es, obwohl für allgemeine Anwendungen und daher ohne spezielle Schachhardware gebaut, an Geschwindigkeit mit "Belle" aufnehmen konnte. Bezeichnenderweise hieß das Programm "Cray Blitz". Es spielte in dem Turnier über fünf Runden nur einmal unentschieden, nämlich gegen "Nuchess". Da es in der Schlußrunde auch gegen "Belle" siegte, reichten die 4.5 Punkte zum ersten Platz. "Belle" teilte das Schicksal seiner "Vorgänger im Amt" und wurde mit 3:2 Punkten nur Sechster.

Damit sind wir mit unserem historischen Überblick schon fast in der jüngsten Vergangenheit, nämlich bei der 5. Weltmeisterschaft angekommen. Nun wird sich jedoch mancher Leser mit Recht fragen, wo in der bisherigen Aufzählung von Programmen und Turnieren, von Fakten und Geschichten die heutzutage so stark auftrumpfenden Mikros geblieben sind. In der Tat haben diese kleinen Geräte in den letzten Jahren gewaltig an Spielstärke aufgeholt, auch wenn sie es noch nicht ganz bis zur Spitze des Computerschach gebracht haben. Die meisten Großrechnerprogramme haben mittlerweile enorme Schwierigkeiten, den direkten Vergleich gegen ihre "kleinen Brüder" für sich zu entscheiden. Deren eigenständige Bedeutung (und das kommerzielle Interesse der Hersteller von Schachcomputern) hat außerdem dazu geführt, daß seit 1980 jährlich eine Mikrorechner vorbehaltene Weltmeisterschaft ausgetragen wird.

Schon 1978 machte erstmalig ein auf einem Mikro laufendes Programm nachhaltig auf sich aufmerksam. Das amerikanische Ehepaar Dan und Kathe Spracklen hatte in Privatinitiative einen Jupiter-Mikrocomputer mit Z80-Prozessor in Assembler programmiert und nannte das Produkt "Sargon II". Ein Turnier für Mikrorechner-Programme im

März 1978 gewannen die beiden locker mit 5:0 Punkten. Für großes Aufsehen sorgten sie jedoch einige Monate später, als sie sich beim 9. ACM-Turnier hinter den Giganten "Belle" und "Chess 4.7" zusammen mit zwei anderen Großrechnerprogrammen (darunter auch der Vorläufer des späteren "Cray Blitz") den dritten Platz teilten. Damit wurde Schach auf Mikros "hoffähig", und wartete beim Auftreten solcher Programme in den Turnieren gespannt auf weitere Überraschungen. Diese blieben auch nicht aus. Bei der WM 1980 in Linz belegte "Challenger", wie "Sargon II" nach der Kommerzialisierung bei Fidelity Electronics nun genannt wurde, zwar mit nur einem Remis den letzten Platz, aber beim ACM-Turnier im gleichen Jahr erreichte es wiederum den dritten Platz. Absoluter Höhepunkt für die Fidelity-Geräte war der Turniersieg beim US-Open-Turnier 1985 für Schachprogramme, als renommierte Gegner wie z.B. "Belle" geschlagen wurden.

Auch andere starke Wettbewerber tauchten in der Mikro-Szene auf. Besonders hervor taten sich die amerikanische Firma Novag mit den Programmen "Mychess" und "Super Constellation" von David Kittinger und der deutsche Hersteller Hegener + Glaser aus München mit der Mephisto-Reihe, entwickelt von Thomas Nitsche/Elmar Henne und später Richard Lang. Diese "Großen Drei", Fidelity, Novag und Hegener + Glaser, macht auch in den letzten Jahren die vorderen Ränge der Mikrocomputer-Weltmeisterschaften unter sich aus und sicherten sich – nicht zuletzt aufgrund dieser Turnierfolge – die größten Anteile am immer größer werdenden Schachcomputermarkt. Dies trifft insbesondere auf die Mephisto-Geräte zu, die bei der letzten Mikro-WM 1985 in Amsterdam, allerdings in Abwesenheit von Fidelity, das übrige Teilnehmerfeld förmlich deklassierten, als sie mit drei Startern die ersten drei Plätze belegten und dabei 22 von 24 möglichen Punkten erreichten!

Mephisto an der Spitze

Wenden wir uns nun zum Abschluß dieses Artikels dem letzten Großereignis zu, das erst vor wenigen Monaten hier in Deutschland über die Bühne gegangen ist. Die Austragung großer Computerschachturniere ist ein kostspieliges Unternehmen, denn es müssen geeignete Turniersäle bereitgestellt, Teilnehmer und Organisatoren untergebracht und vor allem für die außerhalb des Turnierortes stationierten Groß-

rechner Kommunikationsleitungen eingerichtet werden. Doch dank des finanziellen Engagements der Kölner Messgesellschaft, in deren Hallen das Turnier ablief, und der Firma Hegener + Glaser konnte die 5. Weltmeisterschaft nach Köln geholt werden. Die Veranstaltung stand übrigens unter der Schirmherrschaft des Deutschen Schachbundes DSB und des Weltschachverbandes FIDE, die damit der wachsenden Bedeutung des Computerschach Anerkennung zollten.

23 der weltbesten Schachprogramme und -computer waren dabei, als am 11. Juni um 10 Uhr die Bretter für die erste Runde freigegeben wurden. Von den vier früheren Weltmeistern war zwar

Mit Titelverteidiger Cray Blitz

nur der aktuelle Titelverteidiger "Cray Blitz" am Start, aber für starke Konkurrenz war trotzdem gesorgt. Diese wurde insbesondere von dem neuen amerikanischen "Wunderprogramm" "Hitech" des Ex-Fernschachweltmeisters Prof. Hans Berliner erwartet, der zusammen mit Studenten an der Carnegie-Mellon University in Pittsburgh, Pennsylvania, eine superschnelle Schachmaschine aus speziell entwickelten VLSI-Chips gebaut hatte (VLSI = very large scale integrated; Elektronikbauteile von sehr hoher Packungsdichte). Die daraus resultierende Geschwindigkeit von ca. 200000 analysierten Stellungen pro Sekunde machte zusammen mit dem Schachwissen des Fachmannes Berliner "Hitech" zum großen Turnierfavoriten – und das trotz der Anwesenheit von "Cray Blitz". Aber auch andere aufstrebende Programme wie z.B. "Be-Be", ebenfalls ein Rechner mit Schach-Hardware des Amerikaners Tony Scherzer (Vizeweltmeister von 1983), der amtierende Mikro-Weltmeister "Mephisto" oder das Programm des Kanadiers Jonathan Schaeffer, "Sun Phoenix", für das 20 parallel arbeitende Kleinrechner benutzt wurden, hatten ihre Chancen. Neben "Mephisto" gab es noch drei weitere deutsche Teilnehmer: "Schach 2.7" von Matthias Engelbach, seit der WM 1980 ständig dabei, "Bobby" von Günther Schröder und mir (zweite Weltmeisterschaftsteilnahme nach New York) und den Newcomer "Chat" von Wolfgang Delmare. Alle drei Programme liefen während des Turniers auf den Universitätsrechnern, auf denen sie auch entwickelt worden waren.

Die erste Runde verlief fast ohne Überraschungen. Die Favoriten mußten noch nicht gegeneinander antreten und setz-

ten sich erwartungsgemäß durch. Die einzige Ausnahme war die Niederlage von "Sun Phoenix" gegen das holländische Mikrorechnerprogramm "Rebel" von Ed Schröder und Jan Louwman, das nach nur 22 Zügen eine schöne Kombination sah, nach der die Stellung für die Kanadier sofort aufgabereif war. Im zweiten Durchgang kam es zu einem der sensationellsten Ergebnisse dieses Turniers. Das Programm "Bobby" der Technischen Universität Braunschweig, etwa 500mal langsamer als "Cray Blitz", konnte dieses in einer positionell gut geführten Partie niederhalten und ließ damit dessen Chancen auf eine Titelverteidigung auf ein Minimum sinken. Die Partie nahm folgenden Verlauf:

	Weiß:	Bobby
	Schwarz:	Cray Blitz
1.	e2-e4	e7-e5
2.	Sb1-c3	Sg8-f6
3.	f2-f4	d7-d5
4.	f4:e5	Sf6:e4
5.	Sg1-f3	Lf8-e7
6.	d2-d4	0-0
7.	Lf1-d3	f7-f5
8.	e5:f6	Le7:f6
9.	0-0	Sb8-c6
10.	Sc3:e4	d5:e4
11.	Ld3:e4	Sc6:d4
12.	c2-c3	Sd4:f3+
13.	Le4:f3	Dd8:d1
14.	Tf1:d1	Tf8-e8
15.	Lc1-f4	c7-c6
16.	g2-g4	Lf6-d8
17.	Kg1-g2	Lc8-e6
18.	a2-a4	Te8-f8
19.	Lf4-d6	Tf8-f7
20.	Td1-e1	Le6-d7
21.	Ta1-d1	Ld8-h4
22.	Te1-e2	Kg8-h8
23.	c3-c4	Kh8-g8
24.	Td1-d3	Ta8-d8
25.	b2-b4	Td8-c8
26.	Te2-e3	Lh4-g5
27.	Te3-e1	Tc8-d8
28.	Ld6-c5	b7-b6
29.	Lc5-d6	Lg5-h4
30.	Te1-e5	Lh4-f6
31.	Te5-e2	Lf6-g5
32.	h2-h3	Td8-c8
33.	a4-a5	b6:a5
34.	Te2-e5	Lg5-d8
35.	b4:a5	Ld8-f6
36.	Te5-e1	Lf6-d8
37.	a5-a6	Ld8-h4
38.	Te1-e2	Tc8-d8
39.	Ld6-c5	Lh4-g5
40.	Te2-b2	Td8-c8
41.	Tb2-b7	Ld7-e6
42.	Td3-d6	Le6:c4
43.	Td6:c6	Tc8-d8
44.	Tb7:f7	Kg8:f7
45.	Lc5:a7	Td8-d2+
46.	Kg2-g3	Lc4-d5
47.	Tc6-c7+	Kf7-g8
48.	Lf3:d5	Td2:d5

49.	La7-b6	Td5-d8
50.	Tc7-b7	Td8-d3+
51.	Kg3-g2	Td3-d2+
52.	Kg2-f3	Td2-d3+
53.	Kf3-e4	Td3:h3

Hier gab Schwarz auf, weil nach a7, Ta3, Tb8+, Kf7, Da8 der Turm geopfert werden muß.

Auch "Mephisto" erlitt einen Rückschlag in der Partie gegen "Plymate", den Schachcomputer des Schweden Ulf Rathsman. Von den Favoriten waren nun nur noch "Hitech", das gegen "Schach 2.7" die vielleicht schönste Partie des gesamten Turniers lieferte, und "Be-Be" ohne Verlustpunkte. Nach der dritten Runde war dann die Spitzengruppe auf zwei Programme reduziert: "Hitech" setzte sich gegen den "Mephisto"-Bezwinger "Plymate" durch, und das Mikrorechnerprogramm "Rebel" sorgte weiter für Furore, als es "Bobby" aus der Eröffnung heraus in überzeugender Manier schlug. "Cray Blitz" konnte durch einen Sieg über "Be-Be" den Anschluß wahren. Ein Teilnehmer mußte in der Zwischenzeit kapitulieren: Der Ungar Attila Kovacs hatte nicht die Erlaubnis bekommen, seinen Atari-ST-Rechner mit nach Deutschland zu bringen. So versuchte er sich in der zweiten Runde (in der ersten war er wegen der ungeraden Anzahl der Teilnehmer spielfrei) auf einem Leihgerät, was prompt zu Schwierigkeiten und einem kampflosen Partieverlust führte. Damit war das Turnier für sein Programm "Kempelen Atari" leider schon gelaufen, bevor es eigentlich begonnen hatte.

Die vierte Runde mußte nun schon die ersten Vorentscheidungen bringen, und so kam es auch. "Hitech" schlug "Rebel" und hatte damit vor der letzten Partie einen ganzen Punkt Vorsprung vor dem Verfolgerfeld. Dazu zählten "Cray Blitz" nach seinem zweiten Sieg in Folge, dann "Be-Be", "Bobby" nach gewonnenem deutschen Duell gegen "Mephisto", "Sun Phoenix", das sich von der Auftaktniederlage gut erholt hatte und zum dritten Mal hintereinander punktete, und natürlich immer noch "Rebel".

Drei Auseinandersetzungen der Schlußrunde, in denen sich die ersten sechs der Tabelle gegenüberstanden, sollten nun über die Vergabe des Titels entscheiden. Die Paarungen lauteten:

Cray Blitz	–	Hitech
Rebel	–	Be-Be
Bobby	–	Sun Phoenix

Die besten Aussichten hatte zweifelsohne "Hitech", dem ein Unentschieden zum ersten Platz reichte. Die deutschen Hoffnungen sanken schon sehr früh, als nach der Eröffnung einer von "Bobbys" Türmen auf Abwege geriet und bis zum

Schluß der Partie ein ständiges Sorgenkind blieb. In auswegloser Stellung gaben Günther Schrüfer und ich nach ungefähr vier Stunden Spielzeit auf. Mit 3:2 Punkten (davon 2 gegen die amtierenden Weltmeister "Cray Blitz" und "Mephisto") reichte es "Bobby" schließlich als bestem deutschen Programm zum

◆ Bobby erreichte 6. Platz

sechsten Platz. Inzwischen hatten "Cray Blitz" und "Rebel" Gewinnstellungen erreicht, so daß die Entscheidung zwischen diesen beiden fallen mußte. Falls beide Programme durchhielten, mußte die bessere Buchholzwertung, d.h. die Summe der Punkte, die die jeweiligen Gegner erreicht hatten, den Ausschlag geben. Dies hätte zum Turniergewinn für "Rebel" gereicht, womit erstmals in der Geschichte des Computerschach ein Mikrorechner den WM-Titel erhalten hätte. Doch das holländische Programm ließ einen gegnerischen Freibauern außer acht und verspielte die Partie. "Cray Blitz" konnte dagegen seinen Vorteil gegen "Hitech" behaupten, sicherte sich durch diesen Sieg die Meisterschaft und ist damit der erste Weltmeister, der seinen Titel verteidigen konnte. Die Abschlusstabelle hatte folgendes Bild:

		Punkte	Buchholz
1.	Cray Blitz	USA	4 14.5
2.	Hitech	USA	4 13.5 (10)
3.	Be-Be	USA	4 13.5 (8)
4.	Sun Phoenix	Kanada	4 12.5
5.	Rebel	NL	3 17.5
6.	Bobby	BRD	3 15.5
7.	Plymate	S	3 14.5
8.	Mephisto	BRD	3 13
9.	Dutch	NL	3 11.5
10.	Nona	NL	3 7.5
11.	Advance	GB	2.5 14.5
12.	Lachex	USA	2.5 14
13.	Ostrich	Kanada	2.5 11
14.	Schach 2.7	BRD	2 15
15.	Cyrus 68 k	GB	2 13.5
16.	Vaxchess	GB	2 11
17.	Chat	BRD	2 10
18.	BCP	GB	1.5 15
19.	Enterprise	DK	1.5 12
20.	Awit	Kanada	1.5 10.5
21.	Rex	USA	1 10.5
22.	Shess	NL	1 8
23.	Kempelen Atari	Ungarn	0 2

Für diejenigen Leser, die durch die vorangegangenen Seiten auf den Geschmack gekommen sind, folgt nun noch eine kleine Literaturübersicht. Ich habe die leider fast ausschließlich englischsprachigen Quellen in drei Klassen eingeteilt.

Bücher: Sie sind vor allem für diejenigen interessant, die die Geschichte des Computerschach detaillierter nachlesen

und auch die eine oder andere Partie nachspielen wollen. Darüber hinaus enthalten sie mehr oder weniger umfangreiche Teile, in denen der Aufbau eines Schachprogramms erläutert wird. Wenn es reizt, ein eigenes Programm zu schreiben, der ist wahrscheinlich mit dem Data-Becker-Buch am besten bedient, das u.a. ein komplettes Basic-Listing eines Schachprogramms bietet, das auch für die CPC-Rechner voll geeignet ist.

D. Levy, M. Newborn:
All About Computers and Chess
Springer-Verlag

P. Frey:
Chess Skill in Man and Machine
Springer-Verlag

R. Bartel, H.-J. Kraas, G. Schrüfer:
Das große Computerschachbuch
Data Becker

Artikel: Hierbei handelt es sich um ca. 10-40 Seiten starke Berichte, die überwiegend in wissenschaftlichen Zeitschriften erschienen sind. Als Einstiegslektüre sind sie kaum geeignet und darüber hinaus oft nur schwer zu beschaffen. Verweise auf bedeutende Artikel können den Anhängen der genannten Bücher entnommen werden.

Zeitschriften: Wer über das aktuelle Geschehen im Computerschach informiert sein will, wird nicht umhin kommen, sich Zeitschriften zu besorgen. Ohne Zweifel ist das vierteljährlich erscheinende ICCA-Journal, das offizielle Mitteilungsorgan des Computerschach-Weltverbandes, die Nummer 1. Ankündigungen von Turnieren und anderen Veranstaltungen, Literaturbesprechungen, Turnierberichte (inkl. Partien) und wissenschaftliche Artikel sind die wichtigsten Beiträge in diesen Heften. Der Bezug dieser Zeitschrift, der übrigens automatisch mit der Mitgliedschaft in der ICCA gekoppelt ist, kann über folgende Anschrift erfolgen:

Dr. H. J. van den Herik
Delft University of Technology
Department of Mathematics and Informatics
Julianalaan 132/Room 2.115
2628 BL Delft/The Netherlands

Mehr an die Schachcomputerbesitzer wendet sich die deutschsprachige Zeitschrift "Computer-Schach & Spiele". Hier finden sich Tests von Schachcomputern, Turnierberichte und sogar eine Computerbörse, in der die Leser Geräte anbieten und Kontakte knüpfen können. Ein Abonnement gibt es beim Falken-Verlag GmbH
Postfach 1120
Schöne Aussicht 21
6272 Niedernhausen
Text und Fotos: H.-J. Kraas

STARGATE ist da!

Science-fiction-Roman auf Diskette

Die Zukunft hat begonnen, machen Sie Ihren Computer zum Textleseterminal. Zwei brandneue Science-fiction-Romane wurden auf Diskette geschrieben und stehen nun zum Lesen bereit.

"Menschen unerwünscht", so heißt die erste Geschichte, und weiter geht's mit "Der Clan der Rebellen", eine nicht minder spannende Science-fiction-Story.

Der Preis, zwei Romane auf einer 3"-Diskette, nur 24.80 DM

Versand erfolgt per Vorkasse zzgl. 2.- DM

Bestellungen oder weitere Informationen

VAN DER ZALM SOFTWARE

Programm-Entwicklung und Vertrieb
Efriede van der Zalm
Schieferstätte, 2949 Wangerland 3,
Telefon 044 61 / 55 24

KABEL-weg



KABEL-weg macht Ihren Schneider PC zu einem richtigen PC. Anschlußkabel, Interface und was sonst herumhängt verschwindet im Kabeltunnel. Oben stehen Bildschirm und zusätzliche Diskettenlaufwerke. KABEL-weg schafft Platz auf Ihrem Arbeitstisch. Und die ganze Anlage wird mit 1 Schalter eingeschaltet.

Maße: 52 x 49 cm
Lackierung: matt-schwarz

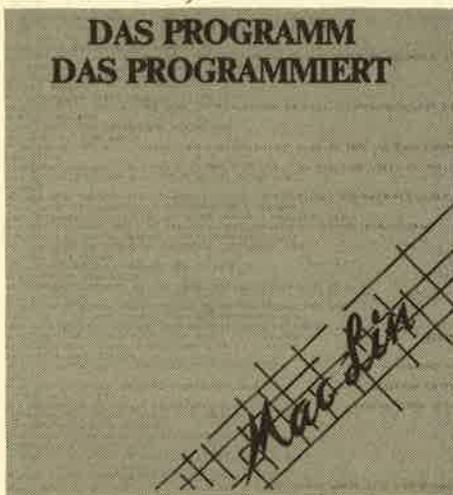
Lieferung gegen Vorkasse: DM 114,-
(PSch-Konto Köln 593 79-501 oder Scheck)
Lieferung gegen Rechnung: DM 124,-
Preise incl. Versandkosten

Prospekt KM gratis

PLAN
Datenverarbeitung GmbH.
Friedrich-Ebert-Str. 134
5600 Wuppertal 1

Müheless Programme schreiben?

Programmgeneratoren sind der neueste Hit beim Programmieren



Haben Sie schon einmal etwas von einem sog. Programmgenerator gehört? Nun, "McLin-Nucleus" ist ein solcher Exot unter den wenig bekannten Anwenderprogrammen.

Ein Programmgenerator setzt keinerlei Programmierkenntnisse voraus. Sie müssen nur wissen, was Sie wollen bzw. was das Programm können soll. Im Dialog werden diese Angaben abgefragt, und der Programmgenerator erzeugt im vorliegenden Fall ein Programm in Malard-Basic, wobei der Code vom Anwender list- und veränderbar ist. Das alles hört sich recht gut an, doch der totale Computerlaie wird mit diesem Programm kaum zu höheren Programmierweihen gelangen. Eines kann man jedoch garantieren: Wer sich wirklich intensiv mit "McLin" beschäftigt, kann das Programm nach einiger Zeit sicher anwenden.

Die Tradition von "McLin-Nucleus" kann man bis in die siebziger Jahre zurückverfolgen. Mittlerweile läuft es mit großem Erfolg unter MS-DOS und auch unter CP/M auf dem Joyce-Computer. Der Käufer erwirbt einen etwas unhandlichen Aktenordner und drei Disketten mit den Programmteilen für die Erzeugung individueller Software. Genau das ist das Ziel des Generators. Der Anwender soll nicht mit vorgekauften Programmen vorlieb nehmen und seine Bedürfnisse an das Programm anpassen, sondern er muß sich eine Anwendung selbst schneiden können, die seinen

gesamten Bedürfnissen so gerecht wird, daß er nirgendwo zurückstecken muß.

Das Anwendungsgebiet von "McLin-Nucleus" liegt hauptsächlich in der Datenverwaltung bzw. Weiterverarbeitung und Ausgabe von Daten, der eigentlichen Stärke des Computers. Nachdem alles angeschlossen ist und man einige Dienstprogramme auf die Startdiskette kopiert hat, kann es eigentlich losgehen.

Getestet wurde der Generator auf dem PCW 8256 mit einem Laufwerk, so daß er die RAM-Disk als zweites anspricht. Die Programmdiskette liegt also im Laufwerk, und die RAM-Disk fungiert als Speicher für das fertige Programm sowie als Zwischenspeicher für die notwendigen Parameter. Durch Ändern einiger Parameter kann auch ein zweites Laufwerk unterstützt werden. Bei der Lösung mit der RAM-Disk ist auf jeden Fall immer darauf zu achten, daß ihr Inhalt nach Beendigung der Arbeit auf einer richtigen Diskette gesichert wird. Ansonsten kann es ein böses Erwachen nach dem Ausschalten bzw. bei einem Reset des Computers geben. Daher erweist sich die Lösung mit der RAM-Disk bei näherer Betrachtung als nicht gerade sicher, denn sobald das System abstürzt oder der Strom nur für Sekundenbruchteile ausfällt, ist die ganze Arbeit verloren. Wer in Erwägung zieht, sehr oft und intensiv mit dem Generator zu arbeiten, sollte sich vorher ein zweites Laufwerk anschaffen, sofern er nicht schon den Joyce mit 512 KByte RAM und 2 Laufwerken besitzt.

Wenn wir jetzt ein wenig die Entstehungsgeschichte eines Programms mit "McLin-Nucleus" betrachten wollen, müssen wir uns darüber im Klaren sein, daß die Möglichkeiten, die sich aus den Basic-Befehlen ergeben, ebensowenig wie alle Fähigkeiten von "McLin-Nucleus" aufzuzählen sind. Daher sei im folgenden nur der generelle Weg eines Programms beschrieben, bis es der Generator fertig auf Diskette schreibt.

Der Generator ist voll Menü-orientiert und steht im krassen Gegensatz zur augenblicklich überschwappenden Pull-Down-Menü- und Mausewelle. Trotzdem findet man sich im Programm nach einiger Zeit hervorragend zurecht, auch

wenn man bisher durch grafische Benutzeroberflächen verwöhnt war. Nach Eingabe des Datums begeben wir uns zu einem umfangreichen Menü, das für die ersten Eingaben nur Punkt für Punkt durchgearbeitet werden muß. Man gibt dem Programm einen Namen, legt ein sog. Anwenderkürzel fest, definiert die zu erzeugende Datei und macht Angaben über die Beschaffenheit jedes einzelnen Datenfeldes. Auf Wunsch kann im Programm auch noch die Option zur Verknüpfung mit anderen Dateien angegeben werden. Wer also für eine größere Auswertung mehrere Dateien benötigt, ist mit diesem Menüpunkt bestens bedient. Am Ende dieser sog. Dateidefinition steht ein Ausdruck, der noch einmal die wesentlichen Daten schwarz auf weiß festhält. Hier liegt übrigens ein kleiner Schwachpunkt des Generators, denn wer den Menüpunkt anwählt, wird sofort vom Rattern des Druckers überrascht. Egal ob Papier einliegt oder nicht, der Computer schreibt Walze und Druckkopf in Grund und Boden. Eine kleine Abfrage des Druckerstatus, und schon wäre die Sache erledigt gewesen.

Die Auswertung der Daten bietet nun so unzählig viele Möglichkeiten, daß es Monate gedauert hätte, alle Features durchzutesten und genau zu beschreiben. Es handelt sich dabei allerdings in der Regel um die Möglichkeiten, die auch in einem fertig gekauften Produkt vorhanden sind: Erstellen und Ausdrucken von sortierten Listen, Ausgabe von Etiketten, Erstellen von Formularen, Mail-Merge und Rechnen mit numerischen Daten. Einem kompletten Dateiverwaltungsprogramm nach eigenen Bedürfnissen steht also nichts mehr im Wege.

Die nächste Station auf dem Weg zum fertigen Programm ist die Eingabemaske. Der Anwender legt fest, welche Felder zur Eingabe bereit stehen sollen und wo er sie plazieren will. Danach erzeugt "McLin-Nucleus" schon einmal die Module für die Dateneingabe, die zusammen mit anderen Dienstprogrammen auf einer Diskette das Eingabeprogramm ergeben.

"McLin-Nucleus" ist ein hervorragendes Programm, aus dem der kreative An-

wender eine Menge an Leistung heraus-
holen kann. Software wird so zu dem,
was sie sein soll, nämlich zu einem
Werkzeug, das man speziell für seine Be-
dürfnisse angefertigt hat und damit ar-
beiten kann, ohne faule Kompromisse zu
schließen. Es dürfte deutlich geworden
sein, daß "McLin-Nucleus" weit mehr
kann, als nur Programme zur Schallplat-
ten-Verwaltung zu erstellen. Software
für die Steuererklärung, Lagerverwal-

tung und Fakturierung rückt in greifbare
Nähe. Dann erscheint auch der Preis
nicht mehr zu hoch, denn bei den Kos-
ten, die man für Spezialsoftware auf-
wenden muß (ab 1000 DM aufwärts),
können Kleinbetriebe und Selbständige
froh sein, diese Möglichkeit zur Pro-
grammierstellung gefunden zu haben.
Nur muß man außer den 475 DM noch
Geduld, Arbeit und sehr viel Zeit inve-
stieren.

System: Joyce PCW 8256/8512
Hersteller: Falco Vertriebs GmbH
Preis: 475.- DM

Bezugsquelle:
SFK-Elektro
Delstener Straße 23
5800 Hagen

Thomas Tai

Do it yourself: 512 KByte RAM für den Joyce PCW 8256

Mit ca. 80.- DM kann jeder seinen kleinen Joyce auf 512 KByte aufrüsten



diesen Bausteinen, ein blankes Metall-
stück, die Heizung oder die Wasserlei-
tung berühren, damit sich eventuell am
Körper vorhandene statische Elektrizität
entladen kann.

Vor den nächsten Handgriffen ist un-
bedingt der Netzstecker zu ziehen, denn
Sicherheit ist erstes Gebot. Nun wird die
Rückwand des Monitors durch Lösen
von sechs Halteschrauben abgenom-
men. Im Inneren kann man ein Metallge-
häuse entdecken, an dem drei Steckver-
binder angebracht sind. Diese Steckver-
binder werden vorsichtig entfernt. Zu-
sätzlich müssen noch die drei Erdungs-
leitungen abgeschraubt werden. Am
besten macht man sich eine Skizze, die
die Lage der Stecker und Leitungen ent-
hält, damit beim Zusammenbau keine
Schwierigkeiten auftreten. Nun kann

Sockeln. Diese Sockel müssen nun mit
den neuen ICs bestückt werden, und
zwar so, daß die Kerbe in die gleiche
Richtung zeigt, wie bei den bereits vor-
handenen Speicherbausteinen.

Abschließend müssen die kleinen
Schalter in der Mitte der Platine so ein-
gestellt werden, daß diejenigen mit der
Bezeichnung A und C sich in der Stel-
lung OFF befinden, während die Schal-
ter B und D in die Stellung ON zu bringen
sind. Damit ist der Eingriff abgeschlos-
sen. Nun geht es an den Zusammenbau,
der sehr sorgfältig erfolgen muß. Zuerst
das Gehäuse zusammenschrauben und
im Monitor unterbringen, dann die Er-
dungsleitungen festschrauben (auf gu-
ten Kontakt achten), die Steckverbinder
anbringen (gleiche Farben gehören zu-
sammen) und zum Schluß die Rückwand
am Monitor anbringen.

Nach dem Einschalten und dem
Durchlauf des "Bootgen-Programmes"
muß für die RAM-Floppy eine Kapazität
von 368 KByte angezeigt werden.

Folgende Fehler können auftreten:

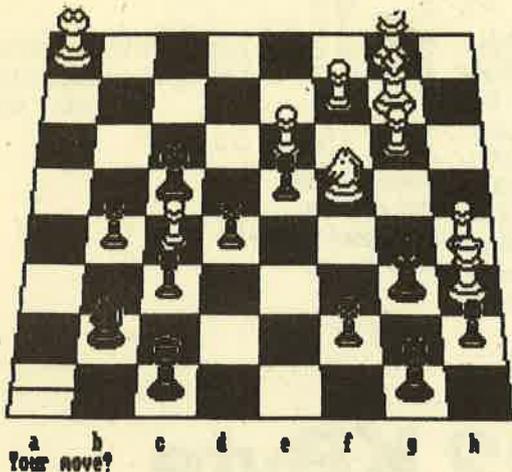
1. Es erscheint keine Meldung: Die
Steckverbinder sind falsch aufge-
steckt oder die Schalter auf der Plati-
ne wurden falsch eingestellt.
2. Die Meldung ist genau wie vor dem
Eingriff: Die Schalter auf der Platine
wurden falsch eingestellt.
3. Der Rechner stürzt nach kurzer Zeit
ab: Ein Speicherbaustein ist defekt
bzw. die Steckverbinder sitzen zu
locker auf der Platine.

H. P. Schwaneck

Wer den Speicher seines Joyce 8256
verdoppeln möchte, muß nicht viel
Geld investieren. Ca. 80.- DM und eine
Stunde Arbeitszeit genügen vollauf. Zu-
nächst besorgt man sich die passenden
Speicherbausteine. Das wären 8 Bau-
steine vom Typ 41256-15. Die 15 in der
Typenbezeichnung ist dabei wichtig,
denn diese Zahl gibt die Zugriffszeit an.
In diesem Falle sind das 150 Nanosekun-
den. Die 8 Speicherchips kosten je nach
Einkaufsquelle ca. 80.- DM. Ein wenig
Marktforschung in einschlägigen Elek-
tronik-Zeitschriften hilft dabei, Kosten
zu sparen. Die Speicherchips sollten al-
lerdings sehr vorsichtig behandelt wer-
den, da es sich um sogenannte CMOS-
Bausteine handelt. Diese Bausteine sind
äußerst empfindlich gegen statische
Aufladungen. Daher sollte man selbst,
grundsätzlich vor dem Hantieren mit

Problemlos ausgerüstet

das Gehäuse nach hinten aus dem Moni-
tor herausgezogen werden. Um es zu
öffnen, sind am äußeren Rand mehrere
Schrauben zu lösen. Auch hier ist eine
Skizze für den späteren Zusammenbau
sehr hilfreich. Im Inneren des Gehäuses
ist die eigentliche Rechnerplatine zu fin-
den. Wenn man diese Platine so vor sich
hinlegt, daß die Steckverbinder vom Kör-
per weg zeigen, dann befinden sich auf
der rechten Platinen-Seite eine Reihe
von acht Speicherbausteinen und links
daneben eine Reihe von acht leeren IC-



In der Redaktion häufen sich die Leserbriefe, in denen nach Spiel-Software für die Joyce-Computer gefragt wird. Viel gibt es aus dieser Richtung noch nicht zu berichten. In England hat es aber bereits die ersten Umsetzungen erfolgreicher Programme gegeben. Sobald uns Testmuster zur Verfügung stehen, werden wir auch diese vorstellen.

Heute will ich mich auf ein neues Schachprogramm beschränken, das aber einiges zu bieten hat. Das unter CP/M 3.0 laufende "Colossus Chess 4.0" aus dem Hause CDS Software Ltd. ist nach "3D Clock Chess" das zweite Programm dieser Art für die Joyce-Rechner. Es ist vollständig in Maschinensprache geschrieben. Man merkt sofort, daß der Autor M. P. Bryant sein Handwerk versteht. Im Anhang der Anleitung ist vermerkt, daß "Colossus" gegen zahlreiche andere Schachprogramme antreten mußte. Alle Gegner, gleich um welche Rechner oder Programme es sich handelte, wurden geschlagen. Von jeweils 16 Partien gewann "Colossus" z.B. gegen "Mychess 2.0" (C 64) 10:6, gegen "Superchess 3.5" (Spectrum) 12:4, gegen "Parker Chess" (Atari) 14:2 und gegen "3D Clock Chess" (Joyce) und viele andere 16:0 – ein beeindruckendes Ergebnis, das schon andeutet, wie spielstark dieses Programm ist. Dazu trägt auch eine Eröffnungsbibliothek von rund 3000 Stellungen und eine Rechengeschwindigkeit von 170 Zügen pro Sekunde bei. Auch sonst bietet "Colossus Chess 4.0" sehr viel.

Die grafische Darstellung des Bretts und der Figuren ähnelt "3D Clock Chess", ist aber nicht ganz so gut. Leider kann man auch hier nicht auf zweidimensionale Darstellung umschalten. Die Züge werden über die Cursor-Tasten gesteuert. Hierzu wird der Cursor auf die zu bewegende Figur gebracht, mit RETURN bestätigt, auf die neue Position gebracht, erneut bestätigt, dann wird die Bewegung ausgeführt. Die jeweils letzten 7 Züge beider Seiten wer-

Spielstärke und Komfort sehr gut!

Für den Joyce: Colossus Chess

	White Opponent 00:00:14	Black Colossus 00:00:03
23	a1xa6	e7-b7
24	a6-a5	b7xc3
25	a5xb6	b7-h6
26	g8-f4	f3-h3
27	f4-g3	h6-h5
28	h5-e6	e8-a8
29	d2-c1	h6-h4

den auf der rechten Monitorseite ausgedruckt. Hier sind auch zwei Schachuhren installiert. Unterhalb des Bretts befindet sich Platz für spielbegleitende Informationen wie z.B. Zugvorschläge. Das Programm zeigt hier auch an, wieviele Stellungen gerade durchgerechnet wurden, welches der zur Zeit beste Zug wäre usw. Wenn der Joyce "denkt", kann man ihn mit einem Druck auf EXIT unterbrechen. Dann wird der aktuelle Zug ausgeführt.

Neben den Cursor-Tasten sind auch andere Tasten belegt, mit denen verschiedene Optionen aufgerufen werden können. Dem heutigen Standard entsprechend, kann der Spieler sich eine Wunschstellung aufbauen, sofern diese den Schachregeln entspricht. Er kann auch einen Zug zurücknehmen, diesen wieder ausführen lassen oder die Uhren beeinflussen.

Interessant ist eine Option, die sich INVISIBLE nennt. Hier kann man entweder die weißen oder die schwarzen Figuren oder beide Seiten unsichtbar machen. Für Anfänger eignet sich besonders die Option L, was für LEGAL MOVE steht. Wird L gedrückt, prüft das Programm alle Möglichkeiten, die der Spieler mit der Figur hat, auf die der Cursor zeigt.

Nachstehend noch einige Optionen in Kurzform. O erlaubt dem Spieler, die Seite zu wechseln, also mit den Figuren des Gegners zu spielen. P veranlaßt "Colossus", gegen sich selbst anzutreten. R steht für REPLAY, womit z.B. eine abgespeicherte Position von Anfang an automatisch nachgespielt werden kann. S schaltet "Colossus" ab; der Spieler kann jetzt beide Seiten selbst übernehmen. D ruft die Load/Save-Option auf. U zwingt "Colossus", den gerade gemachten Zug zurückzunehmen, um den nächstbeste-

ren (also in der Regel den schlechteren) auszuführen. V schaltet den Piepser ab. W ruft die Druckeroption auf. Hier stehen 3 Möglichkeiten zur Verfügung. Es kann entweder das Brett ausgedruckt, alle bisher gemachten Züge aufgelistet oder ein fortlaufendes Protokoll geschrieben werden. Mit der letzten Option kann man den Spielmodus einstellen, also wählen, ob man Blitzschach oder im Turniermodus spielen will, ob eine normale Runde gespielt oder ein Problem analysiert werden soll usw. Auch hier sind die Möglichkeiten vielfältig.

In dieser Preisklasse ist "Colossus Chess 4.0" wohl einmalig. Spielstärke und Komfort sind einfach hervorragend. Als Zugabe sind auf der Diskette übrigens noch 34 Partien und 20 Problemstellungen abgespeichert, mit denen man sich die Zeit vertreiben kann. Hier kann man unter anderem verfolgen, wie sich Victor Korchnoi gegen einen CRAY 1 geschlagen hat. Abgesehen von der Grafik, die etwas besser sein könnte, halte ich "Colossus Chess 4.0" für das beste Schachprogramm, das zur Zeit angeboten wird.

System: Schneider Joyce PCW 8256
 Hersteller: CDS Software
 Bezugsquelle: Joysoft
 Rolf Knorre



SIREN SOFTWARE

DISC MATE

DISC UTILITIES PACKAGE

AMSTRAD PCW 8256/8512

Drei nützliche Joyce-Utilities

Mit Directory Editor, Copyfile und Zipdisc

Bisher gibt es auf dem Software-Markt noch wenig Angebote für Joyce-Computer. Eines davon trägt die Bezeichnung "Disc Mate". Auf der Programmdiskette sind drei Utilities abgespeichert, die das Leben mit dem Joyce etwas erleichtern sollen. Die einzelnen Files heißen: Directory Editor, Copyfile, Zipdisc. Getestet habe ich die Programme auf dem PCW 8256, sie laufen aber natürlich auch auf dem 8512.

Das File "Zipdisc" kann vom Anwender nicht beeinflusst werden. Es dient lediglich der Beschleunigung des Floppy-Laufwerks. Hat man die Programmroutine geladen, wird die Arbeitsgeschwindigkeit um ca. 20 % erhöht. Da im Speicher kein Platz belegt wird, kann "Zipdisc" mit jedem anderen Programm in Verbindung gebracht werden.

Ergiebiger als "Zipdisc" ist "Copyfile". Wie man schon dem Namen entnehmen kann, handelt es sich um ein Kopierprogramm. Die Handhabung ist denkbar einfach. Nachdem "Copyfile" geladen ist, muß die Originaldiskette eingelegt werden. Jetzt werden nacheinander alle Files, die auf der Diskette vorhanden sind, auf den Bildschirm gebracht. Der Benutzer kann nun entscheiden, ob ein File kopiert werden soll oder nicht. Die Abfrage der Diskette wird so lange fortgeführt, bis entweder kein weiteres File mehr vorhanden oder der Speicher des Joyce voll ist. Im letzteren Fall muß eine neue Diskette eingelegt werden, auf die dann kopiert wird. Erst jetzt sucht das Programm auf der wieder

eingelegten Originaldiskette weitere Files.

Der Diskettenwechsel entfällt, wenn man auf die RAM-Disk oder auf Laufwerk B kopiert. Die hier beschriebene Arbeitsweise gilt immer dann, wenn eine komplette Diskette kopiert werden soll. Unter Verwendung von File-Namen oder Wildcards können auch einzelne Programme oder Files kopiert werden. "Copyfile" stellt eine wertvolle Hilfe bei der täglichen Arbeit dar, da das Anfertigen von Sicherheitskopien bisher nicht so einfach war.

Gleiches gilt für den letzten Programmteil des Disc-Mate-Systems, den Directory-Editor. Nach dem Laden wird der Anwender aufgefordert, die zu bearbeitende Diskette einzulegen. Der Editor liest das Disketten-Inhaltsverzeichnis und gibt es unter dem Hauptmenü auf dem Bildschirm aus. Dabei erhält jedes aufgelistete Programm eine laufende Nummer, um die spätere Bearbeitung zu erleichtern. Folgende Menüpunkte stehen zur Verfügung:

1. SET TO R/O. Wird ein File mit R/O versehen, kann es nur noch geladen, von der Diskette aber nicht mehr gelöscht werden.
2. SET TO R/W. Das Gegenstück zu R/O. Ein File kann geladen und gelöscht werden.
3. SET TO SYS. Mit dieser Kennung versehen, erscheint ein File nicht mehr im Directory, es kann aber normal geladen werden.
4. SET TO DIR. Hebt Option 3 wieder auf.
5. ERASE A FILE. Ein File wird von der Diskette gelöscht.
6. UNERASE. Macht Option 5 rückgängig.
7. CHANGE DISC. Erlaubt den Diskettenwechsel, um ein anderes Directory zu bearbeiten.

8. HELP. Holt eine Help-Seite mit Kurzinformationen zu den einzelnen Optionen auf den Bildschirm.
9. QUIT. Löscht den Directory-Editor und kehrt zu CP/M zurück.
10. SAVE CHANGES. Mit dieser Option werden die Änderungen, die man vorgenommen hat, auf Diskette abgespeichert. Versäumt man diesen Vorgang, war die ganze Arbeit umsonst.
11. NEXT PAGE. Wird nur benötigt, wenn das Inhaltsverzeichnis einer Diskette eine Bildschirmseite überschreitet.

```
CP/M Plus Amstrad Consumer Electronics plc
v 1.4, 61K TPA, 1 Laufwerk, 112K Laufwerk M:

A)dir
A) ZIPDISC COM : COPYFILE COM : DMATE COM
A)
A)
A)zipdisc
SIREN SOFTWARE Zipdisc (C) 1986
*** zipdisc installed ***
A)
A)copyfile
SIREN SOFTWARE CP/M file copy program ver 1.0 (C) 1986
written by S.R. Mot and M.S. Lee

Command line format is : COPYFILE SOURCE DESTINATION
Where:- SOURCE = drive:filename.ext
and DESTINATION = drive:
(drive: defaults to current drive)
(and filename.ext defaults to *.**)
*** NO TABS AT ANY INPUT ***

INSERT SOURCE DISK IN DRIVE A: and PRESS ENTER
```

Installation von Zipdisc und Start Copyfile

Neben den hier beschriebenen Optionen bietet der Editor noch einige Informationen am Rande. So wird z.B. der freie Speicherplatz der Diskette genannt. Außerdem handelt es sich bei dem Directory um eine erweiterte Version, die neben Namen und Art des Programms auch noch die aktuelle User-Nummer und den Status ausgibt (R/O - R/W - Sys).

Die Utility-Sammlung "Disc Mate" ist meiner Meinung nach eine Bereicherung für alle Joyce-Anwender. Ich bin zwar fast sicher, daß im Laufe der Zeit andere, komfortablere Programme dieser Art auf den Markt kommen werden, bis dahin ist "Disc Mate" jedoch unverzichtbar.

System: Joyce PCW 8256/8512
 Hersteller: Siren Software
 Bezugsquelle: PR8-Softwaredienst
 Rolf Knorre

Hauptmenü des Directory-Editors und das Directory der momentan eingelegten Diskette. Alle Attribute werden angezeigt, ähnlich dem STAT-Befehl unter CP/M.

```

  (A) Set To R/O  (B) Set To DIR  (C) Change Disc  (S) Save Changes
  (D) Set To R/W  (E) Erase A File  (H) !!! HELP !!!  (P) Next Page
  (G) Set To SYS  (F) Unerase A File  (I) Quit To CP/M+

  MATCHPOI.  USER ** READ/WRITE DIR  MATCHPOI.  USER ** READ/WRITE DIR
  MATCHPOI.  USER ** READ/WRITE DIR  MATCHDAY.    USER ** READ/WRITE DIR
  MATCHDAY.  USER ** READ/WRITE DIR  MATCHDAY.  USER ** READ/WRITE DIR
  KNIGHTLO.  USER ** READ/WRITE DIR  KNIGHTLO.  USER ** READ/WRITE DIR
  KNIGHTLO.  USER ** READ/WRITE DIR  CYRUS      .BAS  USER ** READ/WRITE DIR
  CHESS      USER ** READ/WRITE DIR  CHESS      USER ** READ/WRITE DIR
  CHESS      USER ** READ/WRITE DIR

```

CP/M – die neue Perspektive (Teil 11)

Wir erstellen weitere Programme in CP/M 2.2

Im letzten Heft haben wir die ersten Programme unter CP/M geschrieben und einige BDOS-Funktionen zur Ausgabe von Zeichen auf Bildschirm und Drucker kennengelernt. Heute geht es mit der Eingabe von Zeichen über die Tastatur sowie mit den ersten Diskettenfunktionen weiter.

Beginnen wir in der heutigen Folge mit einem kleinen Rückblick, um den roten Faden wieder aufzunehmen. Wir haben 3 BDOS-Funktionen zur Ausgabe von Zeichen kennengelernt: BDOS 02H (CONOUT) gab ein Zeichen und BDOS 09H (STROUT) eine Zeichenkette ab der aktuellen Cursorposition auf dem Bildschirm aus. Zusätzlich lernten wir die BDOS-Funktion 05H (LIST) kennen, die ein Zeichen zum Drucker sendete. Diese sind die wesentlichen Ausgabefunktionen für Zeichen.

Neben der Ausgabe spielt die Eingabe von Zeichen über die Tastatur für interaktive Programme eine entscheidende Rolle. CP/M stellt dafür im wesentlichen 3 BDOS-Funktionen zur Verfügung, die im folgenden beschrieben werden sollen. Neben der Eingabe von Zeichen (-ketten) ist ihre programmbedingte Interpretation (was für Zeichen, sind sie zulässig etc.) von großer Bedeutung, so daß hierauf besonders hingewiesen wird.

Eingabe eines Zeichens

Die zentrale Eingabe-Funktion ist die BDOS-Funktion "console input" (CONIN) mit der Funktionsnummer 01H. Sie holt ein Zeichen von der Tastatur und übergibt es im Akkumulator an das laufende Programm (ASCII-Zeichenwert). Diese Funktion wartet auf die Eingabe eines Zeichens von der Tastatur. Danach wird es gleichzeitig an der aktuellen Cursorposition auf dem Bildschirm ausgegeben. Dieses Verhalten wird unter CP/M als Echo bezeichnet. Die Funktionsnummer 01H wird wie üblich ins C-Register gelegt, und der ASCII-Wert des übergebenen Zeichens befindet sich nach Beendigung der Funktion im Akkumulator. Da diese Funktion auf die Eingabe wartet, kann sie mit den Basic-Anweisungen

```
10 a$=INKEY$
20 IF a$<>" " THEN PRINT a$ ELSE
   GOTO 10
```

verglichen werden. Die Anwendung dieser Funktion liegt im besonderen in der Auswahl aus einem Menü (über ein bestimmtes Zeichen einen bestimmten Programmteil wählen), bei der Abfrage, ob z.B. eine Eingabe richtig war (J/N), und der Eingabe von Informationen mit einer Länge von einem Zeichen. Hier ist schon zu erkennen, daß es neben der reinen Eingabe auch auf die Interpretationen des Zeichens ankommt. Sehen Sie sich dazu das Programm "PROG06" genauer an. Es wartet auf die Eingabe eines bestimmten Zeichens (hier Carriage Return, 0DH). Die Compare-Anweisung (010DH) vergleicht den Akkumulatorinhalt (Zeichen der Eingabefunktion) mit 0DH. War es ein Carriage Return (Zero-Flag gesetzt), wird das Programm beendet. Im anderen Fall erfolgt eine Meldung auf dem Bildschirm, und die Eingabe

muß solange wiederholt werden, bis die ENTER-Taste betätigt wurde.

Bei manchen Anwendungen ist es nicht notwendig, auf ein bestimmtes Zeichen zu warten, sondern vielmehr nur auf eine beliebige Tastenbetätigung (z.B. "Weiter ... Bitte eine Taste"). Auch stört in diesem Fall das Echo des Zeichens auf dem Bildschirm. Die BDOS-Funktion 0BH ("get console status") fragt nur, ob ein Zeichen an der Console ansteht, nicht aber welches. Stand zum Zeitpunkt der Abfrage ein Zeichen an, so wird im Akkumulator 0FFH übergeben, im anderen Fall beinhaltet der Akkumulator den Wert 00H. Mit Hilfe dieser Funktion soll das folgende kleine Basic-Programm in ein CP/M-Programm umgeschrieben werden:

```
10 PRINT "Waiting ..."
20 a$=INKEY$: IF a$="" THEN 20
30 END
```

PROG06

```
*****[PROG06]**
*
*      CONSOLE INPUT (CONIN)      *
*      Zeichen v. d. Tastatur holen *
*
*****
;
0005 =      BDOS      EQU      0005H
0100 =      TPA      EQU      0100H
0001 =      CONIN     EQU      01H
0009 =      STROUT    EQU      09H
;
0100      ;
;
0100 111E01 LABEL1 LXI  D,TEXT1
0103 0E09      MVI  C,STROUT
0105 CD0500    CALL  BDOS
;
0108 0E01      MVI  C,CONIN; [C] Funktionnr.
010A CD0500    CALL  BDOS; warte auf Zeichen
;
010D FE0D      CPI   0DH; Vergleiche [AC]
010F CA1D01    JZ   ENDE; Richtig? Ende
0112 113B01    LXI  D,TEXT2
0115 0E09      MVI  C,STROUT
0117 CD0500    CALL  BDOS
011A C30001    JMP  LABEL1
;
; ENDE
011D C9      RET
;
011E 4269747465TEXT1 DB 'Bitte ein RETURN betaetigen $'
013B 2046414C53TEXT2 DB ' FALSCH! Noch einmal!',0DH,0AH,'$'
0153      END
```

Da solche Routinen im weiteren Verlauf dieses Kurses des öfteren benötigt werden, sollten sie als Unterprogramm ausgelegt sein. Das Anlegen einer Unterprogramm-bibliothek bringt gerade bei Routinen, die man in sehr vielen Programmen benötigt, einen großen Vorteil: Sie müssen nur einmal geschrieben werden! Der Quelltext eines Unterprogramms läßt sich dann jederzeit in ein Programm zuladen. Sehen Sie sich bitte das Programm "PROG07" an, das sein Unterprogramm mit der Aufgabe, auf eine Taste zu warten, aufruft. Das Unterprogramm selbst legen Sie bitte zusätzlich als ASCII-File auf der Arbeitsdiskette ab. Alle benutzten Register werden zu Beginn des Unterprogramms auf dem Stapel gerettet und am Ende wieder zurückgeholt. Dadurch ist gewährleistet, daß alle Register mit dem Inhalt aus dem Unterprogramm zurückkehren, mit dem sie aufgerufen wurden.

Eingabe einer ganzen Zeile

Bei vielen Anwendungen ist neben der Eingabe eines Zeichens die einer ganzen Zeichenfolge notwendig. Eine Realisierung dieses Problems wäre z.B. über eine Schleife möglich, in der mehrere einzelne Zeichen (BDOS 01H) eingelesen und gespeichert werden. Der Aufwand steigert sich aber um ein Vielfaches, wenn die eingegebenen Zeichen auch noch editiert werden sollen (Löschen eines Zeichens, Löschen der gesamten Eingabe etc.). Doch auch hier bietet CP/M für die Programmierung eine komplette Lösung an.

Die BDOS-Funktion "read console buffer" (BDOS 0AH) ist eine Routine, die die Eingabe einer kompletten Zeile über die Tastatur erlaubt (für die Kommandoeingabe wird diese Funktion ebenfalls benutzt). Es lassen sich bis zu 255 Zeichen auf einmal einlesen, die in einem Zwischenspeicher bei der Eingabe abgelegt werden. Dieser Zwischenspeicher (Puffer) wird vom Programmierer in der TPA angelegt und reserviert. Weiterhin muß angegeben werden, wie viele Zeichen maximal einzugeben sind. Die Eingabe wird abgeschlossen, wenn entweder die ENTER-Taste (auch ^M) betätigt wird oder aber die angegebene maximale Anzahl von Zeichen erreicht ist. Dieses ist sehr nützlich, wenn sichergestellt sein soll, daß z.B. nur 4 Zeichen eingegeben werden dürfen. Eine weitere Besonderheit ist, daß die Eingabezeile mit einigen Steuerzeichen editiert werden kann (siehe weiter vorne).

Der Puffer für die Zeichen muß einen ganz bestimmten Aufbau haben und vom Programmierer selbst angelegt werden:

```

PROG07
;*****[PROG07]**
;*
;* GET CONSOLE STATUS (CONSTAT) *
;*
;* Zeichen v. d. Tastatur holen *
;*
;*****
;
0005 = BDOS EQU 0005H
0100 = TPA EQU 0100H
0009 = STROUT EQU 09H
000B = CONSTAT EQU 0BH
;
0100 ; ORG TPA
;
0100 CD0401 ; CALL WAIT
;
0103 C9 ; RET
;
;*****
;* Wartet auf eine Tastedruckt *
;*****
0104 F5 WAIT PUSH PSW; Alle Register
0105 C5 PUSH B; retten
0106 D5 PUSH D
0107 111D01 LXI D,WTEXT
010A 0E09 MVI C,STROUT
010C CD0500 CALL BDOS
010F 0E0B WAIT1 MVI C,CONSTAT; Status
0111 CD0500 CALL BDOS; abfragen
0114 FE00 CPI 00H; [A] = 00H?
0116 CA0F01 JZ WAIT1; Ja, warten
0119 D1 POP D; Alle Register
011A C1 POP B; wieder holen
011B F1 POP PSW
011C C9 RET ; WAIT-Ende
;
011D 5741495449WTEXT DB 'WAITING ...$'
;
0129 ; END

```

1. Byte: Maximale Eingabelänge der Zeile.
2. Byte: Zeigt nach Eingabe der Zeile an, wie viele Zeichen tatsächlich eingegeben wurden (0 bis maximale Anzahl).
3. Byte: Beginn des eigentlichen Puffers. Ab hier sind mindestens so viele Bytes zu reservieren, wie maximal aufgenommen werden dürfen (siehe 1. Byte).

Beispielsweise will man eine Zeile mit maximal 32 Zeichen eingeben. Dann muß das erste Byte des Puffers mit dem Wert 32 (max. Anzahl) geladen werden. Der Zeichenpuffer ab dem 3. Byte muß dann mindestens 32 Zeichen aufnehmen können, d.h., hier sind 32 Bytes zu reservieren! Für das Reservieren eines Speicherbereichs steht die Pseudo-Anweisung DS zur Verfügung. Nach dem Abschluß der Eingabe zeigt das 2. Byte an, wie viele Zeichen eingegeben wurden. Nach der Eingabe von "HALLO" steht dort der Wert 5. Mit Hilfe dieses Wertes können dann z.B. alle gültigen Zeichen wieder ausgelesen werden (Schleife, siehe PROG08). Vor dem Aufruf der Funktion (C-Register = 0AH) ist das DE-Register mit der Pufferadresse zu laden:

```

LXI D,PUFFER
MVI C,0AH
CALL BDOS

```

Der Puffer kann an einer beliebigen Stelle der TPA folgendermaßen angelegt sein:

```

PUFFER DB 20H ; 32 Zeichen max.
DS 01H ; 1 Byte für Anzahl
DS 20H ; Zeichenpuffer

```

Während der Eingabe kann die aktuelle Eingabezeile mit einigen Steuerzeichen bearbeitet werden. Hier die wichtigsten:

```

DEL : Löscht letztes Zeichen
^X : Löscht gesamte Zeile
ENTER: Eingabeende

```

Alle Steuerzeichen wurden in Folge 2 bereits beschrieben.

"PROG08" zeigt ein Anwendungsbeispiel. Nach der Eingabe (entweder mit ENTER oder Pufferüberlauf) werden alle Zeichen nacheinander ausgelesen und noch einmal auf dem Bildschirm angezeigt. Es wurde ein weiteres Unterprogramm für die Ausgabe eines Zeichens im E-Register erstellt, das Sie in ihre Unterprogramm-bibliothek aufnehmen sollten. Darin werden alle Register gerettet, da die BDOS-Funktion 02H alle Inhalte der Register zerstören kann.

Mit den Funktionen zur Ein- und Ausgabe von Zeichen unter CP/M wurde der Grundstein für die Programmierung unter CP/M gelegt. Schon bei diesen Routinen, die nur sehr einfache Funktionen ausführen, sind die wichtigsten Vorteile klar zum Ausdruck gekommen: Universelle Programmierung der wesentlichen Ein- und Ausgabe-Funktionen ohne Kenntnis der rechner-spezifischen Hardware! Sie sollten mit den bisher beschriebenen Funktionen Unterprogramme zu folgenden CPC-eigenen Routinen schreiben:

- Bildschirm löschen (CLRSCR)
- Mode wählen (20, 40, 80 Zeichen)
- Farbe wählen
- Meldungen ausgeben

Alle diese Unterprogramme können Sie mit BDOS 02H und BDOS 09H erreichen, indem Sie die CPC-eigenen Steuerzeichen "ausprinten". (Diese Steuerzeichen wurden bereits im 2. Teil dieser Serie beschrieben!)

Versuchen Sie sich auch einmal an einem universellen Unterprogramm für eine Eingabezeile.

An dieser Stelle sind alle wichtigen Ein- und Ausgabe-Funktionen für den Bildschirm und die Tastatur beschrieben. Sie spielen für die weitere Programmierung unter CP/M eine entscheidende Rolle. Erarbeiten Sie sich die Funktionen daher vollständig.

Systeminformationen

Eine besondere Funktion ist die BDOS-Funktion "get versions number" (BDOS 0CH). Sie übergibt die Information, mit welcher Version des Betriebssystems Sie gerade arbeiten. Nach dem Aufruf der Funktion (C-Register = 0CH) erhalten Sie folgende Informationen zurück:

H-Register : Welches Betriebssystem
 00H : CP/M
 01H : MP/M

L-Register : Versionsnummer
 00H : Version vor 2.0
 20H : Version 2.0
 21H : Version 2.1
 22H : Version 2.2

 2FH : Version 2.15

Bei Ihrem CP/M müssten Sie die Information CP/M 2.2, also H-Register = 00H und L-Register = 22H erhalten. Ein Wort noch zum Betriebssystem MP/M ("multi processing monitor control program"). MP/M ist ein Multi-Task-Betriebssystem auf der Grundlage von CP/M. Es kann mehrere Prozesse "gleichzeitig" verarbeiten, so daß entweder mehrere Terminals an einen Computer angeschlossen sein können oder aber sich

mehrere Programme "parallel" verarbeiten lassen. Schreiben Sie ein kleines (Unter-)Programm, das Ihnen diese Informationen auf dem Bildschirm ausgibt (z.B. "CP/M 2.2").

Die Diskettenlaufwerke

Neben den bisher behandelten Funktionen zur Ein- und Ausgabe von Informationen (Tastatur und Bildschirm) übernimmt das CP/M im wesentlichen die Steuerung der angeschlossenen Diskettenlaufwerke. Der Großteil aller

BDOS-Funktionen bezieht sich deshalb auch auf die angeschlossenen Laufwerke. Obwohl CP/M 2.2 grundsätzlich die Verwaltung von bis zu 16 Laufwerken (A bis P) unterstützen kann, sind es beim CPC unter CP/M nur 2 Laufwerke (A und B). Der Grund dafür ist nicht beim CP/M, sondern bei der Hardware der CPC-Computer zu suchen. Die Funktionen zur Behandlung der Laufwerke sind sehr weitreichend und komfortabel. Neben der Verwaltung von Daten und Programmen (Directory, Löschen, Umbe-

PROG08

```

;*****[PROG08]**
;*
;*   READ CONSOLE BUFFER (BUFIN)
;*
;*   Zeichenkette einlesen
;*
;*****
;
0005 = BDOS      EQU      0005H
0100 = TPA      EQU      0100H
0002 = CONOUT   EQU      02H
0009 = STROUT   EQU      09H
000A = BUFIN    EQU      0AH

0100
;
;
0100 113901      LXI      D,TEXT
0103 0E09        MVI      C,STROUT
0105 CD0500      CALL     BDOS

;
0108 3E20        MVI      A,20H; Max. Eingabelaeenge
010A 324E01      STA      PUFFER

;
;
010D 114E01      LXI      D,PUFFER; Adresse des Puffers
0110 0E0A        MVI      C,BUFIN; Puffer einlesen
0112 CD0500      CALL     BDOS

;
0115 114B01      LXI      D,CRLF; Zeilenvorschub
0118 0E09        MVI      C,STROUT; ausfuehren
011A CD0500      CALL     BDOS

;
011D 214F01      LXI      H,ANZAHL
0120 46          MOV      B,M
0121 23          LABEL1  INX
0122 5E          MOV      E,M
0123 CD2B01      CALL     AUSGABE
0126 05          DCR      B
0127 C22101      JNZ      LABEL1
012A C9          RET

;
;*****
;*   Zeichen [E] ausgeben
;*
;*****
012B F5          AUSGABE  PUSH     PSW
012C C5          PUSH     B
012D D5          PUSH     D
012E E5          PUSH     H
012F 0E02        MVI      C,CONOUT
0131 CD0500      CALL     BDOS
0134 E1          POP      H
0135 D1          POP      D
0136 C1          POP      B
0137 F1          POP      PSW
0138 C9          RET      ; AUSGABE-Ende

;
0139 4E616D656E TEXT    DB 'Namen eingeben : $'
014B 0D0A24      CRLF    DB 0DH,0AH,'$'
014E 00          PUFFER  DB 00H
014F 00          ANZAHL  DB 00H
0150             INHALT  DS 40H

0190             END

```

nennen etc.) sind vor allem Funktionen zur Erstellung und Bearbeitung von Dateien zu finden. Besonders interessant ist dabei, daß sowohl sequentielle als auch Random-Dateien möglich sind. Zunächst sollen aber die Funktionen behandelt werden, die allgemeine Informationen über die angeschlossenen Laufwerke liefern.

Gerade bei Anwenderprogrammen ist die Information wichtig, über wie viele

- L-Register : Bit 0 = Laufwerk A
- : Bit 1 = Laufwerk B
-
- : Bit 7 = Laufwerk H
- H-Register : Bit 0 = Laufwerk I
- : Bit 7 = Laufwerk P

Anhand dieser Information ist die Entscheidung nicht mehr schwer. Beim CPC liefert das HL-Register die Werte 1 (nur A) oder 3 (A und B).

mierer wissen, auf welcher User-Ebene (Benutzerebene) der Diskette er sich befindet. Auch das kann er wieder über eine BDOS-Funktion erfahren. Die Funktion "get/set user" (BDOS 20H, C-Register = 20H) erfüllt dabei zwei Aufgaben. Einmal ermittelt GET die aktuelle User-Ebene, und zum anderen setzt SET die User-Ebene neu. Welche Aufgabe die Funktion erfüllen soll, entscheidet sich durch den Inhalt des E-Registers. Wird dieses auf den Wert OFFH gesetzt, so wird die aktuelle User-Ebene ausgelesen (GET) und ihre Nummer (00H bis 0FH) im Akkumulator zurückgeliefert. Enthält das E-Register einen anderen Wert, so wird dieser als neue User-Nummer interpretiert und das System darauf umgeschaltet (00 bis 0FH). Schreiben Sie auch hierzu ein kleines Unterprogramm.

Abschließend soll noch die Funktion "select disc" (BDOS 0EH, C-Register = 0EH) behandelt werden. Sie wählt eines der angeschlossenen Laufwerke als neues aktuelles Laufwerk aus. Welches gewünscht wird, ist abhängig vom Inhalt des E-Registers:

- E-Register : 00H = Laufwerk A
- : 01H = Laufwerk B
- :
- : 0FH = Laufwerk P

Wird ein Laufwerk angegeben, das nicht angeschlossen ist, erfolgt die Fehlermeldung "Disc Missing". Das ausgewählte Laufwerk bleibt bis zu einem Warmstart oder einem erneuten Umschalten das aktuelle Laufwerk. Um diese Funktion einwandfrei zu benutzen, d.h. einen Absturz zu vermeiden, sollte zunächst mit der BDOS-Funktion 18H (Ermittlung der aktiven Laufwerke) festgestellt werden, ob das gewünschte Laufwerk überhaupt vorhanden ist oder nicht (in diesem Falle Umschaltung verhindern).

So, für heute soll es erst einmal genügen. Versuchen Sie, alle Funktionen anhand kleiner Beispiele komplett zu erarbeiten und zu verstehen. In der nächsten Folge werden dann die restlichen "einfachen" Funktionen für die Laufwerke behandelt, und dann gehen wir langsam in die Vollen.

Manfred Walter Thoma

PROG09

```

;*****[PROG09]**
;*
;* RETURN CURRENT DISC (CURRENT) *
;*
;*   Aktuelles Laufwerk angeben *
;*
;*****
;
0005 = BDOS      EQU      0005H
0100 = TPA      EQU      0100H
0002 = CONOUT   EQU      02H
0009 = STROUT   EQU      09H
0019 = CURRENT  EQU      19H
;
0100          ;          ORG      TPA
;
0100 111601   LXI      D,TEXT
0103 0E09     MVI      C,STROUT
0105 CD0500   CALL     BDOS
;
0108 0E19     MVI      C,CURRENT; Laufwerknummer
010A CD0500   CALL     BDOS; nach [A] holen
010D C641     ADI      41H; [A] nach ASCII wandeln
010F 5F       MOV      E,A
0110 0E02     MVI      C,CONOUT; und ausgeben
0112 CD0500   CALL     BDOS
0115 C9       RET
;
0116 4C61756677TEXT DB 'Laufwerk : $'
0122          END

```

angeschlossene Laufwerke das System eigentlich verfügt und welche es sind. Wenn Daten zum Laufwerk B gesendet werden, muß natürlich sichergestellt sein, daß Laufwerk B auch vorhanden ist. Beim Schneider CPC ist diese Frage noch sehr einfach zu beantworten: Entweder Laufwerk A oder A und B (B allein geht nicht). Bei größeren Systemen ist diese Frage schon wesentlich diffiziler.

Die BDOS-Funktion "get login vector" (BDOS 18H, C-Register = 18H) hilft einem hier weiter. Sie übergibt eine 16-Bit-Information (HL-Register), wobei jedes einzelne Bit den Zustand eines der möglichen 16 Laufwerke interpretiert. Ist das Bit gesetzt (1), so ist das entsprechende Laufwerk aktiv. Ein gelöscht Bit (0) kennzeichnet ein inaktives Laufwerk.

Doch welches der angeschlossenen Laufwerke ist das aktuelle, auf das sich z.Z. alle Ein- und Ausgaben beziehen? Auch hierauf weiß CP/M eine Antwort. Die BDOS-Funktion "return current disc" (BDOS 19H, C-Register = 19H) liefert im Akkumulator einen Wert zwischen 0 und 15, der das momentan aktuelle Laufwerk kennzeichnet:

- Akkumulator : 00H = Laufwerk A
- : 01H = Laufwerk B
- :
- : 0FH = Laufwerk P

Das Beispielprogramm "PROG09" liest das aktuelle Laufwerk und gibt es auf dem Bildschirm aus.

Neben der Information, welches das aktuelle Laufwerk ist, muß der Program-

**Ihr direkter Draht
zur Redaktion:
☎ 07252/42948**

Pascal-Kurs (Teil 10)

In unserer letzten Folge beschäftigen wir uns mit dynamischen Variablen, die bei Pascal eine große Bedeutung haben

Zum Abschluß unseres Pascal-Kurses wollen wir uns mit den dynamischen Variablen beschäftigen. Diese Datenstrukturen stellen in der Hand des Programmierers ein mächtiges Werkzeug dar. Ihre Anwendung ist allerdings nicht immer problemlos. Aus diesem Grund soll in dieser Folge die Grundlage für die Verwendung geschaffen werden.

Grundlagen

Im Gegensatz zu den normalen Variablen, die statisch sind, d.h. zu Programmbeginn festliegend und während der Programmausführung nicht änderbar, erlauben dynamische Variablen, die erst während des Programmablaufs erzeugt werden, sehr flexible Datenstrukturen. Insbesondere Listen und Bäume, die in der Informatik sehr beliebt sind, lassen sich mit den bisher beschriebenen Datenstrukturen, die auch unter Basic verfügbar sind, nur schwer nachbilden.

Der Aufbau dynamischer Variablen geschieht über sogenannte Zeigervariablen. Dies sind statisch erzeugte Variablen, die dazu dienen, während des Programmablaufs auf eine bestimmte Stelle im Speicher zu zeigen. An dieser Stelle können dann beliebige Datenstrukturen aufgebaut werden. In besonderer Weise eignet sich hierzu die Record-Struktur, da sich in dieser beliebige skalare Datentypen zusammenfassen lassen. Dynamische Variablen dürfen alle vorhandenen Datentypen enthalten (bis auf den Typ File). Insbesondere darf auch der Typ der Zeigervariablen in einem Record enthalten sein. Auf diese Weise kann innerhalb eines spezifischen Datentyps auf einen anderen Datentyp mit ebenfalls dynamischer Struktur verwiesen werden. Diese Möglichkeit bildet die Grundlage für den Aufbau von Listen und Bäumen. Bei diesen muß immer ein Bezug zum Nachfolger und/oder zum Vorgänger vorhanden sein.

Gleich zu Anfang der Hinweise auf Gefahren bei der Verwendung von Zeigern (Pointern). Die Zeigervariablen sind statischer Natur, werden also innerhalb der anderen statischen Variablen angelegt. Da das Turbo-Pascal-System seine Variablen zu Programmbeginn nicht initialisiert, stehen an diesen Stellen willkürliche Werte. Benutzt man also diese Zei-

gervariablen, ohne ihnen vorher einen gültigen Wert zuzuweisen, kann man, je nachdem welcher Wert an dieser Stelle zufällig stand, durchaus im Programmtext oder sogar im Betriebssystem Werte verändern. Das hat natürlich einen Systemabsturz zur Folge – also Vorsicht!

Vom Umgang mit Pointern

Für den Umgang mit Pointern bietet Pascal einige Standardprozeduren an. Bevor wir uns mit ihnen beschäftigen, wollen wir uns zunächst anschauen, wie eine dynamische Variable im Speicher angelegt wird. Der Beschreibung in Bild 1 liegt folgende Typen- und Variablen-deklaration zugrunde:

```

TYPE
    zeiger = ^eintrag;
    eintrag = RECORD
        inhalt : string (.40);
    END;
VAR
    q : zeiger;

```

Der linke Teil von Bild 1 zeigt die Situation im Speicher nach der Compilierung und bei Programmbeginn. Innerhalb der normalen statischen Variablen ist ein Platz für die Zeigervariable q reserviert. Der Wert für q wird nicht festgelegt und entspricht dem Wert, der zufällig an dieser Stelle steht. Erst bei der Verwendung der Standardprozedur New wird im freien Speicherbereich, der durch die Systemvariable Heap gekennzeichnet ist, eine dynamische Variable des Typs angelegt, der durch die Typendeklaration definiert wurde. Die Adresse dieser Variablen wird in der Variablen q gespeichert. Ein Zugriff ist nur über die Zeigervariable möglich. Deshalb ist die Zuweisung, wie im rechten Teil von Bild 1 beschrieben, notwendig, um der dynamischen Variablen einen Wert zuweisen zu können. Die oben gezeigte Typendeklaration ist natürlich recht einfach gewählt, aber um das Prinzip zu verdeutlichen, ist diese Struktur ausreichend. Bei der Verwendung komplizierter Strukturen ist das Prinzip das gleiche, ein Unterschied besteht lediglich im verbrauchten Speicherplatz.

Standardprozedur New

Die Standardprozedur New haben wir ja bereits in der obigen Beschreibung

kennengelernt. Bevor man eine dynamische Variable verwenden kann, muß mit New Speicherplatz für sie reserviert werden. Alle diese Variablen werden auf einem "Haufen" abgelegt, liegen also im Speicher übereinander. Aus diesem Grunde kann man nur über die Pointer auf die Variablen zugreifen. Dies hat aber noch zur Konsequenz, daß durch die Verwendung von New einer bestimmten Variablen eine Adresse zugewiesen wird. An dieser und den folgenden Adressen werden dann die gewünschten Werte gespeichert. Wendet man dann auf dieselbe Zeigervariable noch einmal New an, so wird eine weitere Variable angelegt und die Adresse in der Zeigervariablen gespeichert. Dadurch wird die alte Adresse überschrieben und kann nicht mehr restauriert werden. Die zu ihr gehörende Variable kann also nicht mehr angesprochen werden, und auch der Speicherplatz ist nicht mehr frei, da die Variable nicht ausdrücklich gelöscht wurde. Sie fristet nun als Speicherleiche ihr Dasein. Man kann sicherlich der Ansicht sein, daß das nicht so schlimm ist; aber man stelle sich die Situation vor, eine solche Zuweisung befindet sich in einer Schleife, die vielleicht noch mehrere tausend Mal durchlaufen wird. Den erzielten Effekt kann sich wahrscheinlich jeder leicht ausmalen.

Standardprozedur Dispose

Das Gegenstück zu New ist die Standardprozedur Dispose. Damit werden zuvor mit New erzeugte Variablen wieder aus dem Speicher entfernt. Der dabei entstehende freie Speicherplatz wird registriert und bei einem erneuten New verwendet, sofern er groß genug ist, um die neue Variable aufzunehmen. Nach der Verwendung von Dispose kann man natürlich nicht mehr auf die Variable zugreifen.

Standardprozeduren Mark und Release

Diese beiden Prozeduren sind eine Besonderheit des Turbo-Pascal-Systems und sind im Standard-Pascal nicht enthalten. Mark und Release stellen einen Ersatz für Dispose dar, sind in ihrer Anwendung allerdings verschieden. Während bei Dispose nur ein bestimmtes

Element auf dem Heap gelöscht wird, werden bei Mark und Release alle Variablen nach der durch Mark gekennzeichneten Variablen gelöscht. Die Heap-Verwaltung bei Mark/Release und Dispose ist unterschiedlich. Deshalb darf innerhalb eines Programms nur eine der beiden Arten vorkommen.

GetMem und FreeMem

Die Prozeduren GetMem und FreeMem erlauben dem Programmierer, sich auf dem Heap einen bestimmten Platz zu reservieren bzw. diesen wieder freizugeben. Im Gegensatz zu New, wo genausoviel Speicherplatz reserviert wird wie für die Datenstruktur benötigt, kann man hier direkt die Anzahl von Bytes angeben, die reserviert werden soll. Dazu wird die Prozedur GetMem verwendet, die mit zwei Parametern aufgerufen wird. Der erste ist eine beliebige Zeigervariable, der zweite ein Integer-Ausdruck, der die Anzahl der Bytes angibt. Um den verwendeten Speicherplatz wieder freizugeben, wird FreeMem verwendet. Der Aufruf erfolgt genau wie bei GetMem. Wichtig ist, daß der Integer-Ausdruck bei FreeMem mit dem bei GetMem exakt übereinstimmt, da sonst eventuell zuviel oder zuwenig gelöscht wird.

MemAvail

Die Standardfunktion MemAvail gibt den Umfang des größten zusammenhängenden freien Speicherbereichs an. Ist die Anzahl der Bytes größer als 32767, so wird eine negative Zahl ausgegeben. Die korrekte Anzahl erhält man durch Addition von 65536.0 zu MemAvail. Bei intensivem Gebrauch von dynamischen Variablen sollte man sich vor ihrer Neuanlage versichern, ob noch genügend Platz vorhanden ist.

Zuweisungen und Relationen

Eine Zuweisung zwischen Zeigern ist erlaubt, solange beide Zeiger vom gleichen Typ sind. Sie bewirkt, daß zwei Zeiger auf die gleiche Variable gerichtet sind. Die Variable selbst existiert nur einmal, läßt sich aber über zwei Zeiger ansprechen. Vorsicht ist geboten, wenn beide Zeiger bereits auf eine Variable deuten. In diesem Fall wird durch eine Zuweisung der Zugriff auf eine der beiden Variablen unmöglich gemacht. Als Relationen zwischen Zeigern sind "=" und "<>" erlaubt, sofern die Zeiger vom gleichen Typ sind. Das Ergebnis ist ein boolescher Wert. Um zu kennzeichnen, daß ein Zeiger (noch) auf keine Variable weist, wird die Funktion nil eingesetzt. Sie ist zu jedem beliebigen Zeigertyp kompatibel. Da man diese Funktion auch in Vergleichen einsetzen darf, kann man damit bestimmte Konstellationen

wie z.B. das Ende einer Liste kennzeichnen.

Anwendungen

Anwendungen für dynamische Variablen gibt es viele. Durch den Vorteil, daß der eingesetzte Speicherplatz nach Erledigung der Aufgabe wieder freigegeben wird, lassen sich in einem Programm Funktionen realisieren, die bei Verwendung statischer Variablen nicht alle ins Programm passen würden. Auch beim Kopieren von Dateien kann über die dynamischen Variablen jeweils der maximal mögliche freie Speicherplatz als Puffer verwendet werden. Eine weitere Möglichkeit ist die Einrichtung eines Software-Druckerspoolers. Alle diese Anwendungen lassen sich auch über statische Variablen realisieren, beanspruchen dann aber einen bestimmten Speicherplatz, der nicht für verschiedene Funktionen gemeinsam genutzt werden kann. Besonders beim Kopieren von Dateien muß der verwendete Puffer eine gewisse Mindestgröße haben, damit der Kopiervorgang nicht zu lange dauert. Das Kopieren selbst ist aber sicherlich keine häufig verwendete Funktion, so daß der verwendete Speicherplatz besonders schmerzt. Im Gegensatz zu den bisher genannten Anwendungen ist die Realisierung verschiedener Datenstrukturen mit herkömmlichen Mitteln nicht oder nur mit erheblichem Aufwand möglich. Einfacher sind diese Strukturen mit dynamischen Variablen realisierbar.

Stacks

Eine dieser Datenstrukturen ist der Stack oder "Kellerspeicher". Wer sich ein bißchen mit Maschinensprache auskennt, der weiß, daß der Z80 ein Register mit dem Namen "Stackpointer" besitzt. Es dient zum Aufbau eines Stacks, auf dem Programmierer und Prozessor Werte sichern können. Seine Struktur ist so gestaltet, daß das Element, das als letztes auf den Stack kommt, als erstes wieder entfernt wird. Man spricht hier von Last-in First-out. Zur Verwaltung eines Stacks reicht ein Zeiger auf das oberste Element aus, denn es kann nur dieses Element entfernt oder ein neues obendrauf gesetzt werden. Der Z80-Prozessor kann also ohne großen Aufwand eine relativ komplexe Datenstruktur für seine Zwecke verwenden.

Ein Beispiel für einen Stack zeigt unser Programm "Stackdemo". Genau wie der für den Prozessor, wird hier ein Stack in Pascal angelegt. Während der Prozessorstack aber nur Worte (Zwei-Byte-Werte) speichert, kann unser Programm für eine beliebige Datenstruktur einen Stack aufbauen. In unserem Beispiel wird ein Stack für einen String von 20

Zeichen Länge eingerichtet. Will man die Stackstruktur ändern, so reicht es, die Record-Definition für stack zu ändern. Wie schon erwähnt, reicht für einen Stack ein Zeiger auf das oberste Element aus. Dieser Zeiger hat in unserem Programm den Namen Top. Er wird von den beiden Hauptprozeduren Einfuegen und Entfernen aktualisiert. Darüber hinaus benutzen wir noch eine boolesche Variable mit dem Namen leer, die anzeigt, ob sich ein Element auf dem Stack befindet oder nicht. Die verwendete Prozedur Ausgabe dient lediglich zur Ausgabe der Stackelemente und trägt nicht zur Funktion des Stacks bei. Die beiden wichtigsten Prozeduren sind Einfuegen und Entfernen. Sie leisten die eigentliche Stackverwaltung. Wie die Namen schon andeuten, sorgt Einfuegen für das Schieben eines Elements auf den Stack (auch push genannt), während Entfernen das oberste Element vom Stack entfernt (auch pop genannt). Der Prozedur Einfuegen wird als Stackinhalt ein String übergeben. Um ein neues Element auf den Stack zu bringen, benutzt die Prozedur eine Hilfsvariable mit dem Namen neu. Durch New (neu) wird eine neue Variable angelegt und der übergebene String in das Inhaltsfeld gebracht. Der im Record enthaltene Zeiger wird dann auf die bisherige Spitze gerichtet (neu^.naechster:=top), anschließend zeigt top auf die neue Variable (top:=neu). Da beim Einfuegen eines neuen Elements der Stack mit Sicherheit nicht mehr leer ist, wird leer auf false gesetzt. Durch einfaches "Verbiegen" einiger Zeiger läßt sich diese Struktur recht einfach ausbilden.

Das Entfernen von Elementen vom Stack funktioniert folgendermaßen. Sollte der Stack nicht leer sein (sonst kann man ja nichts entfernen), wird der Zeiger top einem Hilfszeiger zugewiesen (p), anschließend wird top auf den Nachfolger von p (und damit auf das vorletzte Element des Stacks) gesetzt. Damit der verwendete Speicherplatz wieder freigegeben wird, wird mittels Dispose die Variable p, die ja bisher als oberstes Element gespeichert war und nun nicht mehr notwendig ist, gelöscht. Der dabei entstehende freie Speicherplatz steht beim nächsten New wieder zur Verfügung.

Zu Beginn des Hauptprogramms wird der Stack durch die Zuweisungen top:=nil und leer:=true initialisiert. Die folgende WHILE-DO-Schleife dient nur zum Ausprobieren der Funktionen. Nach jeder Aktion wird der Stack neu ausgegeben.

Puffer

Eine weitere Datenstruktur, die eine gewisse Ähnlichkeit mit den Stacks hat,

ist der Pufferspeicher. Auch hier werden Elemente nacheinander abgelegt. Allerdings werden Werte, die als erste eingegeben werden, auch als erste ausgegeben. Man spricht hier von First-In First-out. Eine Struktur dieser Art befindet sich auch im Betriebssystem des Schneider-Computers. Die Tastatureingabe wird in einem Puffer gespeichert, bei dem die zuerst gedrückten Werte auch als erste wieder ausgegeben werden. Sollte ein Programm mit der Tastatureingabe nicht Schritt halten können, so werden in diesem Puffer die Werte zwischengespeichert und nach Anforderung durch das übergeordnete Programm wieder in der gleichen Reihenfolge ausgegeben. Eine weitere Anwendung findet diese Datenstruktur bei Terminalprogrammen. Hier kann es vorkommen, daß die empfangenen Daten schneller ankommen, als der Rechner sie anzeigen oder abspeichern kann. In diesem Fall werden die eingehenden Daten zwischengespeichert und verspätet ausgegeben. Macht der Sender dann eine Pause, kann der Puffer wieder geleert werden.

Ein Programm, das diese Datenstruktur benutzt, ist "Pufferdemo", das dem vorhergehenden ähnlich ist. Die beiden Hauptprozeduren sind auch hier wieder Einfügen und Entfernen. Der wichtigste Unterschied zum vorherigen Programm besteht in der Verwendung zweier Zeiger, einen auf die Spitze der Kette und einen auf das Ende. Von der Spitze werden Elemente entfernt, am Ende werden neue angefügt. Auf diese Weise erreicht man die angestrebte Struktur. In unserem Programm heißen die beiden Zeiger top und bottom. Die boolesche Variable leer zeigt an, ob der Puffer leer ist. Im Gegensatz zum vorherigen Programm wurde hier die Prozedur init mitaufgenommen, die das Initialisieren des Puffers übernimmt.

Die Prozedur Einfügen muß bei der Pufferverwaltung einige Sonderfälle berücksichtigen. Ist der Puffer leer, zeigen top und bottom auf nil. Ein neues Element ist dann sowohl das erste als auch das letzte Element. Nun zeigen top und bottom auf diese neue Variable.

Im zweiten Fall sind ein oder mehrere Elemente im Puffer. Hier muß der Zeiger auf die Spitze des Puffers nicht verändert werden. Das Ende wird erweitert, indem der Zeiger des letzten Datenelements auf das neue Element gesetzt wird, um die Verkettung zu erhalten. Dann wird der Zeiger bottom auf dieses Element gesetzt, das jetzt das neue Ende darstellt.

Das Entfernen von Elementen aus dem Puffer ist genauso einfach wie im vorherigen Beispiel. Wenn der Zeiger

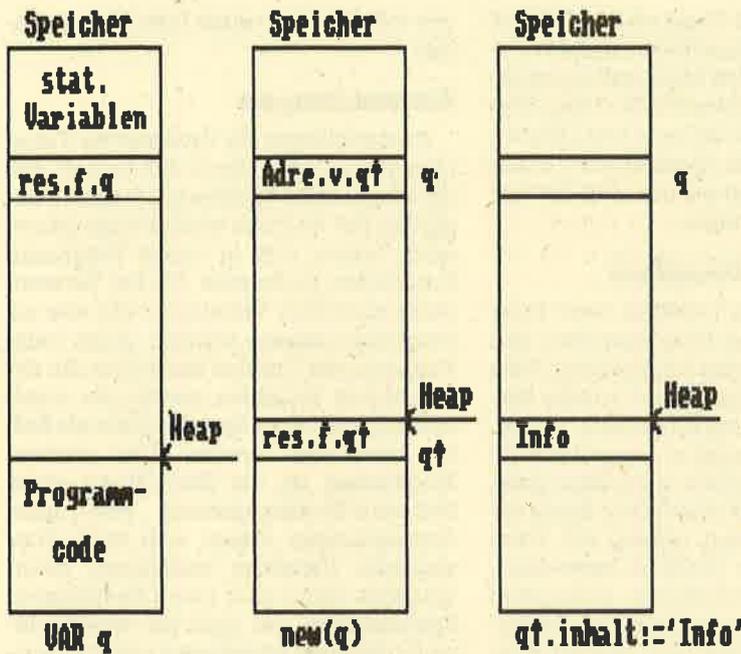


Bild 1

Vorgänge in Speicher bei Verwendung dynamischer Variablen



Bild 2a Einfach verkettete Liste

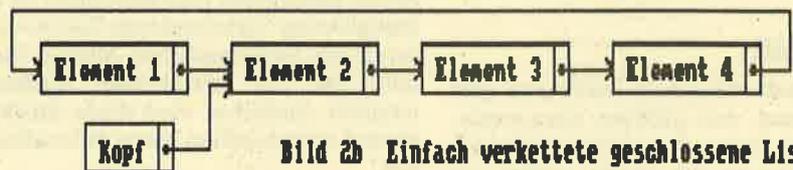


Bild 2b Einfach verkettete geschlossene Liste

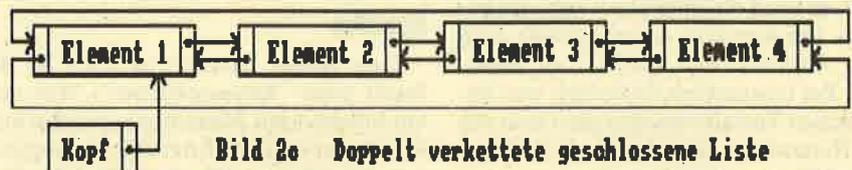


Bild 2c Doppelt verkettete geschlossene Liste

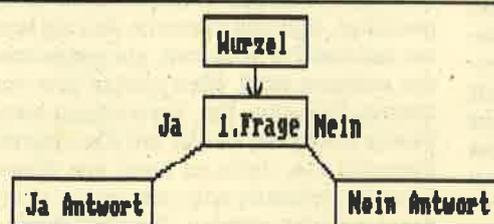


Bild 3a

Baum mit einer Frage

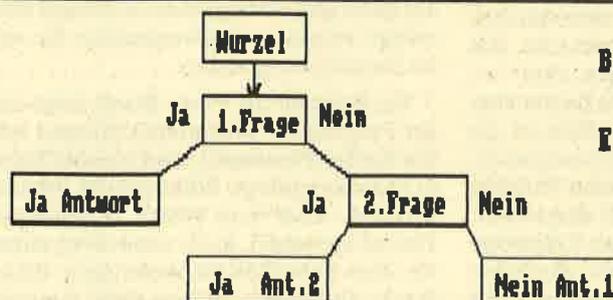


Bild 3b

Einfügen einer Frage

auf die Spitze nicht auf nil gerichtet ist, wird das Element, auf das top zeigt, entfernt. Ist der Puffer leer, so wird der Zeiger auf das Ende (bottom) auch auf nil gesetzt.

Die Prozedur ausgabe dient zur Ausgabe des Puffers, während das Hauptprogramm den Umgang mit einem Puffer demonstriert. Durch Experimentieren mit diesem Programm kann man auch hier sehr gut die Funktionsweise erkennen.

Diese beiden Programme zeigen Datenstrukturen, die in der Praxis häufig benötigt werden. Ihre Kenntnis ermöglicht das Entwickeln von anspruchsvollen Programmen, die mit relativ geringem Aufwand erstellt werden können und trotzdem leistungsfähig sind.

Listen

Neben den beschriebenen Datenstrukturen Stack und Puffer, die von ihrer Organisation her relativ einfach sind, spielen Listen in der Informatik eine große Rolle. Unter einer Liste kann sich sicher jeder etwas vorstellen. Listen außerhalb der Informatik entstehen meistens, um umfangreiche Sachverhalte anschaulich darzustellen. Grundlage ist immer ein gewisses Ordnungsprinzip, denn sonst fehlt die Übersicht. In der Informatik bildet man Listen, um auch dem Computer eine Ordnung beizubringen. Natürlich besteht die Möglichkeit, auch durch andere Datenstrukturen Ordnung zu schaffen. Beispielsweise gibt es verschiedene Routinen, die ein Array in einer bestimmten Reihenfolge sortieren. Der Nachteil dieser Methode ist jedoch der proportional zur Größe des Arrays steigende Verwaltungs- und damit Zeitaufwand. Günstiger in dieser Beziehung sind die Listen. Während die Reihenfolge der Elemente innerhalb eines Arrays festliegt und nicht verändert werden kann, stehen die Listenelemente an einer beliebigen Stelle im Speicher. Die Liste selbst wird durch einige Zeiger gebildet, die auf das nachfolgende oder auch auf das vorhergehende Element zeigen. Soll in diese Liste ein Element eingefügt werden, wird an einer freien Stelle im Speicher Platz für die neue Variable reserviert, und anschließend werden nur einige Zeiger verbogen. Ein Array muß nach Einfügen eines neuen Elements meistens neu sortiert werden. Das Umkopieren der einzelnen Elemente dauert natürlich länger als das Verbiegen einiger Zeiger.

Es gibt verschiedene Listenstrukturen. Die einfachste Form ist die einfach verkettete Liste (Bild 2a). Hier existiert in den Elementen neben den Informationen ein einzelner Zeiger, der auf das nächste Listenelement zeigt. Um die Li-

ste ansprechen zu können, existiert ein Zeiger auf den Anfang (Kopf); das letzte Listenelement zeigt auf nil. Will man ein Element dieser Liste finden, muß man sie immer vom Kopf an durchsuchen, bis es gefunden ist. Das Einfügen eines Elements in diese Liste gestaltet sich recht einfach, man muß aber den Sonderfall der leeren Liste berücksichtigen. Ein Beispiel für diese Listenform ist das Programm "Listendemo".

Diesen Nachteil vermeidet die einfach geschlossene Liste (Bild 2b). Hier zeigt das letzte Element nicht mehr auf nil, sondern auf das erste Element. Auch bei dieser Struktur ist ein Zeiger auf den Kopf der Liste notwendig. Sie läßt sich, ebenso wie die einfach verkettete Liste, nur in eine Richtung durchsuchen, im ungünstigsten Fall muß zuerst die gesamte Liste durchsucht werden, bevor das Element gefunden wird.

Durch Verwendung eines zweiten Zeigers, der auf das vorherige Element zeigt, läßt sich dieser Nachteil beheben. Nimmt man dann eine geschlossene Liste, so läßt sich diese von jeder beliebigen Stelle in jede beliebige Richtung durchsuchen (Bild 2c). Trotzdem sollte ein Zeiger auf ihren Anfang gesetzt werden, um verschiedene Operationen, wie Ausgabe der Liste auf Bildschirm, Drucker oder Floppy, zu beschleunigen. Neben diesen drei vorgestellten Listenformen gibt es natürlich noch weitere Kombinationen, die für einen speziellen Fall erforderlich sind.

Als Demonstration für die Operation mit Listen soll unser Beispielprogramm "Listendemo" dienen. Es liest das Inhaltsverzeichnis von Laufwerk A und erstellt eine alphabetisch geordnete Liste der Filenamen. Diese wird dann auf dem Bildschirm ausgegeben. Wie man das Inhaltsverzeichnis unter CP/M in Turbo Pascal liest, haben wir bereits in einer vorherigen Folge beschrieben. In diesem Programm ist die Prozedur Einfuegen wichtig. Ihr wird der gefundene Eintrag übergeben. Anhand der Funktion Kleiner, die zwei Strings vergleicht und zu True wird, wenn der erste String kleiner ist als der zweite, wird der übergebene Eintrag in die bereits bestehende Liste eingefügt. Dazu benötigt die Prozedur die zwei Hilfszeiger p und q; q wird auf den Anfang gesetzt, p auf nil. Solange das Listenelement nicht erreicht und der Eintrag kleiner als der neue Eintrag ist, wird p mit q gleichgesetzt und q auf den Nachfolger in der Liste gesetzt. Nun zeigt p also auf das vorherige und q auf das aktuelle Element der Liste. Erfüllt kein Element die Bedingung (das neue ist das bisher kleinste Element) oder ist die Liste leer, so zeigt p auf nil. Daran kann der Sonderfall der leeren Li-

ste und des Einfügens am Listenanfang erkannt werden. Im anderen Fall zeigt p auf das Element, nach dem das neue Element eingefügt werden muß. Für dieses wird ein Listeneintrag erzeugt (new(q); q^.filename = eintrag). Wenn p auf nil zeigt (s.o.), dann wird das neue Element durch die Folge

```
q^.naechster = liste; liste = q
```

am Anfang der Liste eingefügt. Im anderen Fall wird es durch

```
q^.naechster = p^.naechster; p^.naechster = q
```

in die Liste eingefügt. Genauso einfach wie das Einfügen ist das Löschen. Dazu muß sein Vorgänger bekannt sein. Dann reicht es aus, den Zeiger des Vorgängers auf den Nachfolger des zu löschenden Elements zu setzen. Um den durch das gelöschte Element besetzten Speicherplatz wieder nutzen zu können, sollte ein Zeiger auf dieses Element vorhanden sein, damit man es mittels Dispose endgültig löschen kann. Die notwendigen Programmschritte sehen folgendermaßen aus.

Der Zeiger p soll auf das zu löschende Element zeigen, q auf den Vorgänger.

```
q^.naechster = p^.naechster; dispose(p);
```

Die im Beispielprogramm enthaltene Prozedur Einfuegen kann für beliebige Dateninhalte benutzt werden. Dazu ist nur die Record-Definition zu ändern. Auch eine andere Reihen- bzw. Rangfolge läßt sich erreichen, indem man die Funktion Kleiner den eigenen Wünschen anpaßt. Die Verwaltung einfacher Listen ist leicht durchführbar und dabei doch sehr effektiv. Soll die Liste eine andere Struktur erhalten, so ist die Prozedur Einfuegen der verwendeten Struktur anzupassen. Besonders bei doppelt verketteten Listen müssen beide Zeiger richtig gesetzt werden. Insgesamt lassen sich mit Listen sehr viele Anwendungen realisieren.

Bäume

Eine weitere Steigerung gegenüber der Listenstruktur ist die Baumstruktur. Diese Struktur kann man sich wirklich wie einen umgekehrten Baum vorstellen. An der Spitze ist die Wurzel, die sich weiter verzweigt, bis sie an einem Blatt endet. Die dabei entstehenden Verzweigungspunkte heißen Knoten des Baumes. Gehen von jedem Knoten maximal n Verzweigungen aus, so spricht man von einem Baum n-ter Ordnung. Von besonderer Bedeutung sind hier die Bäume mit n=2, die auch binäre Bäume genannt werden. Bei ihnen gehen von jedem Knoten maximal zwei Verbindungen aus. Man spricht beim Knoten vom Vater, bei den beiden Verbindungen

vom rechten und linken Sohn. Die Verwendung der Baumstruktur hat verschiedene Vorteile. Während bei einer Listenstruktur im Durchschnitt mindestens die Hälfte der Liste durchsucht werden muß, kommt man bei Bäumen, die eine der Anwendung optimal angepaßte Struktur haben, mit wesentlich weniger Suchoperationen aus. Dieser Vorteil tritt besonders dann in den Vordergrund, wenn die Daten aufgrund ihres Umfangs nicht mehr in den Hauptspeicher passen, sondern sich auf einem externen Datenträger befinden. Hier ist für jede Suchoperation ein Zugriff auf den externen Datenträger notwendig, was bekanntlich ziemlich viel Zeit in Anspruch nimmt.

Der zweite große Vorteil ist die einfache Beschreibung eines Baumes. Diese läßt sich nämlich rekursiv aufbauen. Für seine Beschreibung gilt: Er ist entweder leer, oder er besteht aus n weiteren Teilbäumen, wobei n die Ordnung des Baumes angibt. Mittels dieser rekursiven Beschreibung lassen sich ganz einfache Operationen mit Bäumen entwickeln, die auch wieder rekursiv sind. Da die Rekursion hier ein mächtiges Mittel darstellt, soll an dieser Stelle eine Erklärung gegeben werden.

Rekursion

Ein Algorithmus läßt sich prinzipiell auf zwei Arten beschreiben. Der eine Weg wird als Iteration bezeichnet. Dabei wird von einem bestimmten Startwert bis zu einem vorbestimmten Höchstwert gerechnet. Diese Vorgehensweise ist allgemein verständlich und wird auch meistens angewandt.

Der andere Weg wird als Rekursion bezeichnet. Bei der Rekursion wird das zu behandelnde Problem auf ein kleineres Problem umgerechnet. Dieser Vorgang wird so oft wiederholt, bis auch das kleinste Problem gelöst ist. Die Rekursion ist in den meisten Fällen nicht so leicht zu durchschauen wie die Iteration, in den meisten Fällen aber wesentlich leistungsfähiger. Um den Unterschied anschaulicher zu machen, betrachten wir die Berechnung der Fakultät einmal mit Iteration und einmal per Rekursion.

```
Iteration: Begin
           Fak: = 1;
           For i: = 1 to n do
             Fak: = Fak*i;
           End;
```

```
Rekursion: Function Fakultaet (n: real): real;
           Begin
             If n = 1 then Fakultaet: = 1
             Else
               Fakultaet: = Fakultaet (n-1);
             End;
```

Hier ist der Vorteil der Rekursion nicht ganz so groß wie in anderen Fällen. Doch zur Erklärung eignet sich dieses Beispiel hervorragend. Beide Berechnungsarten gehen von verschiedenen Formeln für die Fakultät aus. Die Iteration benutzt die allgemein bekannte Formel:

$$n! = 1 \times 2 \times 3 \times \dots \times n$$

Die Rekursion benutzt dagegen die weniger bekannte Formel:

$$n! = n \times (n-1)!$$

Das Problem der Fakultät von n wird also auf das Problem der Fakultät von $n-1$ zurückgeführt. Bekannt ist, daß $1! = 1$ ist. Wiederholt man diesen Vorgang des Reduzierens solange, bis man $1!$ erreicht hat, so ist die Fakultät insgesamt berechnet. Die Bedingung $1! = 1$ wird auch als Abbruchbedingung bezeichnet.

Als Beispiel für einen binären Baum soll das letzte Programm des Pascal-Kurses dienen. Das bekannte Spiel "Tiere-Raten" ist hier in einer Pascal-Version mit einem binären Baum realisiert. Seine Struktur ist in Bild 3a angegeben. Der dort skizzierte Baum enthält eine Frage sowie eine Ja-Antwort und eine Nein-Antwort. Als Datenelement dient der Record mit dem Namen Eintrag. Es wurde darauf verzichtet, für Blätter und Knoten getrennte Elemente zu definieren. Ein Blatt wird dadurch gekennzeichnet, daß sowohl jfolger als auch nfolger auf nil zeigen.

Das Programm gliedert sich in drei Hauptprozeduren. Dies sind fileout, filein und abfrage. Neben diesen drei Prozeduren gibt es die Funktionen getantwort und open_file sowie die Prozedur neue_datei. Zum Schreiben bzw. Lesen eines Baumes auf Diskette dienen fileout und filein. Die Funktion open_file eröffnet dazu eine entsprechende Datei. Ist der Versuch erfolglos, kann mit der Prozedur neue_datei eine neue Datenbasis definiert werden. Die Funktion getantwort erfragt vom Benutzer eine Antwort und speichert auch die vorherige, die später noch benötigt wird.

Die beiden Prozeduren fileout und filein sind rekursiv aufgebaut. Unter dem Betriebssystem CP/M muß vor Verwendung rekursiver Funktionen und Prozeduren die Compileroption a deaktiviert werden. Nur so wird ein Code erzeugt, der rekursionsfähig ist.

Um einen Baum in eine sequentielle Folge umzuwandeln, gibt es drei Möglichkeiten: preorder, inorder und postorder. Der Unterschied besteht nur in der Reihenfolge, in der der Baum durchlaufen wird. In unserem Programm wurde die preorder-Möglichkeit realisiert. Solange kein Blatt erreicht ist, wird die Knoteninformation auf Diskette ge-

schrieben, danach der linke Teilbaum bearbeitet, dann der rechte. Bei der inorder-Möglichkeit wird zuerst der linke Teilbaum bearbeitet, dann die Information auf Diskette geschrieben und anschließend der rechte Teilbaum bearbeitet. Die postorder-Möglichkeit schließlich bearbeitet zuerst den linken, dann den rechten Teilbaum und speichert anschließend die Informationen auf Diskette.

Das Einlesen eines solchen Baumes muß natürlich entsprechend der Erzeugung erfolgen und wird durch die Prozedur filein realisiert. Diese benutzt eine Funktion, die einen Satz von der Diskette liest und entsprechend ihrem Wert entweder auf nil gesetzt wird oder einen weiteren Satz von der Diskette holt. Auf diese Weise ist es möglich, die mehrschichtige Struktur des Baumes in eine sequentielle Dateistruktur und wieder zurück zu verwandeln.

Der Dialog mit dem Benutzer wird über die Prozedur abfrage erledigt. Ausgehend von der Wurzel wird die erste Frage gestellt. Je nach Antwort wird dann auf dem linken oder auf dem rechten Teilbaum weitergefragt. Bei Erreichen eines Blattes (jfolger und nfolger gleich nil) wird eine Vermutung geäu-



500.- Deutsche Mark gibt es jeden Monat für den besten Tip oder Trick rund um die Schneider-Computer. Ob CPC 464, 664, 6128 oder Joyce-Computer, jeder hat die gleiche Chance. Schauen Sie mal nach, ob nicht irgendwo auf einer Diskette oder Cassette ein 500-DM-Tip auf seine Entdeckung wartet. Wenn ja, dann nichts wie her damit.

Kennwort:
»Tips und Tricks«

CPC-Magazin
Postfach 1640
7518 Bretten

Bert und abgefragt. Trifft die Vermutung zu, ist das Spiel beendet. Ansonsten wird nach dem erdachten Tier und wie man es von der Vermutung unterscheiden kann gefragt. Nachdem alle diese Eingaben getätigt sind, wird der Baum um diese Frage erweitert. Dieser Vorgang ist in Bild 3b beschrieben. Die Darstellung geht davon aus, daß auf die erste Frage mit nein geantwortet wurde, die Antwort aber nicht richtig war. Die zweite Frage hat als Ja die Ja-Antwort 2, als Nein die Nein-Antwort 1.

Durch diese Erweiterungsmöglichkeit lernt unser Programm immer dazu, vorausgesetzt die Dateibasis wird immer abgespeichert. In der Forschung bezüglich der künstlichen Intelligenz wird ein solches System als Expertensystem bezeichnet. Unser System ist natürlich noch kein Experte, sondern muß erst noch mit Wissen angefüllt werden. Außerdem sind die Verzweigungsmöglichkeiten stark eingeschränkt. Trotzdem kann man sich auch mit diesen einfachen Mitteln eine Dateibasis aufbauen, die einem bei Routinearbeiten unterstützt. Dabei ist man natürlich nicht auf Tiere beschränkt, sondern kann jedes beliebige Thema als Grundlage für eine Datenbasis nehmen. Insbesondere Dia-

gnoseprogramme benutzen Expertensysteme für die schnelle Fehlereinkreisung. Die Forschung auf diesem Gebiet ist in vollem Gange, so daß hier noch erhebliche Fortschritte zu erwarten sind.

Das umfangreiche Gebiet der Listen und Bäume konnte innerhalb dieses Kurses natürlich nur angerissen werden. Der Stoff über diese Datenstrukturen füllt Vorlesungen, und mehrere Bücher haben sich auf dieses Thema spezialisiert. Mit unserer kleinen Einführung wollten wir Ihnen nur ein wenig Appetit machen. Die Programmierung mit dynamischen Variablen eröffnet dem Anwender ganz neue Möglichkeiten und ist die eigentlich größte Stärke der Programmiersprache Pascal.

Mit dieser Folge endet unser Pascal-Kursus, der für diese Programmiersprache Interesse wecken sollte. Natürlich konnten nicht alle Aspekte behandelt werden; insbesondere die Schneiderspezifischen Möglichkeiten sind etwas zu kurz gekommen. In der Zukunft werden wir sicher einige weitere Beiträge über Pascal auf dem Schneider veröffentlichten. Für Anregungen diesbezüglich sind wir dankbar.

H.J. Janke

Stackdemo

Program Stachdemo1

(Die Wirkungsweise eines Stacks oder Kellerspeichers soll mit diesem Programm demonstriert werden)

```
TYPE zeiger=^stack;
      stack = RECORD
          inhalt : string[20];
          naechster : zeiger;
      END;
      str20 = string[20];
```

```
VAR
    top : zeiger;
    stackvar: stack;
    ch : char;
    line : str20;
    leer : boolean;
```

PROCEDURE einfüegen (info :str20);

VAR neu : zeiger;

```
BEGIN
    neu(new);
    neu^.inhalt:=info;
    neu^.naechster:=top;
    top:=neu;
    leer:=false;
END;
```

PROCEDURE entfernen ;

VAR p : zeiger;

MN Michael Naujoks

CPC	Preise: Cass./Disk	Joyce	Preise: Cass./Disk
Split Personalities	DM 29,- /49,-	Cyrus Chess	DM 59,-
Boulder	DM 27,90/44,90	3-D-Clock Chess	DM 49,-
Stainless Steel	DM 31,- /49,-	Colossus Chess	DM 59,-
Rescue on Fractalus!	DM 36,- /49,-	Aftershock	DM 59,-
Glass	DM 36,- /48,-	Bat Man	DM 49,-
Mission Omega	DM 33,90/49,-	Fairlight	DM 49,-
Johnny Reb II	DM 33,90/44,90	Hitchhikers Guide	DM 89,-
Hunchback		Lord of the Rings	DM 89,-
- The Adventure	DM 36,- /,-,-	Tomahawk	DM 69,-
Nick Faldo	DM 36,- /,-,-	SAS Raid	DM 49,-
Room 10	DM 29,- /49,90	Adreßverwaltung	DM 129,-
Knight Games	DM 36,- /59,-	Vereinsverwaltung	DM 249,-
Ghosts'n Goblins	DM 35,- /49,-		

Hardware SCHNEIDER CPC Software

dk'tronics Speech-Synthesizer (Cassette 464/664)	DM 109,-	dk'tronics Speichererweiterungen 64K für 464/664	DM 129,-
(ROM 464/664)	DM 149,-	256K für 464/664	DM 329,-
(ROM 6128)	* DM 149,-	256K für 6128	* DM 329,-
dk'tronics Light-Pen (Farbmon.) (Cassette 464/664)	DM 59,-	dk'tronics Silicon-Disc 64K für 6128	* DM 98,-
(ROM 464/664)	DM 89,-	256K für 464/664	DM 309,-
(ROM 6128)	* DM 89,-	256K für 6128	* DM 309,-

Adapter für Geräte mit *: DM 29,-.

Alle Geräte haben einen durchgeführten Systembus und können hintereinander auf den Erweiterungsport gesteckt werden. Für die mit * gekennzeichneten Geräte benötigen Sie deshalb auch nur einen Adapter zur Umsetzung von Schneider- auf Amstrad-Anschluß. Auf alle Geräte 6 Monate Vollgarantie. Händleranfragen erwünscht.

Kostenlosen Katalog B10/86 anfordern!

Entwicklung & Vertrieb von
Computer Soft- und Hardware
Rottmannstr. 40, 6900 Heidelberg

Hotline:
(06221) 46885

Mac Lin

Sind es leid, teure Programme zu kaufen, die dann doch nicht Ihren Ansprüchen gerecht werden oder zuviel versprochen haben? Dann müssen Sie **Mac Lin 2 Nucleus**, den **Programmgenerator** kaufen. Sie können mit **Mac Lin 2 Nucleus** selbständig, schnell und individuell Ihre Programme selbst entwickeln, wie Sie es gerne möchten, sogar in der **Relativ-Diskettenverwaltung**. Programmierkenntnisse sind keine Voraussetzung, denn die Eingaben erfolgen in **Deutsch** und eine **exzellente Bedienführung** führt Sie durch die einzelnen Programmebenen. Der **Mac Lin 2 Nucleus** kostet bei uns exklusiv für Sie **nur DM 450,-**.

Vielleicht interessiert Sie aber auch unser **Komplettangebot: Joyce Plus PCW 8512 + Mac Lin 2 Nucleus zum Schlagereinführungspreis von nur DM 2548,-**

Exklusiv bei uns: **Mac Lin 2 Nucleus** in der **MS-DOS-Version** für den neuen **Schneider 16-Bit PC**. **Einführungspreis DM 848,90**
Händleranfragen erwünscht

COMPUTER

Joyce Plus PCW 8512	2098,- DM
CPC 464 + DDI-1 + 25 Spielprogramme	1198,- DM

Der **NEUE Schneider 16-Bit-Computer** ist da, zu unseren **Superpreisen:**

PC-1512 SD mit PC-MM	1948,- DM
PC-1512 SD mit PC-CM	2448,- DM
PC-1512 DD mit PC-MM	2448,- DM
PC-1512 DD mit PC-CM	2948,- DM
PC-1512 SD mit HD 10-MM	2948,- DM
PC-1512 SD mit HD 10-CM	3448,- DM
PC-1512 SD mit HD 20-MM	3948,- DM
PC-1512 SD mit HD 20-CM	4448,- DM

Disketten und Zubehör

Panasonic 3"-Disketten CF 2 D	10er Pack	75,- DM
No Name 3.5"-Disketten MF 1 DD	10er Pack	39,95 DM
No Name 3.5"-Disketten MF 2 DD	10er Pack	45,90 DM
No Name 5.25"-Disketten MD 1 D	10er Pack	9,90 DM
No Name 5.25"-Disketten MD 2 D	10er Pack	12,50 DM
Diskettenbox SS 50 für 50 3"- od. 3.5"-Disketten, abschließbar/tragbar, Rauchglas		19,90 DM
Diskettenbox DX 85 für 5.25"-Disketten		19,90 DM
LCD-Herren-Armbanduhr, 30 mtr. wasserdicht		9,98 DM



Ferdi's Computer Software
EDV-Service



Inh. Ferdinand Göddeker
Höftestraße 32 · 4400 Münster · Tel. 02 51 / 61 98 81

```

BEGIN
IF not leer THEN BEGIN
  writeln(top^.inhalt)
  p:=top;
  top:=p^.naechster;
  dispose(p) END
ELSE
  writeln('Stack ist leer!')
IF top=nil THEN
  leer:=true;
END;

PROCEDURE ausgabe ;
VAR i:integer;
    p:zeiger;

BEGIN
  p:=top;
  clrscr;
  gotoxy(1,5);
  i:=0;
  IF p<>nil THEN
  BEGIN
    WHILE (p <> NIL) AND
      (i<20) DO
      WITH p^ DO
      BEGIN
        lowvideo;
        writeln(inhalt);
        p:=naechster;
        i:=i+1;
      END;
      IF i<20 THEN
        write ('<<NIL>>');
      normvideo;
      gotoxy(22,5);
      write('<-- TOP');
    END
  ELSE
    writeln('Stack ist leer!');
  END;
END;

BEGIN
  top:=nil;
  leer:=true;
  ch:= ' ';
  WHILE upcase(ch)<>'Q' DO
  BEGIN
    ausgabe;
    gotoxy(40,10);
    write ('(E)ingeben,(A)usgeben,
      ' oder (Q)uit ?');
    read (kbd,ch);
    ch:=upcase(ch);
    CASE ch OF
      'E' : BEGIN
        gotoxy(40,12);
        writeln('Bitte ',
          'Text eingeben ');
        gotoxy(40,14);
        readln(line);
        einfuegen(line);
      END;
      'A' : IF not leer THEN
        entfernen;
    END;
  END;
END;

```

Pufferdemo

PROGRAM pufferdemo

(Dieses Programm realisiert einen Pufferspeicher, wie er z.B. in Terminalprogrammen Verwendung findet)

```

TYPE  str20  = string[20];
      zeiger = ^puffrec;
      puffrec = RECORD
        inhalt  : str20;
        naechster : zeiger;
      END;

VAR   top,bottom : zeiger;
      line       : str20;
      ch         : char;
      leer       : boolean;

```

```

PROCEDURE init;
BEGIN
  leer:=true;
  top:=nil;
  bottom:=nil;
END;

PROCEDURE einfuegen (info : str20);
VAR neu : zeiger;

BEGIN
  neu:=new;
  neu^.inhalt:=info;
  neu^.naechster:=nil;
  IF bottom <> nil THEN
    bottom^.naechster:=neu;
  ELSE IF top <> nil THEN
    top^.naechster:=neu;
  ELSE
    top:=neu;
    bottom:=neu;
  leer:=false;
END;

PROCEDURE entfernen ;
VAR p : zeiger;

BEGIN
  IF top<> nil THEN
  BEGIN
    write(top^.inhalt);
    p:=top;
    top:=p^.naechster;
    dispose(p);
    leer:=top=nil;
    IF leer THEN
      bottom:=nil;
    END
  ELSE
    leer:=true;
  END;
END;

PROCEDURE ausgabe;
VAR p : zeiger;

BEGIN
  clrscr;
  p:=top;
  WHILE p<> nil DO
  BEGIN
    WITH p^ DO
    BEGIN
      lowvideo;
      write (inhalt, ' ');
      p:=naechster;
    END;
    normvideo;
  END;
  IF top=nil THEN
    write ('Puffer ist leer!');
  END;
END;

BEGIN
  init;
  ch:= ' ';
  WHILE upcase(ch) <> 'Q' DO
  BEGIN
    ausgabe;
    gotoxy(1,22);
    write('(E)ingeben,(A)hohlen oder ',
      '(Q)uit');
    read(kbd,ch);
    ch:=upcase(ch);
    CASE ch OF
      'E' : BEGIN
        gotoxy(1,25);
        write('Bitte Text ',
          'eingeben ');
        readln(line);
        einfuegen(line);
      END;
      'A' : IF not leer THEN
        entfernen;
    ELSE
      write('Puffer ist leer');
    END;
  END;

```

```

END;
END;
END.

```

Listendemo

Program Listendemo

(Dieses Programm zeigt die Anwendung einer geordneten Liste. In diesem Fall wird das Directory eingelesen und sortiert ausgegeben)

```

TYPE  da = string[255];
      typ = string[35];
      zeiger = ^listenrec;
      listenrec = RECORD
        filename:string[20];
        naechster:zeiger;
      END;
      name = STRING[20];

VAR   liste : zeiger;
      eintrag: name;
      dirname: name;
      fcb : typ;
      dma : da;

```

PROCEDURE reset;

```

BEGIN
  bdos(13);
END;

```

PROCEDURE anmelden (drive:byte);

```

BEGIN
  IF drive>0 THEN
    bdos(14,(drive-1) and 15);
  END;

```

FUNCTION erster_Eintrag (byte);

```

BEGIN
  bdos(26,addr(dma)+1);
  Erster_Eintrag:=bdos(17,addr(fcb)+1);
END;

```

FUNCTION naechster_Eintrag : byte;

```

BEGIN
  Naechster_Eintrag:=bdos(18);
END;

```

FUNCTION kleiner (str1,str2:name) : boolean;

VAR i : integer;

```

BEGIN
  kleiner:= (str1<str2);
END;

```

PROCEDURE einfuegen (eintrag:name);

VAR p,q : zeiger;

```

BEGIN
  q:=liste;
  p:=nil;
  WHILE q<>nil DO
    WITH q^ DO
      IF kleiner(filename,
        eintrag) THEN
        BEGIN
          p:=q; q:=naechster; END
        ELSE
          q:=nil;
          new(q);
          q^.filename:=eintrag;
          IF p=nil THEN
            BEGIN
              q^.naechster:=liste;
              liste:=q;
            END
          ELSE BEGIN
              q^.naechster:=p^.naechster;
              p^.naechster:=q;
            END;
          END;

```

```

          END;
          ELSE BEGIN
            write('Puffer ist leer');
          END;
        END;
      END;
    END;
  END;

```

```

PROCEDURE lies_directory;
VAR i:integer;
    d:byte;
BEGIN
  reset;
  FOR i:=1 TO 35 DO
    fcb[i]:=chr(0);
  FOR i:=2 TO 12 DO
    fcb[i]:='?';
  liste:=nil;
  d:=ster_eintrag;
  WHILE d < 255 DO
    BEGIN
      dirname:=copy(dma,dx32+2,11);
      insert('.',dirname,9);
      FOR i:=1 TO 12 DO
        dirname[i]:=chr(ord
          (dirname[i]) and 127);
      einfuegen(dirname);
      d:=naechster_eintrag;
    END;
  END;

```

```

PROCEDURE ausgabe;
VAR p:zeiger;
    i:integer;
BEGIN
  p:=liste;
  i:=0;
  clrscr;
  WHILE p<>nil DO
    WITH p^ DO
      BEGIN
        write(p^.filename:15,' ');
        i:=succ(i);
        IF (i mod 4)=0 THEN writeln;
        p:=naechster;
      END;
    END;
  BEGIN
    lies_directory;
    ausgabe;
  END.

```

Tiere

```

Program tiere;
( Das Spiel Tiererraten hier in einer
  Pascalversion zur Demonstration
  binärer Baume )
CONST ja = true;
      nein = false;
      ende = STRING[60]='NIL';
TYPE zeiger = ^eintrag;
      eintrag = RECORD
        fragant : string[60];
        jfolger : zeiger;
        nfolger : zeiger;
      END;
      str60 = string[60];
VAR sfile : file of str60;
    wurzel : zeiger;
    lastant : boolean;
    ch : char;
    filename : string[20];

```

```

(0A-)
PROCEDURE fileout (p:zeiger);
BEGIN
  IF p<>nil THEN
    BEGIN
      write(sfile,p^.fragant);
      fileout(p^.jfolger);
      fileout(p^.nfolger);
    END
  ELSE
    write(sfile,ende);
END;

```

```

PROCEDURE filein;
FUNCTION file_in : zeiger;
VAR q : zeiger;
    c : str60;
BEGIN
  IF not eof(sfile) THEN
    BEGIN
      read(sfile,c);
      IF c=ende THEN file_in:=nil;
      ELSE
        BEGIN
          new(q);
          q^.fragant:=c;
          q^.jfolger:=file_in;
          q^.nfolger:=file_in;
          file_in:=q;
        END;
      END;
    END;
  BEGIN
    reset(sfile);
    IF filesize(sfile)>0 THEN
      wurzel:=file_in;
    END;
  (0A+)

```

```

FUNCTION getantwort : boolean;
BEGIN
  lastant:=chr('J');
  REPEAT
    read(kbd,ch);
    ch:=upcase(ch);
  UNTIL ch IN ['J','N'];
  IF ch='J' THEN BEGIN
    getantwort:=ja;
    writeln(' Ja ');
  END;
  ELSE BEGIN
    getantwort:=nein;
    writeln(' Nein ');
  END;
END;

```

```

PROCEDURE neue_Datei;
VAR p,q : zeiger;
BEGIN
  new(p);
  wurzel:=p;
  WITH p^ DO BEGIN
    jfolger:=nil;
    nfolger:=nil;
    writeln('Erste Frage ? ');
    readln(fragant);
    new(q);
    q^.jfolger:=nil;
    q^.nfolger:=nil;
    writeln('Antwort auf Ja ? ');
    readln(q^.fragant);
    jfolger:=q;
    new(q);
    q^.jfolger:=nil;
    q^.nfolger:=nil;
    writeln('Antwort auf Nein ? ');
    readln(q^.fragant);
    nfolger:=q;
  END;
END;

```

```

FUNCTION open_file : boolean;
BEGIN
  clrscr;
  gotoxy(10,10);
  write('Bitte Dateinamen eingeben: ');
  readln(filename);
  assign(sfile,filename);
  (0I-) reset(sfile) (0I+);
  open_file:=ioreresult=0;
  IF ioreresult<>0 THEN
    BEGIN
      writeln(filename,' nicht vorhanden!');
      close(sfile);
    END;
  END;

```

```

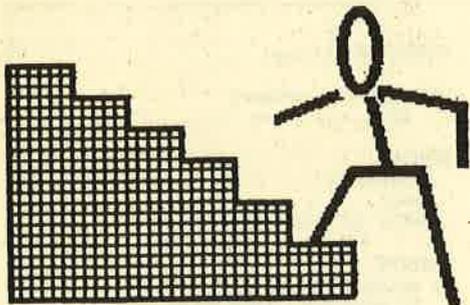
PROCEDURE abfrage;
VAR p,q,r,s : zeiger;
    chl : char;
BEGIN
  p:=wurzeln;
  simp;
  WHILE (p^.jfolger<>nil) or
    (p^.nfolger<>nil) DO
    BEGIN
      write(p^.fragant);
      IF getantwort THEN BEGIN
        simp;
        p:=p^.jfolger;
      END;
      ELSE BEGIN
        simp;
        p:=p^.nfolger;
      END;
      writeln;
      writeln('Ich glaube ich weiss es!');
      writeln('Handelt es sich um eine(n) ',
        p^.fragant,'?');
      IF getantwort THEN BEGIN
        writeln('Hab ich mir ''s',
          ' doch gedacht!');
      END;
    END;
  ELSE
    BEGIN
      new(q);
      new(r);
      writeln('Um welches Tier handelt',
        ' es sich dann? ');
      readln(q^.fragant);
      q^.jfolger:=nil;
      q^.nfolger:=nil;
      writeln('Durch welche Frage unter',
        ' scheidet es sich von ',
        p^.fragant,'? ');
      readln(r^.fragant);
      writeln('Wie laut die Antwort',
        ' auf diese Frage? ');
      read(kbd,ch);
      IF upcase(ch)='J' THEN BEGIN
        IF lastant THEN
          s^.jfolger:=r;
        ELSE
          s^.nfolger:=r;
          r^.nfolger:=p;
          r^.jfolger:=q;
        END;
      END;
    END;
  END;

```

```

(0A-)
BEGIN (Hauptprogramm)
  chl:= ' ';
  wurzel:=nil;
  IF open_file THEN
    filein;
  ELSE
    BEGIN
      write('Neues Datenfile anlegen? ');
      IF getantwort THEN
        neue_Datei;
      ELSE
        halt;
    END;
  REPEAT
    chl:= ' ';
    abfrage;
    writeln('Neues Spiel? ');
  UNTIL not getantwort;
  writeln('Datenbasis abspeichern? ');
  IF getantwort THEN BEGIN
    write('Bitte Dateiname eingeben: ');
    readln(filename);
    assign(sfile,filename);
    (0I-) rewrite(sfile) (0I+);
    IF ioreresult=0 THEN
      fileout(wurzel);
      close(sfile);
    END;
  END.

```



Basic-Stufen (Teil 6)

Wir programmieren einen Spielautomaten

An dieser Stelle begrüßen wir Sie, liebe Leser, wieder bei unserem Basic-Kurs und hoffen, daß Ihnen bei dem doch recht flotten Tempo nicht schwindelig geworden ist.

Heute wollen wir uns mit der Nachbildung eines in vielen Gaststätten stehenden Spielautomaten befassen. Solch ein Gerät besteht aus mehreren Fenstern, hinter denen verschiedene Bilder erscheinen, die wir bequem mit Windows nachbilden. Dann sind Tasten zur Betätigung der Walzen vorgesehen, die sich hinter den Fenstern drehen. Auch die wollen wir mittels der Fenstertechnik gestalten. Wir setzen – wie im Ausblick des zweiten Teiles bereits angedeutet – die Farbe dieser Fenster auf schwarz. Nur wenn sie leuchten sollen, färben wir sie rot und geben den Text (START oder STOP) mit schwarzer Schrift aus. Die Durchführung sehen Sie in den Zeilen 70-120 und 160-210. Es mag Sie überraschen, daß diese Teile so weit vorne stehen, während sich die Ihnen bekannten Fenstereinteilungen und Ink-Zuweisungen am Ende des Programms befinden. Dies hat aber seinen Grund. Jedesmal, wenn der CPC einen GOTO- oder GOSUB-Befehl erhält, beginnt er mit der ersten Zeile und prüft nun alle Zeilen durch, bis er die mit der angegebenen Nummer findet. Damit dürfte klar sein, daß man zur Erreichung einer hohen Abarbeitungsgeschwindigkeit die häufig benutzten Unterprogramme (in unserem Falle die Tastenabfrage) an den Anfang des Programms plazierte, während man die Teile, die nur einmal durchlaufen werden (z.B. das Titelbild oder die Ink-Zuweisungen) ans Ende setzt und von Zeile 30 aus mit GOSUB anspringt.

Unser Spielautomat soll in drei Fenstern bunte Blumen abbilden. Als Gewinne werden 2 oder 3 gleichfarbige Blumen gewertet. Zu Beginn soll der Spieler seinen Einsatz tätigen; Gewinne und Verluste werden jeweils dem Bestand zugerechnet oder abgezogen. Am Ende erhält der Spieler noch eine Auswertung seines Spieles. Zu diesem Zwecke ist es nötig, von allen 7 Windows der Schneider-Computer bei der Gestaltung des Bildschirms Gebrauch zu machen.

Bleiben wir gleich bei diesem Bildschirmaufbau-Teil ab Zeile 960, denn in Zeile 1080 kommt eine neue Variante des PRINT-Befehls vor, er erhält die Ergänzung USING "###". Das bedeutet, daß Sie Ihrem Computer hier mitteilen, die jeweils angetroffene Zahl wie eine dreistellige Zahl zu behandeln und sie mit der Einerstelle immer an der gleichen Position beginnend auszudrucken. Dadurch wird erreicht, daß die Einer immer untereinander stehen, auch wenn es sich um unterschiedlich lange Zahlen handelt. Diejenige mit den meisten Stellen bestimmt die Anzahl der #-Zeichen zwischen den beiden Anführungsstrichen. Soll es sich um Kommazahlen handeln, so wird an der entsprechenden Stelle ein Dezimalpunkt gesetzt.

In der nächsten Zeile 1090 entdecken Sie den nächsten neuen Befehl, SYMBOL AFTER, mit nachfolgendem Zahlenwert. Dieses Kommando dient dazu, dem CPC mitzuteilen, daß man andere als die von ihm zur Verfügung gestellten Zeichen verändern will (das haben wir im letzten Teil durchgenommen). SYMBOL AFTER muß also im Programm einmal vorkommen, der Wert gibt die Nummer des letzten unveränderten Zeichens an. Danach folgt die Definition der neuen Bilder. Benötigt wird dieser Befehl, wenn man entweder mehr Zeichen verändern will oder diese andere Nummern haben sollen als vorgegeben. Im Programm ist letzteres der Fall, denn in Zeile 790 wird die Nummer des auszugehenden Zeichens durch eine Berechnung bestimmt. Auch das ist möglich. Eine Einschränkung gibt es allerdings bei den veränderbaren Zeichen; ihre Nummer darf im niedrigsten Falle 33 sein. Natürlich werden Sie sofort fragen, was es dann mit den Zeichen von 0 bis 32 auf sich hat. Diese sind nicht veränderbar, denn sie dienen der internen Steuerung des Computers, können aber auch benutzt werden. Als Beispiel seien hier die Zeilen 650 und 810 angeführt. Dort treffen wir das Zeichen CHR\$(22) mit folgendem CHR\$(1) bzw. CHR\$(0). Man muß nun wissen, daß das Zeichen 22 für den Transparentmodus zuständig ist, d.h. der CPC druckt dann ein neues

Zeichen über ein bisheriges, läßt dieses aber stehen. Auf diese relativ einfache Art und Weise kann man sich selbst vielfarbige Bilder schaffen. Der Zusatz CHR\$(1) schaltet den Transparentmodus ein, CHR\$(0) schaltet ihn wieder aus. Die genaue Aufzählung aller 32 Steuerzeichen würde den Rahmen einer Einführung sprengen. Die wichtigsten will ich aber vorstellen.

- CHR\$(2) schaltet den Textcursor auf unsichtbar.
- CHR\$(3) schaltet den Textcursor auf sichtbar.
- CHR\$(7) läßt einen Piepston erklingen, der gut als Warnton benutzt werden kann.
- CHR\$(8) setzt den Cursor um ein Zeichen zurück.
- CHR\$(9) setzt den Cursor um ein Zeichen vor.
- CHR\$(10) setzt den Cursor eine Zeile tiefer.
- CHR\$(11) setzt den Cursor eine Zeile höher.
- CHR\$(16) löscht das Zeichen unter dem Cursor.
- CHR\$(24) schaltet auf inverse Darstellung um und wieder zurück.

Dies sind die wichtigsten Steuercodes. Wer sich dafür näher interessiert, der sei auf Listen aller Anwendungen verwiesen.

Ich will mich jetzt einem anderen wichtigen Kapitel zuwenden, den Vergleichen. Wohl jeder Leser einer Computer-Zeitschrift ist in Listings schon auf Zeilen mit IF und verschiedenen Zeichen (=, <, >, <>) gestoßen. Davor und dahinter stehen Variablen, die Zahlenwerte beinhalten.

Was geschieht nun in einer solchen Zeile? Nehmen wir den Fall $a < b$ und ein folgendes THEN GOTO 100 an. Der CPC rennt zu den Schubkästen mit den Variablen a und b, holt sich dort die Werte und vergleicht sie. Wenn a kleiner als b ist, arbeitet er in Zeile 100 weiter, wenn a aber gleich oder größer als b ist, geht er zur Bearbeitung der nächsten Zeile über. Soweit ist alles klar. Aber was soll das Doppelzeichen <>? Es bedeutet,

daß die beiden Werte nicht gleich sein dürfen, nur dann arbeitet der CPC in Zeile 100 weiter; es ersetzt das in der Mathematik übliche durchgestrichene Gleichheitszeichen, welches auf Computern nicht existiert. Testen Sie diese Funktion mit folgendem kleinen Programm.

```
10 cls
20 b=11
30 for a=1 to 20
40 if a<b then print a
50 next a
60 end
```

Lassen Sie es laufen und werten Sie das Ergebnis aus. Dann ersetzen Sie bitte in Zeile 40 das < durch >, danach durch <> und letztlich durch =. Nach 3 weiteren Probeläufen müßte alles klar sein!

An dieser Stelle möchten wir Sie auf eine Eigenart von Basic hinweisen, die bisweilen zu erheblichen Problemen führt. In Zeile 40 der letzten Fassung dient das Gleichheitszeichen einem Vergleich der Variablen a und b. In den Zeilen 20 und 30 wird genau dasselbe Zeichen verwendet, aber hier werden den Variablen a und b verschiedene Werte zugewiesen! Einziger Unterschied für Programmierer und Computer ist der, daß in Zeile 40 vor dem ersten Wert das Befehlswort IF steht. Nach einem vorhergehenden Kommando führt Ihr CPC also einen Vergleich durch, ansonsten weist er Werte zu. Einige Programmiersprachen, wie z.B. C, verwenden für Zuweisungen a=b, für Vergleiche hingegen a==b. Seien Sie bitte sehr vorsichtig mit Eingaben wie a=b=7! Ihr CPC akzeptiert das zwar, nimmt aber das erste Gleichheitszeichen als Wertzuweisung, das zweite wertet er als Vergleich aus. Das heißt, wenn b ungleich 7 ist, schreibt er in a eine 0; wenn b aber

gleich 7 ist, weist er der Variablen a den Wert -1 zu. Sie können das in einer FOR-NEXT-Schleife für b ausprobieren und sich jeweils a und b mit PRINT ausgeben lassen.

Bei unserem Spielautomaten gibt es aber drei Möglichkeiten, daß bei 3 Fenstern 2 gleich sind. Nun kann man für jede Möglichkeit eine Extrazeile schreiben, die Bedingungen aber auch logisch verknüpfen. Betrachten Sie Zeile 400, dort sehen Sie zwischen den Gleichheitszuordnungen das Wort OR. Es bedeutet, daß von den dort aufgeführten Bedingungen eine einzige erfüllt sein muß, dann führt das Programm den THEN-Teil aus. Anders ist es in Zeile 390, dort steht AND, und das sagt dem CPC, daß er nur dann mit THEN weiterarbeiten soll, wenn wirklich beide Bedingungen gleichzeitig zutreffen. In unserem Falle hieße dies, daß alle drei Blumen die gleiche Farbe haben.

Einen weiteren Vergleich finden Sie in den Zeilen 520-550. Dort wird zuerst die Differenz zwischen geleistetem Einsatz und derzeitigem Bestand errechnet. Beachten sie bitte in diesem Zusammenhang, daß bei allen Rechnungen die Variable, die das Ergebnis beinhalten soll, vor dem Gleichheitszeichen steht, während die gewünschte Operation sich dahinter befindet. Dies ist genau umgekehrt zur Mathematik, aber Computern muß man halt leider zuerst sagen, wohin das Ergebnis kommen soll. Nach der Berechnung wird nun je nach Größe der Differenz ein anderer Satz der Variablen satz\$ zugeordnet, der dann in Zeile 590 ausgedruckt wird. In Zeile 600 finden Sie das oben erklärte <>. Es bewirkt hier, daß die Differenz als Zahl nur ausgedruckt wird, wenn sie wirklich vorhanden ist - alles andere wäre ja auch sinnlos!

Neu in diesem Programm ist auch der Befehl INKEY\$. Er ordnet der vorangestellten Textvariablen das nächste auf der Tastatur gedrückte Zeichen zu. Allerdings ist die Abfrage so schnell, daß wir eine Schleife bauen müssen, die bei fehlender Rückmeldung immer die Abfrage wiederholt. Selbstverständlich kann man sich den Buchstaben mittels einer PRINT-Anweisung anschauen wie im folgenden Beispiel:

```
10 cls
20 a$=inkey$
30 if a$="" then goto 20
40 print a$
50 goto 20
60 end
```

Wir aber wollen nichts auslesen, sondern nur wissen, ob eine Taste gedrückt wurde. In diesem Falle genügt uns die Rückmeldung, daß überhaupt etwas betätigt wurde. Die Ausführung entnehmen Sie den Zeilen 180-200. Sobald ein Zeichen eingegeben wurde, fährt der CPC mit der Bearbeitung fort. Beachten Sie, daß zwischen den beiden Anführungszeichen absolut nichts stehen darf. Würden Sie hier ein Leerzeichen eingeben, dann reagiert ihr CPC ausschließlich auf das Drücken der Leertaste!

Wir hoffen, daß Sie nun in der Lage sind, das Listing voll zu durchschauen und nachzuvollziehen. Rückfragen sind jederzeit willkommen. Versuchen Sie doch bitte auch, die Zeile 620 so umzuschreiben, daß der CPC den Spieler fragt "Noch ein Spiel (J/N)?" und die Antwort ausgewertet. Bedenken Sie bitte, daß sowohl J und j als auch N und n gedrückt werden können. Mehr zu den Befehlen mit dem \$-Zeichen erfahren Sie von uns im November. Bis dahin wünsche ich Ihnen viel Basic-Spaß!

Berthold Freier

Blumenspiel

```
10 REM *** Blumenspiel ***
20 REM *** Copyright by Berthold Freier
1986 ***
30 GOSUB 830 : GOSUB 1090 : GOTO 230
40 REM *** START-Taste ***
50 SOUND 1,150,20
60 zeiger=zeiger+1
70 PAPER #4,6:CLS #4
80 LOCATE #4,2,2:PRINT #4,"START"
90 FOR zeit=1 TO 2000
100 eingabe$=INKEY$
110 IF eingabe$="" THEN NEXT zeit ELSE z
eiger=zeiger+1:PAPER #i,11:CLS #i
120 PAPER #4,12:CLS #4
130 RETURN
140 REM *** STOP-Taste ***
150 SOUND 1,150,20
160 PAPER #5,6:CLS #5
170 LOCATE #5,2,2:PRINT #5,"STOP"
180 FOR zeit=1 TO 2000
190 eingabe$=INKEY$
200 IF eingabe$="" THEN NEXT zeit
210 PAPER #5,12:CLS #5
220 RETURN
230 REM *** Steuerteil ***
240 FOR i=1 TO 3:PAPER #i,10:CLS #i:NEXT
i
250 FOR i=1 TO 3
260 zeiger=0
270 PAPER #i,11:CLS #i:SOUND 1,100,40
280 GOSUB 140
290 zeiger=zeiger+1
300 farbe(i)=INT(RND*7)+1
310 GOSUB 630
320 IF i=3 THEN GOTO 360
330 IF zeiger=1 THEN GOSUB 40
340 IF zeiger=3 THEN GOTO 280
```

```

350 FOR zeit=1 TO 1000:NEXT zeit
360 NEXT 1
370 REM *** Gewinnfeststellung ***
380 PEN #7,14:LOCATE #7,3,2

390 IF farbe(1)=farbe(2) AND farbe(2)=farbe(3) THEN PRINT #7,"Strauss 2 DM":bestand=bestand+2:GOTO 420
400 IF farbe(1)=farbe(2) OR farbe(2)=farbe(3) OR farbe(1)=farbe(3) THEN PRINT #7,"Gewinn 1 DM":bestand=bestand+1:GOTO 420

410 PRINT #7,"Verlust 1 DM":bestand=bestand-1
420 PEN #7,2:LOCATE #7,10,6:PRINT #7,USING "###";bestand
430 FOR zeit=1 TO 3000:NEXT zeit
440 LOCATE #7,3,2:PRINT #7,STRING$(12," ")
450 IF bestand<1 THEN GOTO 510
460 REM *** Weiterspielen ***
470 LOCATE #6,2,2:PRINT #6,"WEITERSPIELE N?"
480 FOR zeit=1 TO 2000
490 eingabe$=INKEY$
500 IF eingabe$="" THEN NEXT zeit ELSE CLS #6:GOTO 230
510 REM *** Schlussanzeige ***
520 MODE 1:INK 0,0:INK 1,24:INK 2,6: BORDER 0
530 differenz=bestand-einsatz
540 IF differenz=0 THEN satz$="ohne Gewinn und Verlust."
550 IF differenz>0 THEN satz$="mit einem Gewinn von "
560 IF differenz<0 THEN satz$="mit einem Verlust von "
570 PEN 1:LOCATE 3,7:PRINT "Bei einem Einsatz von ";einsatz;" DM"
580 LOCATE 3,9:PRINT "und einem Bestand von ";bestand;" DM"
590 LOCATE 9,11:PRINT "spielten Sie "
600 PEN 2:LOCATE 3,13:PRINT satz$;
610 IF differenz<>0 THEN PRINT ABS(differenz);" DM."
620 PEN 0:LOCATE 1,20:END
630 REM *** Blumenzeichnen ***
640 PAPER #1,10:CLS #1
650 PRINT #1,CHR$(22);CHR$(1);
660 PEN #1,farbe(i)
670 LOCATE #1,2,2
680 PRINT #1,CHR$(200);CHR$(201)
690 LOCATE #1,2,3
700 PRINT #1,CHR$(202);CHR$(203)
710 PEN #1,0
720 LOCATE #1,2,2
730 PRINT #1,CHR$(204);CHR$(205)
740 LOCATE #1,2,3
750 PRINT #1,CHR$(206);CHR$(207)
760 PEN #1,9
770 FOR j=4 TO 7
780 LOCATE #1,2,j
790 PRINT #1,CHR$(j*2+200);CHR$(j*2+201)
800 NEXT j
810 PRINT #1,CHR$(22);CHR$(0);
820 RETURN
830 REM *** Titelbild ***
840 MODE 1
850 BORDER 21

860 INK 0,21:INK 1,6:INK 2,3:INK 3,21,4
870 PEN 1:LOCATE 12,1:PRINT "BLUMENSPIEL "
880 PEN 3:LOCATE 12,2:PRINT "* * * * * "
890 PEN 2:LOCATE 1,5:PRINT "Dieses Spiel simuliert einen Spielautomaten mit bunten Blumen."
900 LOCATE 1,8:PRINT "Gewonnen haben Sie , wenn 2 oder gar alle 3 Blumen die gleiche Farbe haben."
910 LOCATE 1,11:PRINT "START und STOP so wie WEITERSPIELEN werden durch alle Tasten ausgelöst."
920 LOCATE 1,20:INPUT "Bitte Ihr Einsatz (1 bis 20 DM) ";einsatz
930 IF einsatz<1 OR einsatz>20 THEN LOCATE 10,24:PEN 3:PRINT "Das geht nicht !!"
FOR zeit=1 TO 1000:NEXT zeit:LOCATE 1,24:PRINT STRING$(40," ");PEN 2:GOTO 920
940 bestand=einsatz
950 RANDOMIZE TIME
960 REM *** Bildschirmaufbau ***
970 MODE 0
980 BORDER 3
990 INK 0,3:INK 1,24:INK 2,1:INK 3,11:INK 4,16:INK 5,15:INK 6,6:INK 7,17:INK 8,26:INK 9,9:INK 10,19:INK 11,13,26:INK 12,0:INK 13,0,6:INK 14,0,4
1000 WINDOW #1,2,5,10,16:PAPER #1,10
1010 WINDOW #2,9,12,10,16:PAPER #2,10
1020 WINDOW #3,16,19,10,16:PAPER #3,10
1030 WINDOW #4,2,0,5,7:PAPER #4,12:PEN #4,13
1040 WINDOW #5,13,19,5,7:PAPER #5,12:PEN #5,13
1050 WINDOW #6,3,18,1,3:PAPER #6,4:PEN #6,14
1060 WINDOW #7,3,18,19,25:PAPER #7,5:PEN #7,2
1070 FOR i=1 TO 7:CLS #i:NEXT i
1080 LOCATE #7,2,6:PRINT #7,"Bestand:";LOCATE #7,14,6:PRINT #7,"DM":LOCATE #7,10,6:PRINT #7,USING "###";bestand
1090 SYMBOL AFTER 199
1100 SYMBOL 200,3,15,63,63,127,127,255,255
1110 SYMBOL 201,192,240,252,252,254,254,255,255
1120 SYMBOL 202,255,255,127,127,63,63,15,3
1130 SYMBOL 203,255,255,254,254,252,252,240,192
1140 SYMBOL 204,0,0,0,0,0,1,3,7
1150 SYMBOL 205,0,0,0,0,0,120,192,224
1160 SYMBOL 206,7,3,1,0,0,0,0
1170 SYMBOL 207,224,192,120,0,0,0,0
1180 SYMBOL 208,7,7,7,7,7,3,3,3
1190 SYMBOL 209,0,0,0,0,120,120,120,120
1200 SYMBOL 210,3,3,97,241,121,61,31,7
1210 SYMBOL 211,120,192,192,192,192,192,224,224
1220 SYMBOL 212,1,1,1,0,0,0,0
1230 SYMBOL 213,224,224,224,224,224,224,224,224,224
1240 SYMBOL 214,0,1,1,3,3,3,3,3
1250 SYMBOL 215,230,252,240,240,224,192,192,192
1260 RETURN

```

Jannek / Mossakowski

ROM-Listing CPC 464/664/6128



676 Seiten
Dieses Buch enthält in konzentrierter Form umfassende Informationen über den Aufbau Ihres CPC Computers. Um es optimal zu nutzen, sollte man mit dem Schneider-Basic vertraut sein und bereits erste Schritte in Maschinensprache unternommen haben. Zu jeder ROM-Routine sind die Übergabeparameter und die Rückmeldungen aufgeführt. Tabellen erleichtern das Auffinden einer Routine und der Systemvariablen. Auch die Funktionsweise der verschiedenen Chips und ihre Programmierung werden ausführlich erklärt.

Bestellnummer CPC 122

DM 64,-

Hans Lorenz Schneider

Arbeiten mit dem Schneider CPC 464



288 Seiten
Das Buch befaßt sich mit der BASIC-Programmierung des Schneider CPC. Dabei werden anhand ausführlicher Beispielprogramme viele wichtige Programmierregeln erläutert. Es werden ernsthafte Themen wie Sortieralgorithmen und Dateiverwaltung behandelt. Auch auf die Programmierung von Spielen wird eingegangen, wobei die Grafik- und Musik-Programmierung nicht zu kurz kommt. Das Buch wendet sich an Besitzer des Schneider CPC 464, die eigene BASIC-Programme erstellen und ihren Computer effektiver für Beruf und Hobby einsetzen möchten.

Bestellnummer CPC 103

DM 38,-

Englisch/Germer/Scheuse/Thrun

**CPC 464 Tips & Tricks
Eine Fundgrube für den CPC-Anwender**



263 Seiten
Rund um den CPC 464 viele Anregungen und wichtige Hilfen. Von Hardwareaufbau, Betriebssystem, BASIC-Tokens, Zeichnen mit dem Joystick, Anwendungen der Windowtechnologie und sehr vielen interessanten Programmen wie einer umfangreichen Dateiverwaltung, Soundeditor, komfortablem Zeichengenerator bis zu kompletten Listings spannender Spiele bietet das Buch eine Fülle von Möglichkeiten.

Bestellnummer CPC 108

DM 49,-

Hans Lorenz Schneider

Das Schneider CPC Grafikbuch



336 Seiten
Der Autor behandelt zunächst die einzelnen Grafik-Modi, die Grafik-Befehle und die wichtigsten Betriebssystem-Routinen. Es folgen universell einsetzbare Basic-Unterprogramme zum Zeichnen bestimmter Figuren (z. B. Quader, Kreise, Ellipsen). Auch der Geschäftsgrafik ist ein Kapitel gewidmet, in dem die Programmierung von Diagrammen (Liniens-, Balken-, Torten- und Säulendiagramme) vorgeführt wird. Diejenigen Leser, die an Animation interessiert sind, finden ein eigenes Kapitel, das sich mit der Erzeugung von Sprite-Grafik befaßt. Eine Hardcopy-Routine, die die Ausgabe der erstellten Grafiken auf einen Drucker ermöglicht, rundet das Buch ab.

Bestellnummer CPC 11

DM 48,-

Siegmar Wittig

BASIC-Brevier für den Schneider CPC 464



224 Seiten
Dieses Buch zeigt, wie man auf dem Schneider CPC in Basic programmiert. Auch ohne Vorkenntnisse kann jeder nach kurzer Zeit seine eigenen Programme schreiben. Zahlreiche Aufgaben und Programmbeispiele tragen dazu bei, das Wissen zu festigen. Hier findet man fast alle Probleme wieder, die sich einem "Einsteiger" mit dem CPC 464 stellen. Dieses Buch würde sogar das Handbuch des Herstellers voll ersetzen.

Bestellnummer CPC 104

DM 29,80

C. Sträusch/H. Pick

CPC 464 für Ein- und Umsteiger



260 Seiten
Über die hervorragende Qualität des CPC 464 sind sich alle Experten einig; die mitgelieferte Dokumentation läßt jedoch eine Menge Fragen offen. Der Computermüll, ebenso aber auch der "alte Hase", der bereits ein anderes System kennt, benötigt hier Hilfe, die ihm dieses Buch gibt. Eine BASIC-Einführung (an den Beispielen Dateiverwaltung und Textverarbeitung) ist selbstverständlich. Darüber hinaus konzentrieren sich die Autoren jedoch auf die besonderen Eigenschaften des 464ers, seine mächtigen Befehle, die andere BASICs ganz schön alt aussehen lassen und seine begeisterten Grafik- und Musikmöglichkeiten.

Bestellnummer CPC 109

DM 46,-

Manfred Walter Thoma

CPC 464/664 Praxis Band 1: Schwerpunkt Grafik



192 Seiten
Der erste Band des zweibändigen Werkes beschäftigt sich schwerpunktmäßig mit den Grafikmöglichkeiten des Schneider CPC-464/664. Es ist besonders darauf geachtet worden, daß alle behandelten Probleme und Programmier-techniken einen konkreten Bezug haben. Neben unzähligen Beispielen findet der Benutzer viele Programme und Unterprogramme, die seine Arbeit mit dem Schneider CPC-464/664 erheblich erleichtern. Gezeigt werden Programme von der Multicolorgrafik bis hin zum kompletten CAD-System, vom Tortendiagramm bis zum 3D-Funktionsplot, von der 3-dimensionalen Wertdarstellung bis zur Analoguhr.

Bestellnummer CPC 102

DM 34,-

Peter Heiß

Z80-Maschinensprachekurs für den CPC 464/664/6128



194 Seiten
Schon im CPC-Magazin 6/86 haben wir diesen wirklich guten Kurs vorgestellt. Er wendet sich an alle Benutzer der CPCs 464/664/6128, die bereits über Basic-Kenntnisse verfügen und nun in die Maschinenspracheprogrammierung einsteigen wollen. Die Befehle des Z80-Prozessors werden anhand kleiner Beispielprogramme erklärt. Die Anpassungen für den 664/6128 sind jeweils angegeben. Das Buch enthält eine Tabelle aller Z80-Befehle und einen einfachen Direktassembler, der auch auf Cassette bezogen werden kann.

Bestellnummer CPC 111

DM 34,-

Chaos Computer Club

Die Hackerbibel



259 Seiten
Überall in den Medien ist seit geraumer Zeit Rede von den Hackern. Und wenn man "Hacker" sagt, meint man in diesem unserm Lande zumeist Mitglieder des Chaos Computer Club Hamburg und seine Ableger. Sei es der 130 000 DM Coup bei der Hamburger Sparkasse oder die diversen Auftritte in Tagesschau, bei Frank Elstner und anderswo: über sie ist viel berichtet worden. Hier erstmals ein Buch von ihnen. Nicht nur über das "Wie" des Hackens, sondern auch über das "Warum".

Bestellnummer H 500

DM 33,33

Dieter Winkler

Das Schneider CPC 6128/664 Praxisbuch



240 Seiten
Dieses Buch eignet sich besonders als Ratgeber für alle diejenigen, die sich einen ersten Überblick über die neuen Geräte, deren Peripherie, Zubehör und Software verschaffen wollen. Es ist für Einsteiger und Profis gleichermaßen interessant und hilft mit, daß die Arbeit am Computer gleich von Anfang an erfolgreich wird.

Bestellnummer CPC 110

DM 29,80

BUCH-BESTELLSCHEIN

Bitte liefern Sie mir folgende Bücher:

Anzahl	Bestell-Nr.	Titel	Einzel-Preis Inkl. MwSt.

Name des Bestellers

Anschrift

PLZ/Ort

Telefon

Ich wünsche folgende Bezahlung:

- Nachnahme (+ 5,70 DM Porto + Versandkosten)
- Vorkasse (keine Versandkosten)

Bei Vorkasse bitte Scheck bellegen oder auf Postscheckkonto Karlsruhe 43423-756 überweisen.

Datum/Unterschrift

Coupon ausschneiden, auf Postkarte kleben und einsenden: Verlag Rätz-Eberle, Postfach 1640, 7518 Bretten.

Wie kopiert man geschützt abgespeicherte Programme?

Die Schutzmethoden werden immer schwieriger und diffiziler

Im letzten Heft haben wir ja schon besprochen, wie ungeschützte bzw. nicht besonders gut geschützte Programme auf Diskette zu überspielen sind. Diesmal wollen wir uns an etwas größere Probleme wagen. Gerade in letzter Zeit passiert es immer häufiger, daß Programme nicht mehr im Standardformat (2-KByte-Blöcke) abgespeichert werden, sondern in allen möglichen Varianten, z.B. ohne Header, in einem Block, in 4-KByte-Blöcken etc. Diese Programme beginnen immer ähnlich. Zuerst wird ein kurzes Basic-Programm geladen. Dieses erstellt eine kurze Titelfgrafik und lädt dann ein extrem kurzes Maschinenprogramm nach. Bis hierher lassen sich die Programme noch einwandfrei kopieren. Dieses Maschinenprogramm lädt dann allerdings andere Programmteile nach, die sich so nicht kopieren lassen. Manchmal wird vorher noch ein Titelbild geladen, manchmal geschieht dies allerdings auch schon durch den MC-Loader.

So gehen wir vor

1. Programmanalyse

Zuerst sollte man das Programm einmal genau analysieren, es also laden und dabei notieren, wie viele Blöcke das erste Basic-Programm, der Screen und das Hauptprogramm haben. Ferner sollten alle im normalen Format abgespeicherten Programme mittels eines Examiners (siehe Teil 10 der Serie) genauestens untersucht werden. All dies erleichtert das Verständnis für den Maschinencode-Loader.

2. Überspielen des Basic-Programms

Meistens beginnen die geschützten Programme ja mit einem kurzen Basic-Programm. Dieses laden Sie bitte. Eventuell müssen Sie es noch mit einem Protector entschützen. Wenn gewünscht, können Sie im Basic-Programm gleich Ihr Signum einbauen. Danach kann das Basic-Programm auf Diskette abgespeichert werden. Da dieses allerdings nur den Loader nachlädt, muß es nachher noch geändert werden, denn es soll ja nicht den Loader, sondern alle Programmteile nachladen. Sollte direkt ein Maschinenprogramm geladen werden, so müssen Sie nachher ein neues

Basic-Ladeprogramm schreiben, welches die anderen Teile nachlädt.

3. Überspielen des Screens

Oft wird an dieser Stelle zunächst einmal ein Bildschirm geladen, den Sie ja einfach auf Diskette überspielen können. Manchmal ist der Screen auch als MC-Programm abgelegt. Aus der Programmanalyse wissen Sie ganz genau, ob und, wenn ja, wann ein Bildschirm folgt. Eventuell wird der Screen aber auch erst vom Loader und dann schon im speziellen Format geladen.

4. Laden, Untersuchen des Maschinencodes und anschließendes Überspielen der verschiedenen Programmteile

Dieser Punkt ist sicherlich mit Abstand der schwierigste. Zuerst einmal stellen Sie Ladeadresse und Aufrufadresse des Loaders fest. Diese kann man aus dem Basic-Programm mit Hilfe des Examiners erfahren. Anschließend laden Sie den MC-Loader. Nun benötigen Sie noch unbedingt einen Disassembler. Gut ist "Mona" aus dem Devpac-Paket von Schneider. Dieser Disassembler kann an jede beliebige Speicherstelle geladen werden, allerdings so, daß er den Loader nicht überschreibt. Anschließend kann von der Aufrufadresse die Programmausführung nachvollzogen werden. Ob Sie das per Disassembler oder per Single-Step tun, bleibt Ihnen überlassen.

Oft wird versucht, die Loader möglichst kompliziert zu gestalten, um den Knacker zu verwirren und die eigentliche Programmfunktion zu verschleiern. Da wird viel gesprungen, es werden zum Teil völlig sinnlose und dann wieder absolut wichtige Unterprogramme aufgerufen. Durch einen solchen Wust findet man sich am besten mit Hilfe eines Ausdrucks hindurch. Manchmal allerdings liegt das Loader-MC-Programm gar nicht im richtigen Zustand vor. So werden häufig Programmteile des Loaders von anderen Programmteilen invertiert oder verschoben oder sonst irgendwie verändert. Bevor Sie Ihr MC-Programm richtig anschauen können, müssen Sie erst diese Programmteile abarbeiten lassen. Sie können ja Breakpunkte setzen und dann die entsprechenden

Programmteile einfach aufrufen. Befindet sich der Loader im richtigen Format, sollte man ihn zur Sicherheit noch einmal abspeichern (gleich auf Diskette).

Nun kann das Programm richtig nachvollzogen werden. Dabei ist natürlich auf CALLs zu achten, die in die Cassettenverwaltung springen, um z.B. headerlose Files zu laden. Eine Auflistung der Einsprünge zur Cassettenverwaltung erfolgt am Ende dieses Teils. Nach dem Laden muß dann allerdings dafür gesorgt sein, daß ins Basic zurückgesprungen wird, damit der Benutzer das Programm normal auf Diskette abspeichern kann. Nachdem für dies gesorgt ist, kann der Loader abgespeichert und der Disassembler verlassen werden. Sie laden nun den veränderten Loader und rufen ihn mit CALL an der richtigen Stelle auf. Dann müßte der entsprechende Programmteil geladen werden. Achten Sie darauf, daß das Band richtig gespult ist. Danach können Sie das gerade geladene Programm auf Diskette ablegen. Dies machen Sie entsprechend mit allen Programmteilen.

Auf folgendes sollten Sie allerdings noch achten: Manchmal werden die Programmteile wie oben beschrieben verändert, d.h. invertiert o.ä. Dann dürfen Sie nicht direkt nach dem Laden ins Basic zurückspringen, sondern müssen die Veränderungsroutine mitdurchlaufen. Somit stellen Sie sicher, daß die richtige Version des Programms auf Diskette abgelegt wird.

Aufrufadresse feststellen

Damit das Maschinenprogramm vom neuen Basic-Loader richtig aufgerufen werden kann, muß festgestellt werden, an welche Adresse der Loader nach Beendigung seiner Aufgabe springt.

Basic-Programm modifizieren

Sie müssen das Basic-Programm so umgestalten, daß die entsprechenden Files richtig geladen werden. Auch der richtige CALL zum Aufruf ist einzubauen.

Einige Probleme

1. MC-Loader wird in Basic-Bereich geladen

Sie müssen den Loader dann an eine andere Adresse laden. Bei CALLs und JPs muß aber die richtige Adresse berechnet werden, zu der gesprungen wird. Deshalb sollte sich das Programm nicht im Single-Step-Modus befinden. Wird ein Teil des Loaders ausgeführt (z.B. beim Laden), müssen die Adressen vorher geändert werden. Beachten Sie, daß der Loader sich nicht im Bereich des zu ladenden Files befindet.

2. Das zu ladende Programm ist zu lang

Sie können dann den Cassettenpuffer in den Bildschirmspeicher legen. Eine genaue Erklärung dieses Prinzips steht in den Z80-Assemblertips, CPC-Magazin 10/86.

3. MC-Programm im Diskettenbereich

Oftmals liegen Maschinenprogramme in dem vom Diskettenlaufwerk belegten Bereich oberhalb von 42619. Die Maschinenprogramme müssen dann an eine andere Adresse geladen werden, wobei die Ladeadresse im MC-Loader zu ändern ist. Außerdem muß ein Verschiebeprogramm geschrieben werden. Dieses Verfahren ist ebenfalls im CPC-Magazin 10/86 bei den Assemblertips erklärt.

Vielleicht sind Sie jetzt enttäuscht, daß ich Ihnen keine detaillierteren Anweisungen geben kann, aber es gibt eine Unzahl von verschiedenen Methoden, Programme zu schützen, so daß eine allgemeine Darstellung unumgänglich war. Sie sollten aber auf jeden Fall versuchen; zuerst einfachere Programme zu knacken, die vielleicht noch im normalen Format abgespeichert sind, und sich dann Schritt für Schritt an schwierigere Programme wagen. Und verzweifeln Sie nicht, wenn Sie mal ein Programm nicht knacken können, versuchen Sie es einfach beim nächsten. Als Faustregel gilt: Je besser und neuer das Programm ist, desto schwieriger ist der Kopierschutz zu überwinden.

Die Einsprünge der Cassettenverwaltung

Wichtig zum Verständnis des Loaders ist die genaue Kenntnis der genauen Funktion aller Einsprünge des Cassettenbetriebssystems. Deshalb haben wir diese hier aufgeführt. Zum Format dieser Einsprünge. Zuerst wird der Standardname der Routine angegeben, daneben die Einsprungsadresse. Die Ein- und Aussprungsbedingungen folgen anschließend. Schließlich wird die Funktion des Einsprungs genauestens beschrieben.

01. CASINITIALISE #BC65

Input: -
Output: HL, DE, BC, AF unbestimmt

Funktion: Offene Ein-/Ausgabedateien werden geschlossen, allerdings werden offene Ausgabedateien nicht mehr abgespeichert. Die Schreibgeschwindigkeit wird auf 1000 Baud gesetzt. Cassettenmeldungen werden zugelassen (siehe Einsprung 3).

02. CASSET SPEED #BC68

Input: HL: Länge für ein halbes Bit
A : Vorprüflänge

Output: HL, AF unbestimmt

Funktion: Wie Sie vielleicht wissen, ist die Angabe der Baudrate immer nur überschlagsmäßig möglich, da die genaue Geschwindigkeit davon abhängt, wieviel Bits gesetzt sind. Ein gesetztes Bit benötigt zur Abspeicherung bzw. zum Einlesen die doppelte Zeit wie ein zurückgesetztes. Baud bedeutet Bit pro Sekunde, d.h., bei 1000 Baud werden 1000 Bits oder 125 Bytes pro Sekunde eingelesen (bei gleicher Verteilung von 0 und 1). Die Standardwerte sind folgende:

Baudrate	Vorprüflänge	Länge eines halben Nullbits
1000	25ms	333ms
2000	50ms	167ms

Durch folgende Beziehungen können aus der Baudrate die beiden erforderlichen Parameter berechnet werden:

Vorprüf = 25ms x Baudrate
Halbe Nullbit - Länge = 333 / Baudrate

Die Baudrate sollte zwischen 2500 und 700 liegen, sonst kommt es sehr leicht zu Lese- oder Schreibfehlern.

03. CASNOISY #BC6B

Input: A=0 Meldungen werden zugelassen
A<>0 Meldungen werden gesperrt

Output: AF verändert

Funktion: Durch ein vorangestelltes Ausrufezeichen im Namen werden Cassettenmeldungen unterdrückt. Dies ist die entsprechende MC-Routine dazu. Folgende Meldungen werden bei A<>0 unterdrückt:

- Press PLAY then any key
- Press REC and PLAY then any key
- Loading "Name" block
- Saving "Name" block
- Found "Name" block

04. CASSTART MOTOR #BC6E

Input: -
Output: Carryflag gesetzt: alles o.k.
Carryflag nicht gesetzt: ESC wurde gedrückt
A: vorheriger Motorzustand
Sonstige Flags verändert

Funktion: Der Motor wird angeschaltet. Falls er nicht schon läuft, wartet die Routine ca. 2 Sekunden bis zum Erreichen der Nenn-drehzahl. Wird ESC gedrückt, läuft der Motor trotzdem weiter. Der alte Motorzustand kann mit Hilfe des Einsprungs 6 verwertet werden.

05. CASSTOP MOTOR #BC71

Input: -
Output: siehe 4.
Funktion: Der Cassettenmotor wird abgeschaltet.

06. CASRESTORE MOTOR #BC74

Input: A: alter Motorzustand
Output: siehe 4. und 5., nur AF ist verändert

Funktion: Der in A gespeicherte alte Motorzustand wird wiederhergestellt.

07. CASIN OPEN #BC77

Input: HL: Adresse des Dateinamens
DE: Adresse des Cassettenbuffers
B : Länge des Dateinamens

Output: Wenn die Datei ordnungsgemäß geöffnet werden konnte:
Carryflag gesetzt
Zeroflag zurückgesetzt
HL: Adresse des Cassettenvorspanns
DE: Ursprüngliche Adresse der Dateidaten
BC: Länge der gesamten Datei in Bytes
A : Dateiart
Flags und IX verändert

Wenn die Datei bereits eröffnet war:

Carryflag zurückgesetzt
Zeroflag zurückgesetzt
HL, DE, BC, AF, IX verändert
Wenn ESC gedrückt wurde:
Carryflag zurückgesetzt
Zeroflag gesetzt
HL, DE, BC, AF, IX verändert

Funktion: Eine Datei wird zum Einlesen eröffnet, der erste Block automatisch in den Pufferbereich gelesen und der Dateiname automatisch auf 16 Zeichen gekürzt. Ist er kürzer, wird er auf 16 Zeichen mit Null-Bytes

erweitert. Kleinbuchstaben werden in Großbuchstaben umgewandelt. Der Dateiname darf überall im RAM stehen, auch unter den ROMs, genauso der Puffer. Der angegebene Puffer bleibt aktiv, bis die Datei geschlossen oder beim Laden abgebrochen wird.

Bei den Dateiararten gibt es folgende Versionen:

Bit 0: Wenn gesetzt, dann ist das Programm geschützt.

Bit 1-3: 0 Basic
1 binär (MC)
2 Bildschirm (Screen)
3 ASCII

Bit 4-7: Version (ASCII = 1, sonst 0)

08. CASIN CLOSE #BC7A

Input: -

Output: Wenn Datei ordnungsgemäß geschlossen wurde:
Carryflag gesetzt
HL, DE, BC, AF verändert

Wenn Datei überhaupt nicht geöffnet war:
Carryflag zurückgesetzt
HL, DE, BC, AF verändert

Funktion: Die eröffnete Datei und der dazugehörige Puffer werden geschlossen.

09. CASIN ABANDON #BC7D

Input: -

Output: HL, DE, BC, AF verändert

Funktion: Diese Routine sollte im Falle eines Fehlers aufgerufen werden. Der Cassettenbuffer und die Eingabedatei werden auf jeden Fall geschlossen.

10. CASIN CHAR #BC80

Input: -

Output: Wenn das Zeichen ordnungsgemäß gelesen wurde:
Carryflag gesetzt
Zeroflag zurückgesetzt
A : gelesenes Zeichen
Flags, IX verändert

Wenn Dateiende erreicht:
Carryflag zurückgesetzt
Zeroflag zurückgesetzt
AF, IX verändert

Wenn ESC gedrückt wurde:
Carryflag zurückgesetzt
Zeroflag gesetzt
AF, IX verändert

Funktion: Von einer eröffneten Datei wird ein Zeichen gelesen, wenn nicht das Dateiende erreicht oder ESC gedrückt wurde. Falls nötig, wird ein neuer Block in den Puffer geladen.

Wird erst einmal ein Zeichen auf diese Art gelesen, so muß auch die ganze Datei so gelesen werden. Es ist nicht möglich, nachher noch auf CAS IN DIRECT (Einsprung 11) umzusteigen.

11. CASIN DIRECT #BC83

Input: HL: Adresse, wo der Block hingeladen wird

Output: Wenn die Datei ordnungsgemäß geladen wurde:

Carryflag gesetzt
Zeroflag zurückgesetzt
DE, BC, AF, IX verändert

Wenn die Datei nicht eröffnet war:

Carryflag zurückgesetzt
Zeroflag zurückgesetzt
HL, DE, BC, AF, IX verändert

Wenn ESC gedrückt wurde:
Carryflag zurückgesetzt
Zeroflag gesetzt
HL, DE, BC, AF, IX verändert

Funktion: Die gesamte Datei wird auf einmal ab der Adresse, die durch HL spezifiziert wird, in den Speicher geladen. Sie muß dabei bereits eröffnet sein. Der Puffer wird in den Speicher kopiert, und die restlichen Blöcke werden direkt an die richtige Stelle gelesen.

12. CAS RETURN #BC86

Input: -

Output: -

Funktion: Das zuletzt per CAS IN CHAR gelesene Zeichen wird wieder in den Puffer geschrieben und kann von dort erneut gelesen werden. Es muß mindestens ein Zeichen gelesen worden sein. Die Routine darf nur einmal aufgerufen werden, bis wieder ein neues Zeichen per CAS IN CHAR gelesen wird. Die Routine kann also wirklich ausschließlich das letzte Zeichen zurückschreiben.

13. CASTEST EOF #BC89

Input: -

Output: Wenn das Dateiende nicht erreicht wurde:

Carryflag gesetzt
Zeroflag zurückgesetzt
A, IX, Flags verändert

Wenn das Dateiende erreicht wurde:

Carryflag zurückgesetzt
Zeroflag zurückgesetzt
A, IX, Flags verändert

Wenn ESC gedrückt wurde:
Carryflag zurückgesetzt

Zeroflag gesetzt
A, IX, Flags verändert

Funktion: Es wird getestet, ob das Dateiende erreicht ist; ohne ein Zeichen zu lesen. Dennoch wird automatisch auf die zeichenweise Abarbeitung der Datei umgeschaltet. Das Lesen per CAS IN DIRECT ist danach nicht mehr möglich. Ein Aufruf von CAS RETURN ist erst nach dem neuen Lesen eines Zeichens zugelassen.

14. CAS READ #BCA1

Input: HL: Adresse, wohin der Satz geladen wird ¹⁰⁰

DE: Länge des Satzes ⁶⁹⁰

A : Satztyp [?]

Output: Wenn der Satz ordnungsgemäß geladen wurde:

Carryflag gesetzt
HL, DE, BC, IX, AF verändert

Wenn ESC gedrückt wurde oder ein Fehler auftrat:
Carryflag zurückgesetzt
A : Fehlerart
HL, DE, BC, IX und Flags verändert

Funktion: Dies ist wohl die wichtigste Routine. Mit ihr können headerlose Daten von beliebiger Länge am Stück geladen werden. Auch ist es möglich, einen Vorspann (Satztyp #2C) oder einen Datensatz (Satztyp #16) zu laden. Es können auch weniger Bytes, als abgespeichert sind, geladen werden. Allerdings nicht mehr; dies erzeugt einen Fehler (normalerweise Fehlertyp 1). Eine Länge von 0 wird als 65536 interpretiert.

Fehlerart Bedeutung

0	ESC wurde gedrückt
1	Baudrate zu hoch; Routine konnte Bit nicht schnell genug lesen
2	Lesefehler

Damit wären alle Cassetten-Einlese-routinen vorgestellt. Achten Sie vor allem auf die letzte, sie wird sehr oft benutzt.

Im nächsten Heft...

Im nächsten Heft wird wie angekündigt erklärt, wie man Unsterblichkeitspokes finden kann. Außerdem werden die Routinen zur Cassettenabspeicherung vorgestellt, nachdem wir in dieser Folge schon die zum Einlesen behandelt haben. So hat der Benutzer alle Cassetteinsprünge zur Hand.

Andreas Zallmann

»LOAD«

PRÄSENTIERT:

**Eine echte Überraschung für alle
Computer-Freaks:**

Die »LOAD«-SUPER-INFO

Mehr als ein Katalog: Info-Broschüre mit Super-Hard- und Software-Angeboten, aktuellen und brandneuen Programmen mit ausführlichen Inhaltsbeschreibungen, allgemeinen Tips & Tricks, Problemlösungen und..und..und...

SOFORT ANFORDERN!

Schutzgebühr
2,- DM in
Briefmarken

GEWINNEN SIE MIT »LOAD«:

Jeder 50. Einsender erhält einen
EINKAUFSGUTSCHEIN im Wert von **40,- DM**
für Software aus einer Liste von ca. 40 Programmen.

Auch das noch: Unter allen Einsendern,
die bis zum 30.11.86 unsere INFO angefordert
haben, verlosen wir einen

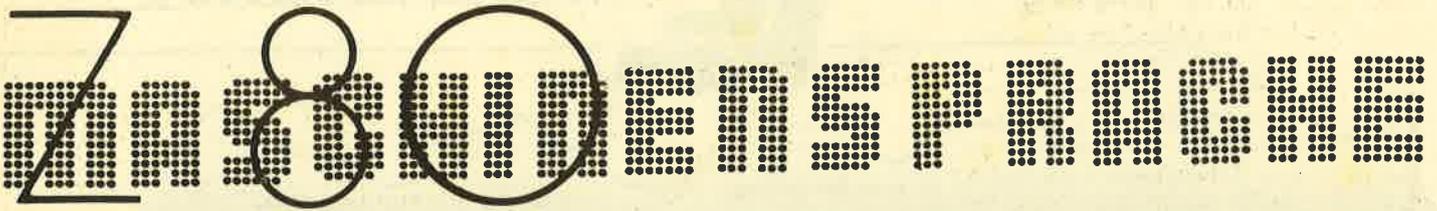
Schneider PC 1512

Der Rechtsweg ist
ausgeschlossen.

Systeme
**SCHNEIDER
COMMODORE**

»LOAD«

Waltraud Kühn KG
Postfach 1225
4620 Castrop-Rauxel



Teil 11

Wie schon in der letzten Folge angekündigt, wollen wir diesmal die Blockvergleichsbefehle besprechen und mit deren Hilfe ein Stringsuchmaschinenprogramm erstellen. Ferner werden noch diverse Sonder- und Spezialbefehle vorgestellt. Da im weiteren noch einige Befehlstabellen angegeben sind, hier kurz deren Aufbau: Zuerst folgt der Befehl im Standard-Z80-Mnemonic, anschließend die Opcodes. Dabei steht nn für eine Ein-Byte-Zahl, die der Benutzer frei wählen kann. Anschließend folgt, welche Flags beeinflusst werden. Steht dort das Kürzel des Flags, so heißt dies, daß das Flag entsprechend der Operation logisch gesetzt wird. (Z.B. ist das Zeroflag gesetzt, wenn bei einer Operation eine Null entstanden ist.) Steht nichts unter dem Flag, so wird das Flag durch den Befehl in keinsten Weise beeinflusst. Ein Fragezeichen bedeutet, daß der Zustand des Flags nach der Befehlsausführung unbestimmt ist, eine 1, daß das Flag in jedem Fall gesetzt ist, und eine Null, daß das Flag in jedem Fall zurückgesetzt ist. Die Kürzel haben folgende Bedeutung: Z = Zeroflag, S = Signumflag (Vorzeichenflag), P = Parityflag (gerade/ungerade) und C = Carryflag (Übertragsflag).

Die Blockvergleichsbefehle

Auch von den Blockvergleichsbefehlen gibt es vier Stück.

1. CPI

Dieser Befehl vergleicht den Akkumulator mit dem Byte, auf das HL zeigt. HL wird anschließend erhöht und BC, das hier wieder als Zähler dient, erniedrigt. Die Flags werden wie folgt von diesem Befehl beeinflusst: Das Carryflag überhaupt nicht, und das Signumflag wird entsprechend der Differenz A-(HL) gesetzt. Diese Differenz wird zum Vergleich ja immer gebildet. A enthält nach der Operation aber nicht etwa diesen, sondern den alten Wert. Die Differenz wird zwar berechnet, aber nirgends abgelegt. Das Zeroflag wird gesetzt, wenn A gleich dem Inhalt der Speicherzelle ist, auf die HL zeigt, ansonsten ist das Zeroflag zurückgesetzt. Das Parityflag gibt wieder den Status des Zählers an. Bei gesetztem Flag ist nach der Ausführung

des Befehls das Register BC ungleich Null. Ist BC nach Ausführung des Befehls gleich Null, so ist das Flag zurückgesetzt.

2. CPD

Der CPD-Befehl unterscheidet sich vom CPI-Befehl nur insofern, daß er den Datenzeiger (das HL-Register) nicht erhöht, sondern erniedrigt. Auch die Flags werden in gleicher Weise beeinflusst.

3. CPIR

Dieser Befehl macht zunächst einmal dasselbe wie der CPI-Befehl. Allerdings setzt er die Ausführung solange fort, bis entweder A gleich dem Inhalt der Speicherzelle ist, auf die HL zeigt, oder der Zähler, also das BC-Register, den Wert Null erreicht. Mit diesem Befehl kann man Speicherbereiche sehr gut auf das Vorkommen eines Bytes untersuchen. Der Zustand der Flags ist dem Zustand nach dem CPI-Befehl ähnlich: Carry wird nicht beeinflusst, Signum nach der letzten Differenz A-(HL), Zero ist gesetzt, wenn Abbruch durch A gleich (HL), Parity ist gesetzt, wenn $BC <> 0$. Parity ist gesetzt, wenn $BC = 0$.

4. CPDR

Der einzige Unterschied des CPDR-Befehls zum CPIR-Befehl besteht darin, daß HL nicht erhöht, sondern um eins vermindert wird. Die Flags werden genauso beeinflusst.

Die Erklärung der Blockvergleichsbefehle wäre damit abgeschlossen. Kommen wir nun zu unserem Stringsuchprogramm. An Parametern wird übergeben: Zuerst einmal der Stringdescriptor des Strings, der zu suchen ist, dann ab welcher Adresse wie viele Bytes durchsucht werden sollen. Diese 3 Parameter werden einfach an den CALL-Befehl angehängt. Dabei zeigt IX auf die Parameter, und A gibt deren Anzahl an.

Zurück gibt das Programm in der Adresse Back eine Meldung, ob die Suche erfolgreich war. Wurde der String gefunden, so steht in Back eine Eins, sonst eine Null. Wenn der String gefunden wurde, steht in Back 1 und Back 1+1 die Adresse, wo der String gefunden wurde.

Noch eine kurze Erklärung zum Stringdescriptor: Zu jedem String besteht ein drei Bytes langer Stringdescriptor. Im ersten ist die Länge des Strings gespeichert, im zweiten und dritten die Adresse des eigentlichen Strings. Die Adresse des Stringdescriptors läßt sich durch die Klammeraffenfunktion ermitteln. Stellt man dem String einen Klammeraffen voran, so wird als Funktionswert die Adresse des Stringdescriptors ausgegeben. Deshalb wird als erster Parameter dem String ein Klammeraffe vorangestellt.

Jetzt wieder zum Programm. Zuerst wird ein Flußdiagramm erstellt (Abb. 1), das später in ein Maschinenprogramm umgesetzt wird. Zunächst einmal ist zu kontrollieren, ob auch wirklich drei Parameter übergeben werden. Ist das nicht der Fall, so wird die Programmabarbeitung sofort abgebrochen. Anschließend werden die Parameter eingelesen: In BC die Länge, in HL die Startadresse und in IX der Stringdescriptor. In Back wird nun eine Null geschrieben und damit "nicht gefunden" gesetzt. Wird der String doch noch gefunden, muß diese Variable auf eins gesetzt werden, wenn nicht, kann einfach mit RET ins Basic zurückgekehrt werden.

Der erste Buchstabe wird aus dem Suchstring ausgelesen und danach wird mittels CPIR versucht, ihn im festgelegten Bereich zu finden. Nun gibt es zwei mögliche Abbruchbedingungen: Entweder wird der Buchstabe gefunden oder der Bereich erfolglos durchsucht. Ist letzteres der Fall (Zeroflag zurückgesetzt), dann wird die Bearbeitung abgebrochen. (In Back ist ja schon die Null für "nicht gefunden" abgelegt.) Ist der Buchstabe gefunden, wird seine Adresse erst einmal abgelegt. Damit ist die eventuelle Stringadresse auch gleich richtig gesetzt. Ist der Zähler jetzt Null, so kann der Buchstabe nicht mehr vollständig im festgelegten Bereich sein; die Bearbeitung wird abgebrochen. Ansonsten werden die nächsten beiden Buchstaben verglichen. Sind diese nicht gleich, wird wieder versucht, den ersten Buchstaben zu finden. Sind sie gleich, wird so lange verglichen, bis der ganze String untersucht ist. Dazu dient ein an-

derer Zähler. Ist der ganze String gesucht, wird in Back eine 1 geschrieben, das Zeichen für den Benutzer, daß der String gefunden ist. Die Adresse wurde ja bereits abgelegt. Während des Vergleichs wird natürlich auch immer noch der allgemeine Zähler kontrolliert. Sollte der String nicht mehr vollständig im festgelegten Bereich stehen, wird die Bearbeitung abgebrochen; der String gilt als nicht gefunden.

Das Maschinenprogramm

Um mit dem Programm zu arbeiten, tippen Sie es bitte in einen Assembler ein. Es wurde mit dem Schneider-Assembler erstellt. Aber auch andere müßten nach kleinen syntaktischen Veränderungen dieses Programm verdauen. Das Programm startet ab Adresse 40000, kann aber natürlich auch an anderen Stellen im RAM betrieben werden. Um es zu aktivieren (Startadresse 40000), ist folgendes notwendig:

1. In einer Stringvariablen ist der Suchstring einzuspeichern.
2. CALL 40000, suchstring\$, Startadresse, Länge

Um z.B. den String "CPC-Magazin" von Adresse 40000 bis 40999 zu suchen, wäre folgende Befehlsfolge notwendig:

```
a$ = "CPC-Magazin"
CALL 40000, a$, 40000, 1000
```

Es ist übrigens genau auf Klein- und Großschreibung zu achten, da für den Computer kleine und große Buchstaben nicht gleich, sondern verschieden sind.

Viele Z80-Befehle wurden in diesem Kurs bereits erklärt. Was noch fehlt, sind keine Befehlsgruppen, sondern nur einige Spezialbefehle, die wir im folgenden und im nächsten Teil erläutern werden.

Der CPL-Befehl

Dieser Befehl (CPL = Complement Accu, komplementiere Akkumulator) invertiert Bit für Bit des Akkumulators. War ein Bit gesetzt, wird es zurückgesetzt, war es zurückgesetzt, wird es gesetzt. Das durch das Invertieren entstandene Ergebnis wirkt erstaunlicherweise auf keine Flags.

Der DAA-Befehl

Der DAA-Befehl ist ein überaus komplexer Befehl. DAA steht für Dezimalanpassung des Akkumulators. Zur Funktion des Befehls muß man ein wenig weiter ausholen. Oft werden Zahlen Ziffer für Ziffer abgelegt. Pro Ziffer wird dann ein Nibble belegt. Mit einem Nibble lassen sich allerdings nicht nur die Zahlen Null bis Neun darstellen, sondern es sind noch 6 Bytes frei. Diese Bytes dürfen allerdings nicht benutzt werden, da es nur zehn Ziffern gibt.

Stringsuche

```
;
; Stringsuchprogramm
;
; Aufruf: CALL 40000, string$, Startadresse, Länge
;
; Zurück: Back: 0 = Nicht gefunden
;           1 = Gefunden
;           Back1: Adresse des Strings im Speicher, falls gefunden
;
;
; cp 3 ; 3 Parameter ?
; ret nz ; Nein: Ende der Bearbeitung
;
; xor a ; A=0
; ld (back),a ; Nicht gefunden setzen
;
; ld c,(ix+0) ; Lowbyte Länge
; ld b,(ix+1) ; Highbyte Länge
; ld l,(ix+2) ; Lowbyte Start
; ld h,(ix+3) ; Highbyte Start
; ld e,(ix+4) ; Lowbyte Stringdescriptor
; ld d,(ix+5) ; Highbyte Stringdescriptor
; push de
; pop ix ; Stringdescriptor von DE in IX
;
loop: ld e,(ix+1) ; Lowbyte Startadresse String aus Descrip.
; ld d,(ix+2) ; Highbyte
; ld a,(de) ; Erster Buchstabe
;
; cpir ; Buchstaben suchen
; ret nz ; Nicht gefunden
;
; dec hl ; HL und BC korrigieren
; inc bc
;
; ld (back1),hl ; Adresse String ablegen
;
; ld a,(ix+0) ; Länge String
; dec a ; Ein Buchstabe schon untersucht
; ld (len),a ; Ablegen
; jr z,ja ; Nur ein Buchstabe: Gefunden
;
loop2: inc de ; Auf nächsten Buchstaben im Suchstring
; inc hl ; Auf nächsten Buchstaben im Speicher
;
; ld a,b
; or c ; Ende des zu durchsuchenen Bereiches?
; ret z ; Ja: Bearbeitung abbrechen
;
; ld a,(de) ; Buchstaben
; cp (hl) ; vergleichen
; jr nz,loop ; Ungleich: Erneut ersten Buchstaben finden
;
; ld a,(len)
; dec a
; ld (len),a ; Stringlänge-1
; jr nz,loop2 ; Solange String noch nicht zuende: Weiter
; ; vergleichen
;
; ld a,l
; ld (back),a ; String zuende: Gefunden
; ; String gefunden setzen
; ret ; Zurück
;
back: defs 1 ; Ein Byte Speicherplatz für Flag, ob
; ; String gefunden (1) oder nicht (0)
back1: defs 2 ; Zwei Bytes für Stringadresse im Speicher
len: defs 1 ; Zwischenspeicher für aktuelle String-
; ; länge
```

Beim Addieren oder Subtrahieren kann es aber leicht mal passieren, daß ein solcher Wert in einem Nibble steht. Der Befehl DAA dient dazu, den Akku so zu verändern, daß die Ziffern nachher wieder stimmen.

Die Funktion des Befehls ist derart komplex, deshalb die Tabelle: Zuerst wird der Wert des oberen Nibbles, dann

der des unteren angegeben. Beide Angaben erfolgen in hex. In der dritten Spalte ist aufgeführt, welcher Wert zu A addiert wird, damit der Wert wieder stimmt. Auch dieser Wert ist in hex abgedruckt.

Dazu sollte man noch einiges erklärend hinzufügen. Hat beispielsweise das untere Nibble den Wert D, dann muß

der Wert 6 addiert werden. Dadurch wird dann das untere Nibble korrigiert und das obere erhöht. Sollte dies dann ebenfalls den Bereich von 0-9 verlassen, dann muß das obere Nibble ebenfalls korrigiert werden.

In der vierten Spalte ist schließlich der Wert des Carryflags nach Ausführung angegeben. Die fünfte Spalte ist auch nicht unwichtig. Hier ist der Wert des N-Flags angegeben, das wir bis jetzt nicht erklärt haben. Außer bei diesem Befehl ist eine Anwendung aber auch relativ unwichtig. Der DAA-Befehl muß natürlich wissen, ob zuletzt addiert oder subtrahiert wurde. Bei einer Addition muß er auch 6 addieren, um zu korrigieren. Wurde allerdings subtrahiert, muß um 6 vermindert werden, wenn das Nibble den zulässigen Bereich von 0-9 verläßt. Es wird dann aber nicht subtrahiert, sondern um $256-6 = 250$ oder hex FA addiert, was denselben Effekt hat.

Das N-Flag zeigt dem Prozessor, ob zuletzt addiert oder subtrahiert wurde. Dieses wird bei allen Befehlen, die vermindern (SUB, SBC, DEC, NEG), gesetzt und bei allen Befehlen, die erhöhen (ADD, ADC, INC), zurückgesetzt. Der Programmierer muß aber aufpassen, daß auch bei einem DEC-B-Befehl, der nichts mit dieser Zahlendarstellung zu tun hat, sondern nur als Zähler dient, das N-Flag gesetzt wird, obwohl vorher vielleicht der Akku erhöht wurde. Auch bei Doppelregisteroperationen wird dieses Flag gesetzt oder zurückgesetzt (je nach Art). Ähnlich wird dies für $N=1$ gehandhabt, nur daß subtrahiert und nicht addiert wird.

Der NOP-Befehl

Dieser Befehl ist recht interessant: Er macht nichts! NOP steht für "No Operation" oder zu deutsch "keine Operation". Das einzige, was dieser Befehl macht, ist, ein bißchen Zeit und Speicherplatz zu verschwenden. Der Befehl hat übrigens den Opcode Null, d.h., wenn man mit CALL einen leeren (mit Nullen gefüllten) Speicherbereich anspringt, wird dieser ganze Bereich durchgegangen, ohne daß sich irgendwelche Veränderungen an Registern ergeben. Dies ändert sich natürlich, wenn der Prozessor bei der Programmabarbeitung auf Opcodes trifft, die ungleich Null sind. Diesen Befehl kann man höchstens zur Verzögerung benutzen, deshalb gebe ich hier einmal die Abarbeitungsdauer an: 1 Microsekunde.

Der NEG-Befehl

Dieser Befehl zieht den Akkumulator von 256 ab und speichert das Ergebnis

Tabelle 1: Blockvergleichs-Befehle

Befehl	Opcodes	Z	C	S	P	Kurzbeschreibung
CPI	EDA1	Z		S	P	Vergleich A-(HL), HL+1, BC-1
CPD	EDA9	Z		S	P	Vergleich A-(HL), HL-1, BC-1
CPIR	EDB1	Z		S	P	Vergleich A-(HL), HL+1, BC-1 bis BC=0 oder A=(HL)
CPDR	EDB9	Z		S	P	Vergleich A-(HL), HL-1, BC-1 bis BC=0 oder A=(HL)

Der Complement-Befehl

Befehl	Opcodes	Z	C	S	P	Kurzbeschreibung
CPL	2F					A-Register wird invertiert

Der DAA-Befehl

Befehl	Opcodes	Z	C	S	P	Kurzbeschreibung
DAA	27	Z	C	S	P	Dezimalanpassung des A-Registers

Tabelle 2: Arbeitsweise des DAA-Befehls

Wert des oberen Nibbels	Wert des unteren Nibbels	Zu A addiert	Carry-flag	N-Flag
0-9	0-9	00	0	0
0-8	A-F	06	0	0
0-9	A-F	66	1	0
A-F	0-9	60	1	0
A-F	A-F	66	1	0

Der NOP-Befehl

Befehl	Opcodes	Z	C	S	P	Kurzbeschreibung
NOP	00					Keine Operation

Der NEG-Befehl

Befehl	Opcodes	Z	C	S	P	Kurzbeschreibung
NEG	ED44	Z	C	S	P	A=256-A oder A=0-A

wieder in A ab. Er ist für die Zweierkomplementdarstellung von Zahlen gut zu verwenden. Gleichzeitig läßt sich mit diesem Befehl errechnen, wie viele Bytes es noch bis zu einem Übertrag, also 256 sind. Beispielsweise: NEG(4) = 252, NEG(200) = 56 usw. Dies kann unter Umständen nützlich sein. Zu beachten ist allerdings dabei, daß bei 0 nicht 256, sondern Null herauskommt. Aber im Ein-Byte-Format ist 256 ja gleich 0.

Welche Flags werden beeinflusst? Das Zeroflag, Signumflag und Parityflag werden wie üblich und entsprechend dem Ergebnis angesprochen. Das Carryflag gibt an, ob das Ergebnis vor der Operation Null war. Ist das der Fall, so wird das Carryflag gesetzt.

Im nächsten Heft...

Im nächsten Heft werden wir die letzten Z80-Befehle vorstellen und – wenn der Platz reicht – auch wieder ein kleines Programm erstellen.

Andreas Zallmann

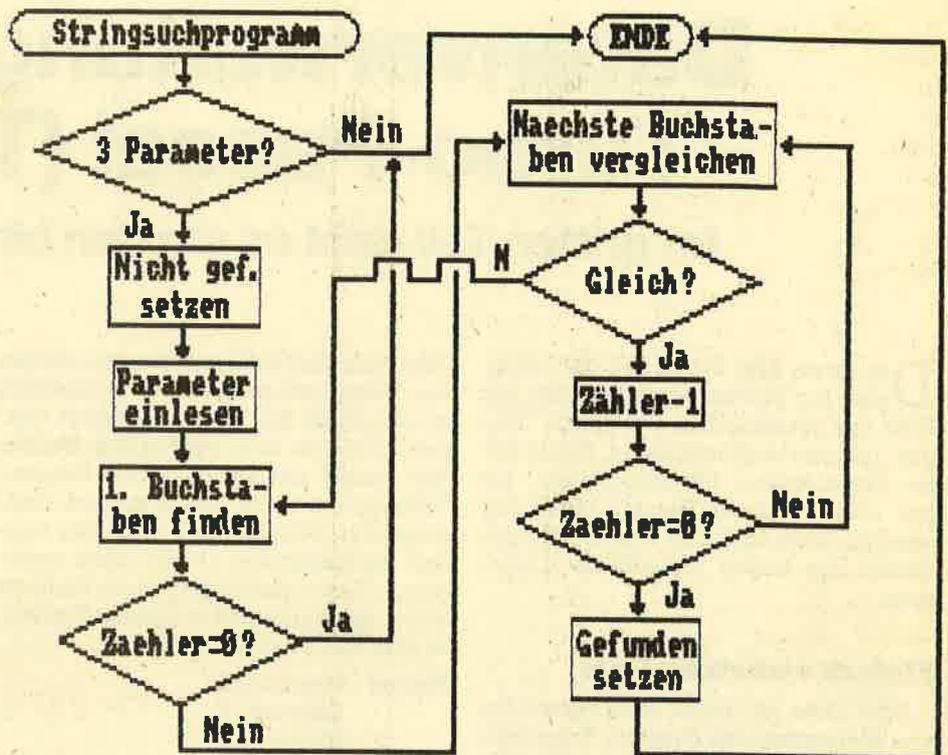


Abbildung 1: Flußdiagramm für das Stringsuch-Programm

DELTA COM
Freier-Homberts-Schmucker GbR
Computer Hard- und Software Vertrieb
An- und Verkauf von gebrauchter Hard- und Software

Kluge Leute arbeiten mit den Sicherungskopien ihrer Originalsoftware! Wollen Sie auch dazu gehören?? MASTER KEY schafft alles!!
MASTER KEY Disk-to-Disk-Kopierer 3" **98.-**
 Bestellungen bis 31.08.86: per V-scheck. **87.-**

Für vortex 3.5" und 5.25" in Vorbereitung.
AMSDISK Tape-to-Disk-Kopierer C **23.-**
 ● NEU ●

ZORGOS deutsches Grafikadventure mit einem speziell für die deutsche Grammatik entwickelten Parser **Preis auf Anfrage**

	C	D		C	D
Yie are Kung Fu	22.-	a. A.	Movi	22.-	38.-
Wintergames	22.-	a. A.	International Karate	29.-	42.-
Wizzards Lair	10.-	36.-	Heavy on the Magick	22.-	38.-
Way of the Tiger	20.-	39.-	Hypersports	22.-	42.-
"V"	22.-	38.-	Hi Rise	9.90	a. A.
Turbo Esprit	22.-	38.-	Crafton/Get Dexter	22.-	42.-
Saboteur	22.-	38.-	Glass	22.-	42.-
Southern Bell	22.-	a. A.	Ghosts'n Goblins	22.-	42.-
Stairway to Hell	10.-	a. A.	Friday the 13th	22.-	42.-
Room Ten	29.-	a. A.	Doom Day/Eden Blues	9.90	28.-
Rescue on Fractacus	30.-	a. A.	Core	22.-	a. A.
Paper Boy	29.-	a. A.	Battle beyond the Stars	10.-	a. A.
Ping Pong	22.-	a. A.	The Ceti	a. A.	38.-
Nick Faldo's Open	22.-	a. A.	The Devils Crown	a. A.	38.-
Mission Omega	32.-	47.-	Miami Vice	a. A.	47.-
Meltdown	20.-	38.-	Blue Ribbon Games	--	36.-
Macadam Bumper	27.-	a. A.	Equinox	a. A.	38.-
Match Day	22.-	a. A.	Codename Mat II	--	36.-

Diskettenboxen 3" 29.-, 3.5" a. A., 5.25" ab 14.-
3"-Disketten 10 Stück 95.-
5.25"-Disketten, MD 1D, 48 tpi 10 Stück BULK 12.50
5.25"-Disketten, MD 2D, 96 tpi 10 Stück BULK 26.50
 Bitte Tagespreise erfragen!

Wir führen Farbänder, Etiketten, Tabellierpapier und Casettenaufkleber. Weitere Soft- und Hardware auf Anfrage. Versand: per Nachnahme + 6.50 DM, per V-Scheck + 5.-DM. Ab 300.- DM Bestellwert versandkostenfrei.

DELTA COM
Hölderlinstraße 16, Postfach 14 1266,
4100 Duisburg 14, ☎ 021 35 / 527 67

Endlich Schluß mit der Rumhackerei!

Spielerisch lernen Sie 10-Finger-Blindschreiben (Tastschreiben) mit Ihrem intelligenten Trainingspartner.

① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨ ⑩ ⑪
 (Q) (M) (E) (R) (T) (Z) (U) (I) (O) (P) (U) (+)
T A S T S E T[®]
 (A) (S) (D) (F) (G) (H) (J) (K) (L) (U) (H) (H)
 (V) (X) (C) (U) (B) (N) (M) (.) (.) (-) (.)

Das Trainingspiel für Anfänger, Fortgeschrittene, Wiedereinsteiger und Programmierer. Im Schulunterricht bewährt, vollständige DIN-Tastatur mit Sonderzeichen. Optimale Unterstützung durch ständige Reaktionszeit- und Fehleranalyse.

3 Spielstufen, Anfängerstufe, 5-Phasen-Training für jeden Zeichenbereich, Testfunktion, gezielte Wiederholung und Vertiefung.

CPC 464 **DM 98.-**
CPC 664/6128 (autom. Ergebnisspeicherung) **DM 108.-**
Produktinfo gegen Briefmarken **DM 2,40**

T A S T S E T[®]
 Lernsystem für Tastschreiben

H. E. Zimmerle · Am Eckbusch 38/110 · 5600 Wuppertal 1 · Tel. 0202/72 1129

Lieferung gegen Vorkasse (Scheck) oder Nachnahme zzgl. DM 5,- Versandkosten.

® reg. Warenzeichen. Händleranfragen erwünscht.

Datenverwaltung unter Turbo-Pascal (Teil 3)

Im dritten Teil geht es um den binären Baum

Das letzte Mal haben wir die Prinzipien zur Verwaltung einer Liste mit Hilfe des dynamischen Datentyps "Zeiger" (pointer) kennengelernt. Heute sollen verschiedene Listenstrukturen bis hin zum "binären Baum" behandelt werden. Doch zunächst fassen wir noch einmal das bisher Behandelte zusammen.

Einfach verkettete Liste

Eine Liste ist immer eine Sammlung von Elementen des gleichen Datentyps (in der Regel ein Record-Verbund). Alle Elemente der Liste sind durch Zeiger miteinander verkettet. Der Zugriff auf die Elemente der Liste ist nur über die Wurzel möglich. Um den Zugriff auf die Wurzel und damit auf die gesamte Liste zu gewährleisten, muß ein Merk-Zeiger auf die Wurzel zeigen. Das Ende der Liste ist dadurch gekennzeichnet, daß das letzte Element der Liste auf NIL zeigt. Um einen schnelleren Zugriff auf das letzte Element der Liste zu erhalten (Liste erweitern), wird ebenfalls ein Zeiger auf dieses letzte Element der Liste verwaltet. Eine leere Liste liegt dann vor, wenn der Zeiger der Wurzel NIL ist. Für das Erzeugen eines neuen Elements haben wir die Prozedur NEW kennengelernt. Der Zugriff auf den Inhalt eines Elements ist über das Dereferenzieren möglich. Soweit der Rückblick.

Das kleine "Adressen-Programm" aus der letzten Folge war ein typisches Beispiel für eine "einfach verkettete Liste". Der Zugriff auf Elemente der Liste war ausschließlich über das erste Element und sequentiell möglich. Die Liste konnte nur in einer Richtung ("von oben nach unten") bearbeitet werden. Das Ende war leicht festzustellen, da es durch NIL gekennzeichnet ist. Nach dem Prinzip der einfach verketteten Listen arbeiten auch die beiden folgenden Listenstrukturen, die ich Ihnen kurz vorstellen möchte.

Warteschlange (FIFO)

Diese Liste arbeitet nach dem "First In, First Out"-Verfahren (zuerst rein, zuerst raus). Sie dient vor allem zur Abarbeitung von Elementen in einer be-

stimmten Reihenfolge. Elemente dürfen bei dieser Listenstruktur ausschließlich an das Ende der Liste angehängt werden. Gelöscht bzw. zugegriffen werden darf immer nur auf das erste Element (Wurzel) der Liste. Sehen Sie sich dazu folgendes Beispiel an. 3 Elemente wurden nacheinander an die Liste angehängt. Das zuerst eingegebene Element bildet die Wurzel, die folgenden Elemente sind entsprechend verkettet.

```
Wurzel Element_1
        Element_2
        Element_3
```

Soll z.B. ein Element bearbeitet werden, wird das Element genommen, auf das die Wurzel zeigt. Die Wurzel wandert dann auf das folgende Element (hier Element_2). Ein weiteres Element wird an das Ende der Liste angehängt. Die Reihenfolge wird entsprechend eingehalten.

Diese Listenstruktur findet sehr häufig Anwendung bei Warteschlangen. So werden z.B. Kommandos, die nacheinander abgearbeitet werden sollen, nach diesem Prinzip verwaltet. Das Kommando, das bearbeitet wurde, fällt dann meistens aus der Liste raus. Die zweite einfache Listenstruktur ist der "Stapel".

Stapel (LIFO)

Der Stapel ist eine häufig verwendete Listenstruktur, die nach dem Prinzip "zuletzt rein, zuerst raus" (Last In, First Out) organisiert ist. Der Zugriff auf Elemente (z.B. Löschen) und das Anhängen von Elementen ist ausschließlich am Listenende möglich. Stapel-Listen werden oft als Zwischenspeicher benutzt, auf die Elemente für eine gewisse Zeit abgelegt (push) und von hier wieder geholt (pop) werden.

Ringliste

Eine abgewandelte Form der einfach verketteten Liste ist die sogenannte Ringliste. In dieser Liste zeigt das letzte Element wieder auf das erste der Liste. Es gibt also kein Listenende. Besteht eine Liste nur aus einem Element, so zeigt dieses wieder auf sich selbst (zyklische Struktur). Bei Ringlisten ist es möglich,

von einem beliebigen Element der Liste aus auf alle anderen Elemente der Liste zuzugreifen.

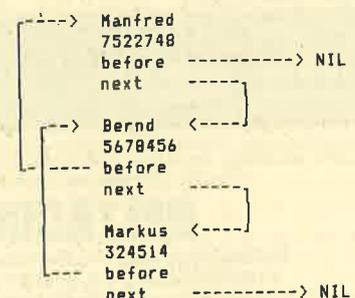
Alle einfachen Listenstrukturen haben einen entscheidenden Nachteil: Sie lassen sich immer nur in einer bestimmten Reihenfolge bearbeiten! Diese ist über die Wurzel durch die Verkettung vorgegeben. Werden bestimmte Elemente gesucht, muß die Liste von der Wurzel an durchsucht werden, bis das gewünschte Element gefunden oder aber das Ende der Liste erreicht ist. Es ist unmöglich, von der aktuellen Listenposition die Liste "rückwärts" zur Wurzel hin zu verfolgen.

Doppelt verkettete Listen

Das Ziel von doppelt verketteten Listen ist die Möglichkeit, von einem Element aus zu dem vorigen (Vorgänger) und zum nächsten Element (Nachfolger) der Liste zu gelangen. Die Verkettung der Elemente untereinander muß mit zwei verschiedenen Zeigern erfolgen. Der eine Zeiger zeigt auf den Vorgänger, der andere auf den Nachfolger. Das Element benötigt also zwei Zeigerkomponenten:

```
TYPE
zeiger = ^element;
element = RECORD
    inhalt : STRING[20];
    before : zeiger;
    next : zeiger;
END;
```

Die Verkettung selbst findet wie in dem folgenden kleinen Beispiel statt. Da die Wurzel keinen Vorgänger hat, zeigt der Before-Zeiger auf NIL. Das Ende der Liste ist dadurch gekennzeichnet, daß "next" auf NIL zeigt (kein Nachfolger):



nen Stein von einem Stapel auf den anderen legen kann, läßt sich mit dieser Tätigkeit ein beliebig großer Stapel umschichten, indem sich die Tätigkeit immer wieder selbst aufruft. Die Abbruchbedingung wäre hier erfüllt, wenn der Stapel komplett umgeschichtet ist.

Der Anwendungsbereich von Rekursionen ist m.E. sehr eingeschränkt und in der Regel nur bei mathematischen Problemen nützlich. Der Vorteil liegt eindeutig in der Einfachheit und Übersichtlichkeit, sofern der Programmierer die rekursiven Regeln beherrscht. Der gravierende Nachteil liegt vor allem in der sehr langsamen Ausführung eines rekursiven Programms und im größeren Speicherplatzbedarf für das Programm selbst (rekursiver Code). Nicht zu vergessen ist, daß bei jedem Aufruf der Prozedur alle benutzten Variablen und Parameter auf dem Stapel zwischengespeichert werden. Die mögliche Rekursionstiefe ist daher auch von der Speicherkapazität des Rechners abhängig.

In fast allen Lern- und Computerbüchern finden Sie als Beispiel für eine Rekursion die Fakultätsberechnung und die Bestimmung von Fibonacci-Reihen, auf die ich an dieser Stelle allerdings verzichten will. Sie machen zwar das Prinzip der Rekursion sehr deutlich, bringen aber gegenüber der iterativen Programmierung (Schleife) keine Vorteile, sondern nur Nachteile.

Wir wollen deshalb das Einfügen eines Elements in einen Baum rekursiv lösen, denn hier wird eine der wenigen wirklich effektiven Möglichkeiten deutlich. Zuvor ist allerdings noch anzumerken, daß Turbo-Pascal grundsätzlich keinen rekursiven Code erzeugt, sondern ihm das mit der Compiler-Direktive

(\$A-) rekursiver Code

(\$A+) nicht rekursiver Code

mitzuteilen ist. Da ein rekursiver Code länger und langsamer ist, sollten nur die Prozeduren, die sich selbst aufrufen, mit den beiden Direktiven geklammert werden.

Das Programm

Das Beispielprogramm "Binär-Baum" hat die Aufgabe, einen beliebigen Text zeichenweise einzulesen, die einzelnen Zeichen sortiert in einen Baum einzufügen und, falls ein Zeichen mehrmals auftritt, die Anzahl zu zählen. Am Ende des Programms sollen alle Zeichen sortiert mit der Angabe, wie oft sie aufgetreten sind, auf dem Bildschirm ausgegeben werden.

Das Hauptprogramm (Zeile 60 bis 83) eröffnet eine Text-Datei und liest Zei-

```

1 PROGRAM Binaer_Baum;
2
3 TYPE
4     zeiger = ^element;
5     element = RECORD
6         begriff : CHAR;
7         anzahl : INTEGER;
8         links : zeiger;
9         rechts : zeiger;
10    END;
11
12
13 VAR
14     wurzel : zeiger;
15     dateiname : STRING[13];
16     datei : TEXT;
17     zeichen : CHAR;
18
19
20 {$A-} { Es wird ein rekursiver Code erzeugt ! }
21
22 PROCEDURE Einhaengen(zeichen: CHAR; VAR pointer : zeiger);
23
24 BEGIN
25     IF pointer=NIL THEN
26     BEGIN
27         NEW(pointer);
28         pointer^.begriff:=zeichen;
29         pointer^.anzahl :=1;
30         pointer^.links :=NIL;
31         pointer^.rechts :=NIL;
32     END
33     ELSE
34         IF zeichen>pointer^.begriff THEN
35             einhaengen(zeichen,pointer^.rechts)
36         ELSE
37             IF zeichen<pointer^.begriff THEN
38                 einhaengen(zeichen,pointer^.links)
39             ELSE
40                 pointer^.anzahl:=pointer^.anzahl+1;
41     END;
42
43
44 PROCEDURE Listen(pointer : zeiger);
45
46 BEGIN
47     IF pointer<> NIL THEN
48     BEGIN
49         Listen(pointer^.links);
50         Writeln('Das Zeichen <',pointer^.begriff,'> war ',
51             pointer^.anzahl:4,' mal vorhanden. ');
52         Listen(pointer^.rechts);
53     END;
54 END;
55
56 {$A+} { Rekursiver Code wieder aufgehoben }
57
58
59 BEGIN
60     wurzel:=NIL; {Initialisieren}
61     CLRSCR;
62     Writeln('Sortiere und zaehle die Zeichen eines Textes');
63     Writeln;
64     Write('> Name der Textdatei (mit Typ) : ');
65     Readln(dateiname);
66     Assign(datei,dateiname);
67     Reset(datei);
68     Writeln;
69     While Not(Eof(datei)) Do
70     BEGIN
71         While Not(Eoln(datei)) Do
72         BEGIN
73             Read(datei,zeichen);
74             Einhaengen(zeichen,wurzel);
75             Write(zeichen);
76         END;
77         Readln(datei);
78         Writeln;
79     END;
80     CLRSCR;
81     Listen(wurzel);
82 END.
83
84 {M.W.Thoma, Hamburg August 1986}

```

chen um Zeichen ein (bis End-Of-Line, dann wird EOLN überlesen, Zeile 78). Das gelesene Zeichen wird dann der Prozedur Einhaengen übergeben. Sie fügt das Zeichen im Baum ein. Am Ende des Programms wird der Baum ausgedruckt.

Die beiden Prozeduren Einhaengen und Listen sind rekursiv und werden mit den Compiler-Direktiven (Zeile 20 und 57) geklammert. Sehen wir uns zunächst die Prozedur Einhaengen an. Ihr wird das Zeichen übergeben, das in den Baum eingefügt werden soll, sowie zusätzlich eine Zeigervariable. Beachten Sie, daß dies ein Variablenparameter ist (der geänderte Inhalt kommt zurück). Spielen wir die Prozedur einmal durch. Beim ersten Aufruf (Zeile 75) werden ihr ein Zeichen (z.B. M) und die Wurzel übergeben, die z.Z. auf NIL zeigt (Zeile 61). Die erste Anweisung in der Prozedur prüft, ob der übergebene Zeiger (die Wurzel) NIL ist. Da dies zutrifft, wird ein neues Element erzeugt und mit den entsprechenden Werten gefüllt. ("anzahl" bestimmt, wie oft dieses Zeichen im Text vorhanden war.) Das war's!

Da der übergebene Zeiger (hier die Wurzel) ein Variablenparameter ist, ent-

hält die Variable wurzel aus dem Hauptprogramm den Zeigerwert vom ersten Element! Das nächste gelesene Zeichen soll ein T sein. Die übergebene Wurzel ist jetzt nicht mehr NIL (sie zeigt auf das Element mit dem Wert M), und die IF-Bedingung ist nicht mehr erfüllt! Es wird also in den ersten ELSE-Zweig verzweigt (Zeile 34). Hier wird der Inhalt des Elements (M) mit dem Zeichen (T) verglichen. Wenn das Zeichen größer ist als der Inhalt, wird die Prozedur rekursiv aufgerufen. Als Zeiger wird "pointer^.rechts" übergeben, da ja im rechten Teil des Baumes weiter gesucht werden muß. Die Prozedur wird also erneut, aber mit einem anderen Zeigerinhalt übergeben. Dieser zeigt auf NIL, und die IF-Bedingung ist erfüllt (Element anlegen und Zeiger auf NIL setzen). Die aufgerufene Prozedur ist damit beendet und kehrt zur aufrufenden Prozedur zurück. Da der Zeiger als Variablenparameter übergeben wird, erhält der formale Parameter (hier "pointer^.rechts") den geänderten Wert, und das ist der Zeigerwert auf das neue Element! Die Verkettung ist damit bereits erfolgt.

Ist das Zeichen kleiner als der Inhalt des jeweiligen Elements, wird die Prozedur rekursiv mit dem Zeiger auf den lin-

ken Teil des Baumes aufgerufen. Ist der Inhalt weder kleiner noch größer als das Zeichen, so kann es nur noch gleich sein. In diesem Fall wird der Zähler (anzahl) des Elements nur um eins erhöht. Nach diesem Verfahren hangelt sich die Prozedur durch den gesamten Baum. Ein wirklich geniales Verfahren!

Die Prozedur Listen arbeitet ähnlich. Von der Wurzel aus wird zunächst solange nach links zum allerkleinsten Element hin verzweigt, bis es nicht mehr weitergeht. Erst dann wird das erste Zeichen ausgegeben. Anschließend erfolgt eine Verzweigung nach rechts (hier könnte ja was folgen). Geht es da weiter, wird von diesem Element aus wieder nach links verzweigt. Die Inhalte werden dann in sortierter Reihenfolge ausgegeben!

Zweifellos reicht die Beschreibung von rekursiven Prozeduren und Funktionen zum vollen Erfassen nicht aus. Versuchen Sie deshalb einmal, ein Beispiel mit Zettel und Papier zu lösen. Sie kommen dann dem Verfahren und dem logischen Prinzip immer näher. Das Programm soll Ihnen dazu eine Hilfe sein.

Manfred Walter Thoma

Anzeige

Anzeige

Der Programm-Generator: Ein Programm, das Programme schreibt!

Nachdem die gängigen Programmier-Techniken eine gewisse Reife erlangt haben und das programmtechnische Methodenrepertoire ausgereizt ist, drängt sich immer mehr der Gedanke an eine **Automation des Programmierens** auf.

Programme durch Programme schreiben zu lassen, das ist die Aufgabe der Programm-Generatoren. Von der Firma **Tüftler-Union (Postfach 82, A-6961 Wolfurt)** wurde nun ein Programm-Generator für das Schneider-Basic entwickelt. Dieser bringt zuerst im Dialog mit dem Benutzer in Erfahrung, was das zu erzeugende Programm tun und wie es gestaltet sein soll. Anschließend schreibt er mit Hilfe des eingebauten Syntax- und Algorithmenwissens das angestrebte Zielprogramm auf ein File, von wo aus dieses geladen und durch MERGE mit anderen Programmen vereint werden kann. Auf diese Weise lassen sich Programme für Masken, Menüs, Options-Schalter, Eingabe- und Resultatausgabeformulare, Tabellen, Listen, Windows, Dialoge, Druckerprotokolle und vieles mehr erzeugen. Die erzeugten Basic-Programme sind frei zugänglich, listbar, modifizierbar, und es werden für sie keinerlei Urheberansprüche erhoben.

Der "Urheber" ist ja kein Mensch, sondern der Generator!

Innerhalb von wenigen Minuten lassen sich große Programmblöcke schreiben, die aufgrund ihrer maschinellen Erzeugung frei von Syntax-Fehlern sind und daher auch nicht mehr aufwendig getestet werden müssen. Während im internationalen Durchschnitt ein Programmierer ca. 15-25 Programmzeilen pro Tag schafft, erzeugt der Generator ein Vielfaches davon in wenigen Sekunden. Die Bedienung des Generators ist weitgehend selbsterklärend.

Preis des Generators: 89.- DM

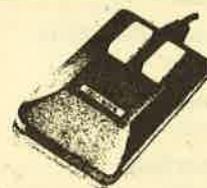
WANTED

Sind **SIE** etwa derjenige, der sich bis heute allen Vorzügen unserer MOUSEPACK-Systeme für CPC und JOYCE zum Trotz noch nicht in den Besitz eines dieser deutschen Qualitätsprodukte gebracht hat?

Stellen Sie sich freiwillig!

Reward

Bearbeitungsgebühren:
DM 228.- für den CPC-Besitzer
DM 249.- für den Joyce-Besitzer



Rechtsbeistand oder Selbstanzeige bei:

Gerdas · Imperial SOFTWARE Systems

Heidegartenstraße 36 · D-5300 Bonn 1
Telefon 02 28 / 25 24 74

RSE software

(023 05) 3770

SOFTWARE-AUTOREN GESUCHT!



Die intelligente Dimension

Adressverwaltung

- 100% Maschinencode
- Relative Dateiverwaltung
- Professioneller Editor
- Schnelles Auffinden von Daten
- 1 Benutzersatz
- 900 Datensätze
- Etikettendruck für verschiedene Formate
- Listendruck mit Einzel- oder Endabblattnummerierung
- Integrierte Hardcopyfunktion
- DIN-Tastaturbelegung
- CPC 464, 664, 6128

Diskette 59,- DM*

Disc-Scanner

- 100% Maschinencode
- Unterstützt 2 Laufwerke
- Disketten kopieren
- Lesen/Schreiben beliebiger Sektoren
- Lesen/Schreiben (Full Screen)
- Sektoren modifizieren in Track/Sektor
- Umrechnen von Blocks in Track/Sektor
- Files Umbenennen und Löschen
- Reaktivieren gelöschter Files
- Graph. Darstellung der Diskbelegung
- Useumnummern wählbar
- Formattieren / opt. Einzelspur
- Vendor / Data / 43 Tracks
- Integrierte Hardcopyfunktion
- CPC 464, 664, 6128

Diskette 79,- DM*

Disc-Sorter

- 100% Maschinencode
- Relative Dateiverwaltung
- Automatische beibehaltene Programm-erfassung von einer Diskette in
- wahlweise manuelle Erfassung
- Automatische Erfassung und Verarbeitung aller Fileinformationen

- Professioneller Editor
- schnelle Sortierroutine
- schnelles Auffinden von Daten
- Listendruck mit Einzel- oder Endabblattnummerierung
- Integrierte Hardcopyfunktion
- CPC 464, 664, 6128

Diskette 69,- DM*

Amsmonix

- Komfortabler Monitor-Debugger
- 100% Maschinencode
- RAM-ROM-AMSDOS-Monitor
- Disassembler
- Such-Fix-Modifizierungsfunktion
- Laden und Speichern von Files
- CAT-Funktion
- Intellig. Kopieren (Blockverschiebung)
- Registeranzeige
- Audruf von Maschinencodeprogrammen
- Texteingabe
- Druckprotokoll
- Anwahl auch von Expansions-Roms
- Monitor ASCII o. Prüfsummen-Anz.
- CPC 464, 664, 6128

Cassette 49,- DM*
Diskette 59,- DM*

Lagerverwaltung

- 100% Maschinencode
- Relative Dateiverwaltung
- Professioneller Editor
- Bestands-/ Inventurliste
- Artikel/VK-Preisliste
- Etikettendruck
- Ausdruck auf Einzel- o. Endlospapier
- Geschützter EK-Preis
- Statusik
- DIN-Tastatur
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- (bis zu 1350 Artikel je Daten)
- Bestell-Vorschlagsliste
- CPC 464, 664, 6128

Diskette 79,- DM*

Mathe-Pack

- Vom Anfänger bis zum Profi
- Eingebauter Taschenrechner
- Integrierte Hardcopyfunktion
- Algebra
- Geometrie
- Trigonometrie
- Mengenlehre
- Analysis
- Kurvenabschätzung
- Benutzerschnitt
- CPC 464, 664, 6128

Cassette 59,- DM*
Diskette 69,- DM*

Karteikasten

- 100% Maschinencode
- Relative Dateiverwaltung
- 400 Karten pro Diskettensatz
- Beliebiges Suchkriterium
- Professioneller Editor
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- Integrierte Hardcopyfunktionen
- Listendruck
- DIN-Tastaturbelegung
- CPC 464, 664, 6128

Diskette 79,- DM*

Vokabeltrainer

- Univers. / jede Fremdsprache nutzbar
- 100 Vokabeln pro Unit
- unbegrenzte Unit-Bearbeitung
- 2 Lernstufen PP (Prägen/Prüfen)
- Einfache Handhabung durch Cursor
- blockstrenge
- Lernen von Mehrfachzuweisungen
- möglich (z.B. unregelmäßige Verben)
- Prozentuale Ergebnisauswertung
- selbst bei Testunterbrechung
- Von Pädagogen empfohlen
- Deutscher Zeichensatz
- CPC 464, 664, 6128

Cassette 59,- DM*
Diskette 69,- DM*

Universaldatei

- 100% Maschinencode
- Relative Dateiverwaltung
- je nach Maske und Indexfelder mehrere Tausend Datensätze möglich
- bis zu 50 Felder pro Datensatz
- Frei definierbare Eingabemaske
- Such- und Druckerprogramme
- erstellbar
- Professioneller Editor
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- DIN-Tastaturbelegung
- Integrierte Hardcopyfunktion
- CPC 464, 664, 6128

Diskette 79,- DM*

Text-Manager

- 100% Maschinencode
- Bandeneinstellung/Tabulatoren
- Blockatz/Flattersatz, Briefkopfhilfe
- ASCII- oder DIN-Tastatur
- dynamischer Wortbruch
- Fontkatalogen
- Wörterbuch und -ersetzen
- Texte kopieren oder verschieben
- Text formatieren (zeilenweise oder Gesamttext)
- Drucker-Speicherspeicher
- löschen, ein- oder ausblenden
- Zeichen/Löschen einzeln oder blocken
- Basic-Programm editieren
- Wahl des Speichermediums
- CPC 464, 664, 6128

Cassette 69,- DM*
Diskette 79,- DM*

Telecom 1000

- Professionelles DPL-Programm
- 100% Maschinencode
- Einstellen d. Übertragungsparameter
- 75, 110, 150, 300, 600, 1200 Baud
- Voll- und Halbduplex
- 1 oder 8 Datenbits
- 1 oder 2 Stopbits
- Keine, gerade oder umgekehrte Parität
- versch. Übertragungsprotokolle
- Über 32 K.T. ext. buffer
- Notizzettel
- Professioneller Editor
- Übertragung von ASCII, Binär, Basic-Dateien
- Convergenzprogramm
- CPC 464, 664, 6128

Cassette 69,- DM*
Diskette 79,- DM*

Copy-Master

- Universelles File-Transfer Programm
- 100% Maschinencode
- Kopiert und analysiert Ihre Software
- Cass. zu Cass.
- Cass. zu Cass.
- Disc. zu Cass.
- Disc. zu Disc.
- Disc Backup
- Disksensitivitätskopie auf Cass. (kompl.)
- Programmresident im Speicher
- Integrierte Hardcopyfunktion
- Programmverschiebung bei AMSDOS-Überschiebung
- CPC 464, 664, 6128

Cassette 49,- DM*
Diskette 59,- DM*

Super-Hardcopy

- 100% Maschinencode
- PSX-Befehle
- Menügesteuert an versch. Druckertypen anpassbar
- Text-Hardcopyfunktion
- Parameter wählbar
- Normal- und Inversendruck
- kompensierende Hardcopy
- 4-Farb-Darstellung möglich
- 1 Graf.-Hardcopy vom Bildschirm
- CPC 464, 664, 6128

Cassette 39,80 DM*
Diskette 49,80 DM*

BRANDNEU...

BRANDNEU...

Der Schneider PC

Der Schneider PC ist ein kompatibler IBM-PC mit hervorragenden zusätzlichen Erweiterungen. Er wird als vollständiges System geliefert und hat in der Standard-Ausführung die folgenden Kennzeichen:

- 512 KB System mit allen Schaltkreisen auf einer einzigen Hauptplatine 8086 Prozessor mit 8 MHz getaktet. Der Hauptspeicher ist auf der Hauptplatine auf 640 KB erweiterbar.
- Colour Grafik-Karte mit speziellem 16-Farb Hi-Res Modus.
- Drei »full-size« kompatible Erweiterungsstecker. Am Standardgerät steht ausreichend Strom zur Verfügung, um eine »hard-disk« und »controllers« zu versorgen.
- Wahl zwischen einer oder zwei 360 KB 5 1/4"-Diskettenlaufwerke

- Die Hauptplatine beinhaltet je eine serielle und parallele Schnittstelle mit Standard-Belegungen
- Lautsprecher mit Lautstärkereglung
- Batterie-gepufferte Hardware-Uhr (real time clock) und Konfiguration RAM

Socket für 8087 »mathem. Co-Prozessor«
IBM-PC-kompatible ROM BIOS
Der wahlweise mitgelieferte Farbmonitor beinhaltet folgende Möglichkeiten:

Textmodus: mittlere Auflösung 16 Farben, 40x25 Zeichen, hochauflösend 16 Farben, 80x25 Zeichen.

Graphikmodus: Mittlere Auflösung drei 4-farb-Paletten, 320x200 Punkte; hochauflösend 2-farbig, 640x200 Punkte (Hi-Res Mode); speziell hochauflösend 16 Farben, 640x200 Punkte

- Komplettes IBM-Standard-Tastenfeld, beleuchtete (Numlock) und (Capslock)-Tasten, Joystick-Anschluß, (Extra-), (Del-) und (Enter)-Tasten.
- 2-Tasten-Mouse (Microsoft-kompatibel)
- IBM-PC-kompatible ROM BIOS
- Umfangreiche mitgelieferte Software (4 Disketten): Microsoft MSDOS 3.2, Digital Research GEM, GEM Desktop, Digital Research GEM Paint, Digital Research DOS PLUS, betreibt MS DOS und CP/M-86, GEM unterstützt Locomotive BASIC 2
- Benutzer-Handbücher

Monochrome-Bildschirm
1 Laufwerk

1999,-

Monochrome-Bildschirm
2 Laufwerke

2499,-

Farb-Bildschirm
1 Laufwerk

2499,-

Farb-Bildschirm
2 Laufwerke

2999,-



IBM ist ein registriertes Warenzeichen für International Business Machines Corporation, Microsoft und MSDOS sind registrierte Warenzeichen für Microsoft Corporation, DOS Plus, GEM, GEM Desktop, GEM PAINT sind Warenzeichen von Digital Research Inc., Locomotive BASIC 2 ist ein Warenz. für Locomotive Software Ltd.

Liebe Leser!

Es ist schon irgendwie ärgerlich! Obwohl mit viel Tamtam der neue Schneider-PC bereits im September der Presse vorgestellt wurde, steht uns in der Redaktion noch kein Schneider-PC zur ständigen Verfügung. Auf waghalsige ein- oder zweitägige Tests können wir uns als gute Fachzeitschrift aber nicht einlassen. Wie das die anderen machen, ist uns schleierhaft. Vielleicht haben die schon einen Schneider-PC zur ständigen Verfügung? Entsprechend fällt der geplante Test des neuen Schneider-PC ins Wasser.

Trotz alledem beginnen wir hier in dieser Ausgabe des Schneider-Magazins mit dem ersten PC-Teil. Den Schwerpunkt für die nächsten Ausgaben haben wir speziell bei der "Einsteiger-Hilfe" angesetzt, nämlich da, wo gerade die neuen PC-Besitzer die ersten Schwierigkeiten haben.

Heute geht es u. a. um die Benutzung der Funktionstasten unter MS-DOS. Bei der Soft- und Hardware beginnen wir mit Produkten der unteren Preisklasse, die für den privaten Anwender noch finanzierbar sind. Aber auch Produktbeschreibungen der gehobenen Preis- und Lei-

stungsklasse finden hier ihren Platz. Der O.K.-Text wird eine ständige und kritische Rubrik im PC-Teil werden. Bereits heute starten wir einen 8088/8086-Maschinensprache-Kurs und schließen uns damit an einen von vielen Lesern geäußerten Wunsch an. Buchbesprechungen und Hintergrundinformationen fehlen natürlich ebenfalls nicht.

In der nächsten Ausgabe folgt, so Schneider will, ein ausführlicher Erfahrungs- und Testbericht über den PC 1512, ein Bericht über den Betrieb von Netzwerken an PCs und natürlich die Fortsetzung des 8088/8086-Kurses. Ich hoffe, daß wir damit den richtigen Einstieg gefunden haben. Für Anregungen, Kritik und tatkräftige Unterstützung von Ihnen sind wir jederzeit dankbar.

Ihr Manfred Walter Thoma



Wie war denn das mit MS-DOS?

MS-DOS ist heute das verbreitetste Betriebssystem für 16-Bit-Prozessoren und wurde von Microsoft auf der Basis der Intel-Prozessoren 8088/8086 entwickelt. Das von IBM vertriebene PC-DOS ist im wesentlichen identisch. MS-DOS hat eine gewisse Ähnlichkeit mit dem CP/M-Betriebssystem und der Umsteiger vom CPC auf den PC 1512 wird bei den Kommandos Ähnlichkeiten vorfinden. Trotzdem orientierte man sich bei MS-DOS (vor allem in der internen Struktur) mehr am Betriebssystem UNIX, das bei Mini-Computern eine große Verbreitung besitzt.

MS-DOS ist ein reines Single-Task- und Single-User-Betriebssystem. Das heißt, es kann nur ei-

ne Aufgabe (Task) pro Zeileneinheit erledigen. Das QDOS aus dem Jahre 1980 von T. Paterson ist ein Vorläufer des heutigen MS-DOS. Es wurde über das 86-DOS hin zum PC-DOS entwickelt und 1981 mit den IBM-PCs vertrieben. Microsoft brachte fast zur gleichen Zeit die MS-DOS Version 1.0 für IBM-kompatible Rechner auf den Markt. Die Version 1.0 unterstützte nur einseitige Laufwerke mit 8 Sektoren pro Spur (160 KByte Speicherkapazität) und das ganze System steckte noch voller Fehler. Mit der Version 1.05 (kostenlose Nachlieferung) wurden die wesentlichen Fehler im System behoben (vor allem im Basic-Interpreter). Aber erst die Version 1.1 konnte man

als fehlerfrei bezeichnen (soweit das überhaupt möglich ist). Speziell im "Nucleus" (Kern des Betriebssystems) wurden erhebliche Verbesserungen eingefügt. Diese Version unterstützte auch doppelseitige Laufwerke (360 KByte). Weiterhin wurden einige Kommandos und Dienstprogramme neu hinzugefügt. Das Directory zeichnete zusätzlich noch die Uhrzeit des letzten Zugriffs auf eine Datei auf.

Die anschließende Version 2.0 ähnelte dann immer mehr UNIX und bot eine hierarchische Dateistruktur an. So ließen sich hier mehrere Subdirectories (Unterverzeichnisse) anlegen und Dateien als geordneter "Baum" verwalten. Neu waren auch Umleitungen und Pipelines. (Mit Umleitungen lassen sich Ein- oder Ausgaben auf eine andere Einheit [Drucker, Diskette etc.] umleiten, und mit Pipelines kann man Informationen [Ausgaben]

eines Programms direkt an ein anderes Programm als Eingabe übergeben.) Die Systemfunktion (Interrupt 21H) wurden bei der Version 2.0 um einige UNIX/XENIX-kompatible erweitert.

Die dann im Jahre 1983 erschienene Version 2.1 bot als Neuerung länderspezifische Eigenschaften (Darstellung von Zeit und Datum etc.), die wahlweise aufgerufen werden konnten. Version 2.11 (deutsche Version 2.1) ist die derzeit noch verbreitetste Version. Version 3.0 brachte keine wesentlichen Änderungen (einige neue Dienstprogramme) und ist als Vorläufer zur Version 3.1 (Ende 1985) zu verstehen. Als wesentliche Neuerung sind die Netzwerkfunktionen zu nennen. Bei zukünftigen Versionen (4.X und 5.X) werden wohl Funktionen zum Multi-Task- und Multi-User-Betrieb zu finden sein.

Manfred Walter Thoma

BÜCHER

Textverarbeitung auf dem Prüfstand

Von Dietmar Eirich
Signum-Verlag
239 Seiten, 48.-DM
ISBN 3-924767-05-X

Dietmar Eirich, der schon durch zahlreiche Veröffentlichungen in verschiedenen Computerzeitschriften und der Fernsehserie "Wer hat Angst vor dem kleinen Chip" auffiel, bietet mit diesem Buch dem absoluten Computerneuling, der sich speziell für Textverarbeitung interessiert, eine "Kaufberatung und Auswahlhilfe".

Das Buch gliedert sich in folgende Abschnitte: Als erstes wird erklärt, was Textverarbeitung ist und was man bei der Auswahl des Programms beachten sollte. Danach geht der Autor kurz auf Zusatzprogramme für Serien-Schriftstücke ein, erklärt die Zukunft der "Elektronischen Post" und gibt Tips für den Bürobetrieb. Danach folgt der Hauptabschnitt: Es werden 29 unterschiedliche Textverarbeitungsprogramme (zwischen 180.-DM und 2850.-DM) auf durchschnittlich je 5 Buchseiten nach einheitlichen Kriterien geprüft. 16 weitere, nur kurz geprüfte Programme folgen im anschließenden Kapitel auf 7 Seiten. Jedes dieser 29 Programme wird in einzelnen kleinen Kapiteln kurz beschrieben und die Systemvoraussetzung genannt. Danach geht Dietmar Eirich auf den Lieferumfang (wieviele Disketten) und die Dokumentation (Umfang und Qualität des Handbuchs) ein. Als Abschluß eines solchen Kapitels steht die Gesamtbewertung.

Sind bei einem Programm Besonderheiten im Umgang mit diesem vorhanden, geht der Autor noch einmal gesondert darauf ein. Dies können positive wie auch negative Punkte sein. Geprüft wurden folgende Programme: Word, WordStar, Tex-Ass,

Perfect Writer II, Easywriter I und II, aber auch unbekanntere und spezielle Programme wie Ghostwriter, M.I.S.-Rechentext, Toptip und Scientex. Im Anschluß an die Prüfung der Textverarbeitungsprogramme folgt ein Kapitel über den Druckermarkt. Auf 17 Seiten vermittelt der Autor noch, was man über Drucker wissen muß (technische Unterschiede der einzelnen Arten, Geschwindigkeit, Qualität des Schriftbildes und Anschluß des Druckers an den Rechner). Den Abschluß des Buches bildet der Bezugsquellennachweis.

Dieses Buch bietet dem Einsteiger in Sachen Textverarbeitung mit dem Computer eine Fülle von Informationen über die benötigten Geräte und Programme. Durch seine durchgängig gleichen Kriterien in der Prüfung gibt es dem potentiellen Käufer eine echte, erste Entscheidungshilfe, wobei man bedenken muß, daß diese Testberichte subjektiv die Meinung des Autors zu den geprüften Programmen wiedergeben. Es erspart aber das viele Lesen der einzelnen Testberichte in den unterschiedlichen Fachzeitschriften, die natürlich auch keine einheitlichen Prüfungskriterien haben können.

Ein Nachteil dieses Buches ist das Problem, daß es nie auf dem aktuellen Stand sein kann. Günstig wäre es auch gewesen, eine abschließende tabellarische Gegenüberstellung der geprüften Programme abzudrucken, was einen besseren Vergleich der einzelnen Programme vereinfacht hätte. Außerdem fehlen Tests zu den Textverarbeitungsprogrammen von integrierter Software völlig. Doch ich meine, die Vorteile des Buches überwiegen die Nachteile bei weitem. Ich finde es auf jeden Fall eine sehr gute und nachahmenswerte Idee auch für andere Programmbereiche.

Monika Ohlfest

Das kann der Schneider PC

Von Bartel/Stein
Verlag Data Becker
173 Seiten, 29.-DM
ISBN 3-89011-250-1

Kaum hat Schneider seinen PC der Presse vorgestellt, kommt Data Becker auch schon mit dem ersten Buch auf den Markt. Damit setzt der Düsseldorfer Verlag seine Reihe von Einsteigerbüchern (früher Premierenbücher) unter neuem Reihentitel fort. Hier sollen dem Einsteiger umfassende Informationen über den zur Auswahl stehenden Computer geboten werden und somit bei der Entscheidung zum Kauf helfen.

Der hier vorliegende Band gliedert sich grob in die drei Teile Einblicke, Hauptteil, Nachschlag. Der erste Abschnitt liefert in knapper, tabellarischer Form einen ersten Eindruck vom Schneider-PC. Die Autoren stellen hier den Soll-Zustand der Personal Computer dem Ist-Zustand des Schneider-PCs entgegen. Eingegangen wird hier auf Hardware (Tastatur, Monitor, Innereien und Umgebung) sowie Software (Betriebssysteme und mitgelieferte Programme). Dieselben Themen werden im Hauptteil noch einmal ausführlich behandelt. Die Autoren erklären dem Leser in jedem Kapitel zunächst einmal die allgemeinen Begriffe und gehen dann aus-

fürlich auf den Schneider-PC ein. Der Computerneuling bekommt so erst einmal Basiswissen vermittelt, lernt dann den eigentlichen Rechner kennen und erhält zum Schluß noch ein paar praktische Anwendungen (Programmierung unter Basic oder Benutzung von MS-DOS). Das abschließende Kapitel des Hauptteils ist der "Einkaufsbummel", der dem Leser direkt bei der Kaufentscheidung helfen will: Worauf ist zu achten; was für Software gibt es; wieviel Geld muß für welche Ausstattung angelegt werden? Der Nachschlag liefert zur Abrundung des Ganzen dann Benchmarktests, ein kleines Fachvokabular und ein Stichwortverzeichnis.

Diese Reihe von Data Becker ist für den absoluten Computerneuling, der sich für einen Schneider-PC interessiert, eine sinnvolle Entscheidungshilfe. Sie lobt den neuen Rechner nicht nur (wie man bei dem frühen Erscheinungstermin vielleicht vermuten könnte), sondern nennt auch Kritikpunkte (z.B. den Monitor). Für den schon etwas fortgeschrittenen Computerbenutzer, der umsteigen will oder den Schneider-PC schon besitzt, bietet dieses Buch meines Erachtens nicht viel Information, die er noch nicht kennt.

Schnelle Bücher haben bei Data Becker immer noch ihre Macken. Unverkennbar ist bei diesem Buch der Data-Welt-Stil, manchmal vielleicht etwas zu flapsig, aber sonst ganz gut lesbar. Was aber die Korrektur der Texte angeht, scheint dies nicht ganz so ernst genommen zu werden: Druckfehler, falsche Bindestriche und ein doppelt gesetzter Abschnitt. (Ab und zu soll es das auch bei uns geben.) Trotzdem kann man sagen, daß das Buch momentan, wo in den Zeitschriften nur knappe Wald- und Wiesentests stehen, dem Leser einige interessante Informationen liefert.

Monika Ohlfest



Assemblerkurs für den PC

In diesem Heft beginnen wir unseren Assemblerkurs für den Schneider-PC-Prozessor 8086

Warum Assembler?

Zuerst ein Vergleich zwischen Basic und Assembler. Basic ist leicht zu erlernen und verfügt über mächtige Befehle. Hier erreicht man z.B. den Ausdruck eines Textes mit PRINT"Text". In Assembler sind dagegen zahlreiche Anweisungen nötig, deren Anzahl sich mehr als verdoppelt, wenn man den Text an einer bestimmten Stelle des Bildschirms positionieren will. Hinzu kommt, daß Assembler schwieriger zu erlernen ist und Kenntnisse der Hardware und des Systems voraussetzt, um die man sich bei Basic nicht zu kümmern braucht. Es sind auch praktisch alle Probleme in Basic lösbar. Warum sollte man also in Assembler programmieren?

Nun, dafür gibt es mindestens zwei gute Gründe: Zum einen kommt man bei Betriebssystemerweiterungen an Assembler nicht vorbei, und zum anderen ist Assembler dort unentbehrlich, wo es auf ein genau definiertes Zeitverhalten des Programms ankommt, also bei der sogenannten Echtzeitverarbeitung. Diese tritt bei Steuer- und Regelvorgängen auf.

Widmen wir zuerst der Hardware unsere Aufmerksamkeit. Bekannt ist aus dem Bereich der Homecomputer, daß sie mit 8-Bit-Prozessoren ausgerüstet sind. Was heißt das? Sehen wir uns den prinzipiellen Aufbau eines Computers etwas genauer an. Er muß mindestens zwei Komponenten enthalten, nämlich einen Teil, der in der Lage ist, Daten zu verarbeiten und Befehle auszuführen, und einen Teil, der diese Daten und Befehle enthält. Den ersteren nennt man "Zentrale Verarbeitungseinheit" (CPU) und den anderen Speicher (Memory).

Zwischen beiden muß es eine Verbindung geben, die in der Lage ist, Daten und Befehle von der CPU im Speicher schreiben und lesen zu lassen. Dazu muß der CPU bekannt sein, wo sich im Speicher die gesuchte Information befindet. Jede Speicherstel-

le wird wie jedes Haus in einer Stadt mit einer Adresse versehen. Dabei werden die Speicherstellen fortlaufend mit einer Nummer versehen, also von 0 bis z.B. 65535. Alle diese Zahlen lassen sich im Dualsystem mit 16 Ziffern darstellen. Jede dieser Dualziffern nennt man ein Bit (Binary digit = Binärziffer). Dieses kann nur die Werte 0 oder 1 annehmen. Man hat sich darauf verständigt, jeweils 8 Bit zu einer Speicherstelle, zu einem Byte, zusammenzufassen.

Um also alle Speicherstellen adressieren zu können, benötigt man zwei Bytes. Technisch macht man das mit einem Leitungsbündel, das für jedes Bit genau eine Leitung enthält, in unserem Beispiel also 16 Stück. Da hiermit Speicherstellen adressiert werden, nennt man dies auch "Adreßbus", weil hier die Adressen "transportiert" werden. Genauso werden auch die Daten auf dem "Datenbus" transportiert. Da Prozessoren wie der 6502, 8080 oder Z80 nur 8 Bit gleichzeitig verarbeiten können (trifft beim Z80 nur bedingt zu), übernehmen sie die Daten nur byteweise; d.h., der Datenbus besitzt nur 8 Leitungen, er ist, wie man sagt, 8 Bit breit. Nun fehlt uns nur noch der Steuerbus. Er weist die CPU an, ob sie aus einer Speicherstelle lesen oder in sie schreiben soll.

Worin unterscheiden sich nun 8-Bit-, 16-Bit- und 32-Bit-Prozessoren? Hier gibt es zwei verschiedene Definitionen. Die erste bezieht sich auf die Breite des Datenbus. Hiernach sind der Z80, der 6502 und der 8088 in jedem Falle 8-Bit-Prozessoren, der 8086, 80186, 80286 und der 68000 wären 16-Bit-Prozessoren und der 68020 ist ein 32-Bit-Prozessor. Die andere Definition geht von der Größe der internen Speicher, der Register aus. Danach zählt der 8088 zu den 16-Bit-, der 68000 zu den 32-Bit-Prozessoren. Aus diesem Grunde sagt man zum 8088 auch, er sei ein 8/16-Bit-Prozessor. Der im Schneider-PC eingesetzte 8086-

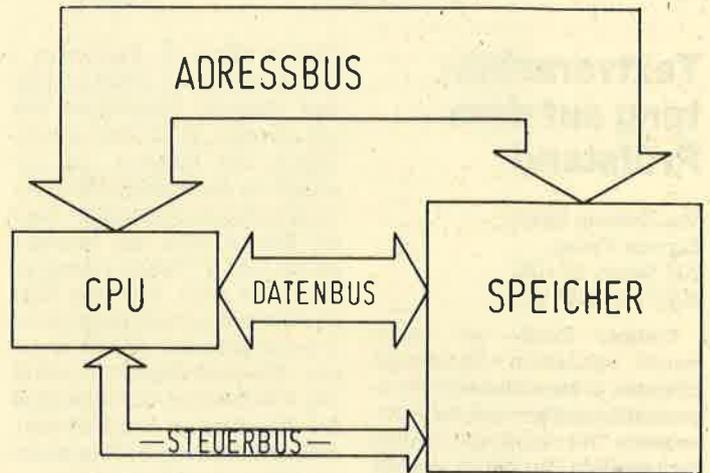


Bild 1: Prinzipieller Aufbau eines Computers

15	AH	7	AL	0
	BH		BL	
	CH		CL	
	DH		DL	

- AX: AKKUMULATOR
- BX: BASISREGISTER
- CX: ZÄHLERREGISTER
- DX: DATENREGISTER

15	SP	0
	BP	
	SI	
	DI	

- STAPELZEIGER
- BASISZEIGER
- QUELLINDEX
- ZIELINDEX

15	CS	0
	DS	
	SS	
	ES	

- CODESEGMENT
- DATENSEGMENT
- STAPELSEGMENT
- EXTRASEGMENT

15	IP	0
----	----	---

- BEFEHLSZEIGER

15	FLAGS	0
----	-------	---

- STATUSREGISTER

Bild 2: der Registersatz des 8086

Prozessor ist nach beiden Definitionen ein 16-Bit-Prozessor.

Wenden wir uns jetzt wieder dem Adreßbus zu. Bei den 8-Bit-Prozessoren ist er 16 Bit breit. Man braucht zur Belegung des Busses also nur zwei 8-Bit-Register. Beim Z80 sind dazu eine Reihe von Doppelregistern vorgesehen, die zusammen angesprochen werden können. Mit 16 Bit können $2 \text{ hoch } 16 = 65536$ Byte (64 KByte) adressiert werden. Will man mehr Speicher verwalten, so muß ein zusätzlicher Aufwand (Bankswitching) betrieben werden.

Wie sieht die Sache nun beim 8086 aus? Er hat intern 16-Bit-Register. Der Adreßbus enthält aber 20 Leitungen. Mit einem Register kann man also den Bus nicht belegen, aber zwei Register ergeben zusammen 32 Bit, also zuviel! Der Prozessor enthält nun nicht, wie man vermuten könnte, ein spezielles 20-Bit-Register oder ein zusätzliches 4-Bit-Register. Um diese 20-Bit-Adresse zu generieren, muß man auf einen Trick zurückgreifen. Man kann, wie wir oben gesehen haben, mit 16 Bit 64 KByte adressieren. Man benutzt nun ein Register dazu, den Anfang eines 64-KByte-Blockes anzugeben und ein anderes Register, um den Abstand von diesem Blockanfang anzuzeigen. Man nennt übrigens einen solchen 64-KByte-Block Segment und den Abstand zum Segmentanfang Offset. Die physikalische Adresse, also die 20-Bit-Adresse, errechnet sich nun folgendermaßen:

Physikalische Adresse = Segmentadresse * 16 + Offset

Rechnen wir einige Beispiele durch. Die Segmentadresse sei 10A0h (h bezeichnet eine hexadezimale Zahl) und der Offset 2222h. Wir erhalten $10A0h * 10h = 10A00h$. ($10h = 16$ dezimal), addieren 2222h: $10A00h + 2222h = 12C22h$. Man sieht sofort, daß sich diese physikalische Adresse auch aus der Segmentadresse 12A0h und dem Offset 0222h oder der Segmentadresse 1000h und dem Offset 2C22h zusammensetzen läßt. (Rechnen Sie es bitte nach!)

Das bedeutet, daß nur die physikalische Adresse eindeutig bestimmt ist, das Paar aus Segmentadresse und Offset ist es dagegen nicht. Vernünftigerweise wird man die Segmentadressen von 0000h dann bei 1000h, 2000h usw. anfangen las-

sen. Um klarzumachen, wo das Segment beginnt, gibt man die Adressen als Paar folgendermaßen an

Segmentadresse : Offset

Bei unserem ersten Beispiel wäre dies also 10A0h : 2222h. Diese Art der Adressierung nennt man "segmentierte Adressierung" im Gegensatz zur direkten Adressierung beim Z80, 68000 oder 68020. Der 80286 erzeugt seine 24-Bit-Adresse jedoch etwas anders. Im Register, das beim 8086 die Segmentadresse enthält, steht beim 80286 ein Zeiger auf eine Descriptorentabelle. Diese enthält neben anderen Einträgen wie Zugriffsberechtigungen und Type die 24-Bit-Adresse des Segmentanfangs, zu der der Offset direkt addiert wird, um die physikalische Adresse zu erhalten.

Soviel zu diesem Thema. Wenden wir uns nun speziell den Verhältnissen beim 8086 und dem PC zu.

Wie ist der Speicher organisiert

Beginnen wir die Speicherbetrachtung ab Adresse 0000h. Die ersten 1024 Bytes enthalten die Interruptvektoren, die bei der Systemkontrolle eine wichtige Rolle spielen. Nach ihnen folgt das DOS, also das Betriebssystem. Hier sind auch die einzelnen Gerätetreiber zu finden und die vom System benutzten Puffer. Dann folgt der Speicherbereich für den Anwender, der unterbrochen wird durch den Bereich, der für den Bildwiederholungspeicher vorgesehen ist. Er liegt in der Regel ab Adresse B000h : 0000h und ist bei Text 4 KByte, bei Grafik 32 KByte groß. Bei einigen Grafikkarten wird der Grafikbereich ab B800h : 0000h eingerichtet. Für den Bildwiederholungspeicher sind also insgesamt 64 KByte im Bereich von B000h : 0000h bis B000h : FFFFh eingerichtet.

Oberhalb davon befindet sich noch ein Teil des Kommandoprozessors, der von der Diskette an diese Stelle geladen wird. Er kann vom Anwenderprogramm überschrieben werden, wird aber nach dessen Ende wieder nachgeladen. Am oberen Ende des Speichers befindet sich der Reset-Vektor, der den Systemstart steuert. Seine Lage ist durch den Mikroprozessor bedingt und von der Hardware abhängig.

Das Programmiermodell des 8086 Mikroprozessors

Unter einem Programmiermodell versteht man eine symbolische Anordnung der Register des Prozessors. Man kann beim 8086 prinzipiell vier Registertypen unterscheiden, die alle 16 Bit breit sind. Da gibt es zuerst die Universal- oder Datenregister. Sie werden mit AX, BX, CX und DX bezeichnet. AX ist der Akkumulator, ein Register mit ausgezeichneter Bedeutung. Nur darüber können Multiplikationen und Divisionen durchgeführt werden. BX nennt man das Basisregister, denn es wird zu besonderen Adressierungen genutzt. CX ist das Zählerregister (Counter register) und findet speziell bei Schleifen seine Verwendung. DX letztlich ist das Datenregister. Alle Register lassen sich für Additionen/Subtraktionen usw. universal verwenden.

Eines unterscheidet diese Register von allen anderen: Man kann sie byteweise ansprechen. Z.B. spricht AL das niederwertige Byte (Bit 0 bis 7) an, während AH das höherwertige Byte (Bit 8 bis 15) betrifft. Entsprechendes gilt auch für BH, BL, DH, DL, CH, CL.

Die zweite Sorte von Registern sind die Zeiger- und Indexregister. Sie bestehen aus dem Stapelzeiger SP, dem Basiszeiger BP, dem Quellindex (Source-Index) SI und dem Zielindex (Destination Index) DI. Alle diese Register werden zu verschiedenen Adressierungsarten verwendet, wobei SP und BP mit dem Stapel zu tun haben.

Drittens gibt es die Segmentregister. Sie teilen sich auf in das Codesegment CS, in dem der Programmcode steht, dem Datensegment DS mit den Daten, dem Stapelsegment SS und einem Extrasegment ES. In diesen Registern steht die Anfangsadresse des jeweiligen Segments. Das Extrasegment wird übrigens bei Stringoperationen genutzt. Der Zugriff und der Umgang mit den Segmentregistern ist eingeschränkt. Wie - das werden wir noch sehen.

Als letztes wäre noch der Befehlszeiger IP (Instruction Pointer, sonst PC = Program Counter genannt) zu nennen. Das Registerpaar CS : IP zeigt immer auf den nächsten auszuführenden Befehl. Auch fehlt noch das Statusregister, das die Statuskenn-

zeichen (Flags) enthält. Damit kennen wir das gesamte Programmiermodell.

Der Assembler

Was macht ein Assembler? Da der Prozessor nur in der Lage ist, zwischen 0 = kein Strom und 1 = Strom an zu unterscheiden, müssen alle Befehle und Daten dem Prozessor in der Form von 0 und 1 dargeboten werden. Ein Programm hätte dann folgendes Aussehen:

```
1011000101110101010...
```

Es ist einzusehen, daß dieses bei der Eingabe sehr fehleranfällig wäre, da es ausreichen würde, eine 0 oder 1 zu vergessen. Aus diesem Grunde kam man sehr früh auf die Idee, ein Programm zu entwickeln, welches in der Lage ist, aus einer sprachlichen Form die maschinenlesbare zu erzeugen. Ein solches Programm nennt man allgemein Compiler und bei einer sehr maschinennahen sprachlichen Ebene heißt solch ein Compiler Assembler.

Während bei einer höheren Programmiersprache die Befehle aus ganzen Worten bestehen, sind sie in Assembler nur Kürzel (Mnemonics). Diese sehen z.B. so aus: ADD für +, SUB für -, MUL für * etc. Jeder Assembler kennt nur den Befehlssatz eines bestimmten Prozessors. So besitzt der 8086 92 verschiedene Befehle und 7 verschiedene Adressierungsarten. Die Kombination aller dieser Befehle und Adressierungsarten führt zu Tausenden von möglichen Befehlen, die alle als Kürzel eingegeben werden.

Wie gibt man Assemblerbefehle ein?

In Basic befindet man sich in einem Editor, der die Eingabe von Text oder Befehlen ermöglicht. Versucht man das gleiche mit dem Assembler, so erfährt man, daß der Assembler bereits einen fertigen Quelltext erwartet. Wie kommt man aber zu einem Quelltext? Man benötigt dazu einen Editor. Und was ist ein Editor? Es ist kurz gesagt ein Programm, das die Eingabe und die Bearbeitung von Texten aller Art zuläßt. Das können auch Textverarbeitungssysteme sein (z.B. WordStar, Word, Textomat o.ä.). Auf der Systemdiskette wird als Editor EDLIN mitgeliefert. Dabei handelt es sich um einen zeilenorientierten Editor, der die Eingabe und Bearbei-

tung von jeweils einer Zeile erlaubt. Näheres dazu in einem separaten Bericht. In vielen Fällen läßt sich auch ein Basic-Editor dazu mißbrauchen, sofern die Texte im ASCII-Format abgespeichert werden können. In neuerer Zeit gibt es dazu auch spezielle Editoren wie den Turbo- oder Norton-Editor. Letzterer hat allerdings die unangenehme Eigenschaft, nach dem letzten Befehl noch ein Carriage Return (CR) zu benötigen, damit der Assembler das Ende der Datei bzw. des Programms erkennen kann.

Das, was man nun als Programm in den Editor eingibt, nennt man Quellprogramm, während man das maschinenlesbare, aber noch nicht ausführbare Programm als Objektprogramm bezeichnet. Um aus einem Objektprogramm ein ausführbares Programm zu machen, muß noch ein weiteres Programm (der sog. Linker) dieses Objektprogramm bearbeiten. Erst nach dem Linken entsteht ein ausführbares Programm.

Der Assembler, mit dem wir uns in der Folge beschäftigen wollen, ist der Macro-Assembler MASM von MicroSoft. Der Zusatz "Macro" deutet an, daß es bei diesem Assembler möglich ist, eine Menge von Befehlen unter einem Namen aufzurufen. Das spart sehr viel Tipparbeit, hat aber auch seine Nachteile, auf die wir an gegebener Stelle noch eingehen werden.

Im allgemeinen wird ein assembliertes und gelinktes Programm nicht sofort in der gewünschten Weise laufen, wie es sich der Programmierer gedacht hat. Vom Basic her ist man es gewohnt, in einem solchen Fall das Programm an beliebiger Stelle anzuhalten und sich den Inhalt

der Variablen anzusehen oder mit einem Trace-Programm zu überprüfen. Das ist in Assemblerprogrammen nicht so einfach möglich. Man ist gezwungen, dazu ein separates Programm zu verwenden, den sogenannten Debugger. Er ist auf der Systemdiskette als DEBUG vorhanden. Der Umgang mit ihm wird in einem separaten Bericht näher erläutert. Daher gehen wir hier nicht näher darauf ein.

Schritte zur Entwicklung eines Assemblerprogramms

Man kann die Entwicklung eines Assemblerprogramms in folgende Schritte einteilen:

1. Entwurf einer Programmkonzeption anhand eines Ablaufplans oder Struktogramms.
2. Entwurf des Programms als Quellcode auf Papier.
3. Eingabe mittels eines Editors.
4. Assemblierung und evtl. Korrektur desselben, wenn der Assembler Fehler meldet.
5. Linken, um eine ablauffähige Version zu erhalten.
6. Ausführung des Programms mit Kontrolle, ob es in der gedachten Weise läuft.
7. Falls es nicht in der erwarteten Weise läuft, Programm mit dem Debugger testen und Korrektur der Fehler.

Im nächsten Heft

Im nächsten Schneider-Magazin führen wir unseren 8086-Assemblerkurs fort und zeigen erste Anwendungen für den PC 1512.

Peter Jaguttis

Die alte Zeile läßt sich dabei ganz oder teilweise für den Aufbau der neuen Zeilen verwenden. Um auf den Inhalt des Eingabepuffers gezielt zugreifen zu können, sind 5 Funktionen vorhanden, die mit den Funktionstasten F1 bis F5 aufgerufen werden können. Allerdings funktionieren diese in der nachstehend beschriebenen Weise nur bei DOS und im Zeileneditor EDLIN; in Anwenderprogrammen können die gleichen Tasten völlig andere Funktionen auslösen.

F1 – COPY1-Taste

Die Funktionstaste F1 kopiert ein Zeichen aus dem Eingabepuffer in die gerade bearbeitete Zeile. Ein Beispiel soll die Wirkungsweise dieser Funktion verdeutlichen: Der Inhalt einer ganzen Diskette soll auf eine zweite Diskette gesichert und danach die Kopie mit dem Original verglichen werden. Dazu wird zuerst die Zeile "DISKCOPY A: B:" eingegeben. Nachdem der Kopiervorgang abgeschlossen ist, müßte man zum Vergleichen die Zeile "DISKCOMP A: B:" einge-

ben. Einfacher geht es, wenn man stattdessen 6mal hintereinander F1 betätigt (in der neuen Zeile steht jetzt "DISKCO") und danach die Buchstaben "MP" eingibt (neue Zeile: "DISKCOMP"). Anschließend ist wieder die Taste F1 zu betätigen, bis die neue Zeile vollständig ist. Am besten, Sie versuchen dieses Beispiel gleich einmal selbst. Dabei wird deutlich, daß die Zeichenpositionierung des Eingabepuffers in die neue Zeile übernommen wird. Das bedeutet im Klartext: Wenn der Cursor in der neuen Zeile an der 5. Spaltenposition steht, dann wird auch das 5. Zeichen aus dem Eingabepuffer durch F1 übernommen.

F2 – COPYUP-Taste

Die Funktionstaste F2 kopiert alle Zeichen des Eingabepuffers bis zum anschließend angegebenen Zeichen. Auch hierzu eignet sich das schon angeführte Beispiel. Im Eingabepuffer befindet sich wieder die Zeile "DISKCOPY A: B:". Statt wie vorher 6mal hintereinander F1 zu benutzen, kann durch F2 und die Eingabe

Die flinken Helfer auf der Tastatur

Das Betriebssystem MS-DOS halt zur Unterstützung bei der Eingabe von Kommandozeilen acht Funktionstasten bereit. Diese Tatsache ist vielen Benutzern nicht bekannt, obwohl die sogenannten DOS-Editiertasten bei richtiger Anwendung viel Zeit und Tipparbeit ersparen. Bevor wir jedoch auf deren Funktionsweise eingehen, sind einige Vorbemerkungen nötig. Jede Zeile,

die über die Tastatur eingegeben und mit RETURN abgeschlossen wird, kommt zunächst in einen Eingabepuffer. Dann wertet das DOS die Eingabezeile aus und leitet die entsprechenden Aktionen ein. Der Inhalt des Eingabepuffers bleibt auch nach der Ausführung der Aktion erhalten und kann zusammen mit den DOS-Editiertasten für die Eingabe neuer Zeilen genutzt werden.

PROGRAMM	
1	
2	
3	
4	
5	
6	
7	
8	
9	
0	

des Buchstabens "P" der gleiche Effekt erzielt werden (neue Zeile: "DISKCO"). Der Rest der neuen Zeile wird dann wie oben erzeugt.

F3 – COPYALL-Taste

Mittels der Funktionstaste F3 wird der Eingabepuffer von der jeweiligen Cursorposition an in die neue Zeile kopiert. Damit läßt sich unser Beispiel zur Perfektion treiben. Die Zeile im Eingabepuffer wird nun nach folgendem Muster editiert: Zuerst wird F2 betätigt und der Buchstabe "P" eingegeben (neue Zeile: "DISKCO"), danach entsteht durch die Eingabe von "MP" die Zeile "DISKCOMP". Durch die F3-Taste wird die neue Zeile vervollständig.

F4 – SPIPUP-Taste

Eine Umkehrung der Funktionstaste F2 bildet die Taste F4. Sie überspringt alle Zeichen des Eingabepuffers bis zum angegebenen Zeichen. Auch hierzu ein kleines Beispiel: Nachdem die

Zeile "DIES IST EIN BEISPIEL" eingegeben ist, soll die Zeile "EIN BEISPIEL HILFT VERSTEHEN" erzeugt werden. Zuerst muß also der Teil "DIES IST" im Eingabepuffer übersprungen werden. Das erreicht man durch Drücken der F4-Taste und der nachfolgenden Eingabe eines "E". Beim ersten Mal werden die beiden vorderen Buchstaben des Eingabepuffers übersprungen, denn auch hier steht ein "E", nach der zweiten Eingabe ist das Ziel erreicht. Nun muß nur noch F3 betätigt werden, um den gewünschten Inhalt des Eingabepuffers in die neue Zeile zu kopieren.

Die eben behandelte Funktion ist ein wenig tückisch, da man das Ergebnis nicht sofort auf dem Bildschirm sehen kann. Man sollte deshalb an dieser Stelle sorgfältig arbeiten.

DEL – SPIP1-Taste

Eine ähnliche Funktion wie F4 hat die Taste DEL. Sie über-

springt einzelne Zeichen im Eingabepuffer und arbeitet bei kurzen Sprüngen effektiver als F4.

Eine weitere ebenfalls sehr nützliche Editiermöglichkeit wird durch die Funktionstaste F5 aufgerufen und bewirkt, daß die gerade bearbeitete Zeile in den Eingabepuffer kommt, ohne daß sie wie bei RETURN ans DOS als Befehlszeile geschickt wird. Den Vorteil dieser Funktion zeigt das nächste Beispiel: Es wurde die Zeile: "CAPY A: BEISPIEL.TXT B: BEISPIEL.BAK" eingegeben. Bevor RETURN gedrückt wird, bemerkt der Benutzer, daß es "COPY" statt "CAPY" heißen muß. Um sich die doppelte Tipparbeit zu ersparen, kann man F5 betätigen und die Zeile im Eingabepuffer speichern. Danach läßt sich durch Eingabe von "CO" und anschließendem Betätigen der F3-Taste die Zeile berichtigen. Ist eine Eingabezeile total mißlungen, so daß eine Bearbeitung mit F3 sich nicht lohnt, dann kann mit ESC (VOID-Taste) die gesamte Zahl gelöscht werden.

INS – INSERT-Taste

INS erlaubt Eingaben in der bearbeiteten Zeile, ohne daß sich der Positionszeiger im Eingabepuffer verschiebt. Steht im Eingabepuffer beispielsweise die Zeile "EIN BEISPIEL", so kann durch Betätigen der INS-Taste und nachfolgender Eingabe von "DIES IST" mit der F3-Taste die neue Zeile "DIES IST EIN BEISPIEL" erzeugt werden.

Wer die Beispiele nachvollzogen hat, wird erkannt haben, welch nützliches Werkzeug die DOS-Editiertasten darstellen. Damit die Zeit der Gewöhnung möglichst rasch vorübergeht, sollte man sich die Tastaturschablone in Abb. 1 kopieren, ausschneiden, die entsprechenden Felder mit den Tastenfunktionen beschriften und als Gedächtnisstütze verwenden. Die Schablone ist so aufgebaut, daß sie auch für Anwenderprogramme, die von den Funktionstasten Gebrauch machen, zu benutzen ist.

H.-P. Schwaneck

Ein oder zwei Laufwerke?

Viele Leser überlegen sich jetzt den Kauf eines PC 1512. Einige haben uns auch schon um Rat gefragt, welche Ausbaustufe sie denn nun kaufen sollen. Mal ganz davon abgesehen, daß es immer eine Frage des Geldbeutels ist, stehen zwei Entscheidungen im Vordergrund: monochromer Bildschirm oder Farbmonitor mit ein oder zwei Laufwerken (oder gar eine Festplatte?).

Bei der Wahl des Monitors muß sich jeder Anwender selbst ins Gewissen reden und prüfen, wofür er seinen PC nutzen will. Wer vor allem mit Textverarbeitungsprogrammen, Datenbanken, Programmiersprachen etc. auf dem PC arbeiten will (oder muß), wird sicherlich mit dem monochromen Monitor gut wenn nicht sogar besser als mit einem Farbmonitor fahren. Generell ist zu sagen, daß die Qualität von monochromen Monitoren (zumindest in dieser Preisklasse) besser ist als die von Farbmonitoren. Wer allerdings im grafischen Bereich seinen Schwerpunkt für die PC-Anwendung findet und viel unter GEM arbeiten will, sollte sich überlegen, ob er mit einer farbigen Darstellung nicht besser be-

dient ist. Der Aufpreis für einen Farbmonitor beträgt beim PC 1512 ca. 500.- DM.

Die Frage, ob ein oder zwei Laufwerke, läßt sich hingegen wesentlich eindeutiger beantworten. Um mit dem PC 1512 effektiv arbeiten zu können, sind unbedingt zwei Laufwerke notwendig. Viele gute Anwendungsprogramme verlangen einfach zwei Laufwerke. Darunter läuft nichts. Bei anderen Programmen (z.B. Compilern) wird die Handhabung mit nur einem Laufwerk durch penetrantes Diskettenwechseln zur Qual. Anmerken muß man allerdings, daß zur Zeit immer mehr Programme erhältlich sind, die mit nur einem Laufwerk auskommen. Der Aufpreis für ein zweites Laufwerk beträgt ebenfalls 500.- DM. Hier sollte man nicht sparen. Lieber ein PC mit monochromem Monitor und zwei Laufwerken, als mit einem Farbmonitor und nur einem Laufwerk (beide kosten 2499.- DM).

Die Versionen mit Festplatte sind zur Zeit noch nicht erhältlich. Wenn schon eine Festplatte, dann aber eine mit (mindestens) 20 MByte (10 MByte sind nichts

Ganzes und nichts Halbes). Der Preisunterschied zwischen 10- und 20-MByte-Platten wird auch nicht so gravierend sein. Man sollte sich auch überlegen, ob man eine Festplatte nicht von einem Fremdhersteller beziehen kann. Vielleicht lassen sich so ein paar Hunderter sparen. Hier wird sich in der nächsten Zeit sicherlich einiges tun.

Wie gesagt, die Entscheidung, welche Ausbaustufe man neh-

men soll, ist maßgeblich vom Geldbeutel abhängig. Empfehlen würde ich den PC 1512 mit zwei Laufwerken und einem monochromen Monitor (2499.- DM). Wer nicht soviel Geld hat, sollte mit der einfachsten Ausbaustufe anfangen und sich vom Weihnachtsmann ein Zweitlaufwerk wünschen.

Manfred Walter Thoma

Gute Software muß nicht teuer sein!

Für 298.-DM bietet das Düsseldorf Softwarehaus Data-Becker jetzt auch das altbekannte Textverarbeitungsprogramm "Textomat" (C 64, C 128, Schneider CPC, Atari ST) für Personalcomputer (IBM PC oder Kompatible) an. Im Lieferumfang sind eine nicht kopiergeschützte Diskette mit dem Textverarbeitungsprogramm und einer kleinen integrierten Adreßverwaltung, das 178 Seiten starke Handbuch im Data-Becker-typischen rotweißen Ringordner und

zwei doppelseitig bedruckte DIN-A5-Blätter (Blockmarkierungen, Cursorsteuerungen, Steuerzeichen und Menüauswahlübersicht) enthalten. "Textomat" kann ca. 10 DIN-A4-Seiten Text und ca. 400 Datensätze in der Adreßverwaltung aufnehmen.

Installation

Vor dem eigentlichen Start mit "Textomat" muß man das Programm erst einmal installieren, d.h. für die eigene Gerätekonfi-

guration vorbereiten. Dazu formatiert man zuerst eine neue Diskette, die zur neuen installierten Textomat-Diskette werden soll. Dann legt man die Original-Textomat-Diskette in Laufwerk A und startet die Installation mit dem Befehl `INSTALL <ENTER>`. Es erscheint ein Bildschirmfenster mit den Punkten: Bildschirm (monochrom oder Farbe), Laufwerk (A, B oder Festplatte als Standardlaufwerk) und Drucker (Bestimmung des eigenen Druckers). Folgende Drucker sind schon vorgegeben: Gabriele 9009, Epson FX, Star NL 10, Taxan K-P, HR-X5, NEC 02-P3 und Epson RX. Es können aber auch andere Drucker selbst installiert werden (Menüpunkt Druckparameter). Wer einen Farbmonitor besitzt, kann die Farben für Hintergrund, Vordergrund I und Vordergrund II selbst wählen. Nachdem diese Punkte beendet sind (Installation beenden), kopiert "Textomat" seine installierte Version auf die vorbereitete Diskette. Für dieses Verfahren braucht man etwas Geduld, weil 9 Dateien kopiert werden müssen. Danach kann mit der eigentlichen Arbeit begonnen werden.

Schreiben mit Textomat PC

Nach Eingabe des Ladebefehls `TEXTOMAT` erscheint nach kurzer Zeit das Titelbild des Programms und einen kleinen Augenblick später ist man direkt im Schreibmodus. Diesen kann man mit dem Menüpunkt `OPTIONEN` noch verändern. Z.B. kann die Menüzeile ein- bzw. ausgeblendet und die Zeilen im Eingabefeld mit Punkten in zwei Stärken oder ohne dargestellt werden. Auch ist es möglich, einen Klingelton wie bei der Schreibmaschine zu erhalten, wenn die Zeile zu Ende ist.

Das Schreibfenster selbst stellt 80 Zeichen in 21 Zeilen zur Verfügung. Sind die 80 Zeichen geschrieben, wird automatisch in die nächste Zeile gesprungen (natürlich mit Wordwrapping). Sind die 21 Zeilen voll, scrollt der Bildschirm immer nur um eine weitere Zeile nach oben. In welcher Zeile/Spalte man sich gerade befindet, zeigt das Zeilenlineal an, ebenso die fünf möglichen Indikatoren (Merker gesetzt, Nummernblock auf Zahleneingabe geschaltet, Einfügemodus an, Caps-Lock an und Scroll-Lock an). Die Zeilenumschaltung findet über die Tasten `SHIFT` und

`ENTER` statt, was etwas gewöhnungsbedürftig ist.

In den Menümodus gelangt man mit der `F1`-Taste, worauf in der untersten Bildschirmzeile die Hauptauswahl erscheint. Die einzelnen Menüpunkte kann man wahlweise durch den großgedruckten Buchstaben (meistens Anfangsbuchstabe) oder die Cursortasten auswählen. Nach Aufruf der Unterprogramme werden Fenster für die Bearbeitung über den Text gelegt. Hier findet man weitere Informationen und kann Eingaben tätigen. Aus allen Menüs und Untermenüs kann jederzeit mit der `ESC`-Taste in den Schreibmodus zurückgekehrt werden. Die Menüführung ist sehr gut gelöst (ähnlich wie bei "Word" oder "Multiplan") und nicht ballastartig übertrieben.

Beim Schreiben fällt sofort auf, daß der Text unformatiert auf dem Bildschirm erscheint. Außer dem Wordwrapping bei der eingestellten Zeilenbreite werden alle weiteren Textformate (fett, unterstrichen etc.) nicht auf dem Bildschirm dargestellt. Will man z.B. einen Text zentriert erstellen, muß dies über eine Steuersequenz angegeben werden. Der Text wird jetzt allerdings nicht zentriert auf dem Bildschirm dargestellt, sondern nur zentriert ausgedruckt (bzw. im `ZEIGEN`-Modus angezeigt). Alle Optionen für die Textausgabe – wie breit, fett, hochgestellte Schrift etc. – müssen über diese gewöhnungsbedürftigen Steuersequenzen aktiviert werden. Hier ist einer der Hauptkritikpunkte an "Textomat PC" zu finden. Bei Textverarbeitungsprogrammen der PC-Klasse kann man eine Darstellung des Textes, wie er später ausgedruckt wird, erwarten.

"Textomat PC" verfügt über alle Operationen, die bei einem Textverarbeitungssystem notwendig sind (Einfügen, Löschen, Ersetzen, Block verschieben etc.). Alle Operationen werden sehr schnell und präzise ausgeführt. Die Schnelligkeit ist zwar beeindruckend, aber für PC-Textsysteme, die einen Text unformatiert auf dem Bildschirm darstellen (zumal sich der gesamte Text im Arbeitsspeicher befindet), als normal zu bezeichnen.

Adreßverwaltung und Serienbrief

"Textomat PC" bietet integriert eine Adreßverwaltung an,

die sich direkt vom Hauptmenü aufrufen läßt (`ADRESSEN`) und ca. 400 Datensätze verwalten kann. Die Eingabemaske besteht aus 6 Datenfeldern (Anrede 4 Zeichen, Name 29 Zeichen, Straße 24 Zeichen, Postleitzahl 4 Zeichen, Ort 19 Zeichen, Zusatz 29 Zeichen). Ein eigenes Feld für die Telefonnummer ist z.B. nicht vorgesehen, d.h., dafür müßte man das Feld für den Zusatz nehmen. Diese Eingabemaske ist nicht mehr veränderbar.

Die Adreßverwaltung kann man immer innerhalb eines Textes aufrufen, ohne daß der alte Text erst gespeichert werden müßte. Ist die Arbeit in der Adreßverwaltung beendet (`QUITT`), kommt man wieder in den gerade bearbeiteten Text zurück, allerdings ganz an den Anfang und nicht auf die Cursorposition, wo man gerade war.

Für so eine relativ kleine Adreßverwaltung sind doch eine ganze Menge Funktionen vorhanden. Daß man eine Datei anlegen, korrigieren, speichern und laden kann, ist eine Selbstverständlichkeit, aber diese kleine Verwaltung bietet zusätzlich noch die Möglichkeit, nach jedem Begriff in der Datei zu suchen (auch nach Anfangsbuchstaben, allerdings nur in der eingegebenen Schreibweise). Es können Adreßaufkleber gedruckt werden, und die Datei läßt sich für Serienbriefe vorbereiten. Dabei ist allerdings zu beachten, daß man in der Serienbrieffunktion keine bestimmten Postleitzahlenbereiche auswählen kann (keine bedingten Serienbriefe möglich). Dies muß man also schon vorher beim Anlegen der Adreßdatei beachten.

Für den privaten Anwender ist diese Adreßverwaltung sicher-

lich mehr als ausreichend, für die kommerzielle Nutzung ist sie aber zu einfach. Es fehlt die Möglichkeit, genügend Informationen in die vorgegebenen Datenfelder einzugeben (z.B. Telefon, Ansprechpartner, Kunden-Nr. etc.). Bei der Serienbrieffunktion fällt vor allem ein Fehlen des bedingten Druckens auf.

"Textomat PC" ist ein Textverarbeitungsprogramm, das m.E. auf den privaten Anwender zugeschnitten ist. Er wird mit dem Programm sicher alle seine Aufgaben zufriedenstellend lösen können. Für Anwender, die häufig und viel mit dem Computer schreiben, ist "Textomat PC" nur bedingt tauglich. Das liegt zum einen an der unformatierten Darstellung des Textes und zum anderen an der mangelnden Textkapazität. Mit ca. 10 DIN-A4-Seiten können zwar alle Briefe etc. geschrieben werden, bei längeren Schriftstücken treten jedoch Probleme auf. Auch ist es fraglich, warum nur ca. 50 KByte für den Text verfügbar sind. Heutige PCs haben in der Regel einen Arbeitsspeicher von 512 KByte, der für die Textfassung auch voll genutzt werden sollte. Die zusätzlich integrierte Adreßverwaltung und die Serienbrieffunktion zeigen nach professionellen Gesichtspunkten zwar Mängel, lassen sich aber im privaten Bereich problemlos anwenden. "Textomat PC" ist jedoch eines der wenigen Textverarbeitungsprogramme, die für PCs unter `MS-DOS` in dieser Preisklasse erhältlich sind. Unter diesem Gesichtspunkt ist "Textomat PC" sicherlich ein gutes Programm, kann sich allerdings mit professionellen Textsystemen wie z.B. "MS-Word" nicht messen (dafür kostet "MS-Word" auch das Fünffache).

Monika Ohlfest

Menü-Modus Zeile: 001 Spalte: 000 (Textomat PC)

Gute Software muß nicht teuer sein!
Textomat PC

←
Für nur 298,- DM bietet das Düsseldorf Softwarehaus Data Becker jetzt auch das altbekannte Textverarbeitungsprogramm Textomat (C-64, C-128, Schneider CPC, Atari ST) für Personalcomputer (IBM PC oder Kompatible) an. Im Lieferumfang sind eine nicht kopiergeschützte Diskette mit dem Textverarbeitungsprogramm und einer kleinen integrierten Adreßverwaltung, das 178 Seiten starke Handbuch im Data Becker typischen rotweißen Ringordner und zwei doppelseitig bedruckte DIN A 5 Blätter (Blockmarkierungs-, Cursorsteuerungs-, Steuerzeichen- und Menüauswahlübersicht) enthalten. Textomat kann ca. 10 DIN A 4 Seiten Text und ca. 400 Datensätze in der Adreßverwaltung verwalten. ←

MENU: Edit Formular Löschen Druck daTel Optionen Adressen Undo Hilfe Quitt

Das Hauptmenü von "Textomat PC", dem bekannten Textverarbeitungsprogramm aus dem Hause Data-Becker

Wir machen Nägel mit Köpfchen

**1000 DM
für das
Spiel
des
Monats**

**1000 DM
für die
Anwendung
des
Monats**



Wir beim CPC Magazin machen Nägel mit Köpfchen. Ab sofort gibt es deshalb nicht mehr das "Topprogramm des Monats", sondern wir bringen jetzt in jeder Ausgabe das "Spiel des Monats" und zusätzlich die "Anwendung des Monats". Jetzt haben also beide Programmier-Gruppen die gleiche Chance, auch ein eventueller Jackpot bleibt getrennt. Ist also mal kein Spielprogramm zum Abdruck geeignet, so gibt es trotzdem die "Anwendung des Monats" und beim nächsten Mal 2000.- DM für das "Spiel des Monats" und wieder 1000.- DM für die "Anwendung des Monats". Wer also auf dem einen oder anderen Gebiet ein besonders gutes Programm geschrieben hat, sollte sich diese Chance nicht entgehen lassen. Wer weiß, vielleicht gehören gerade Sie zu den nächsten Gewinnern. Zögern Sie nicht, machen Sie mit!

Senden Sie Ihre Programme an das CPC
Magazin, Postfach 1640, 7518 Bretten

Sieben auf einen Streich

An dieser Stelle wie immer 7 neue Grafik-Gags von unserem Programmierer Christoph Schillo

464

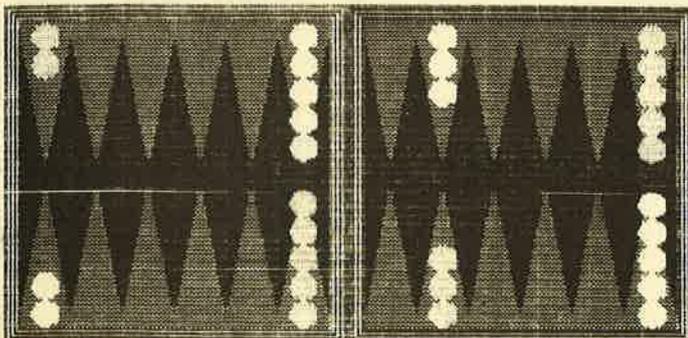
664

6128



Und weiter geht's in unserer Serie mit kleinen Malprogrammen. Ob Sie die 7 Teile einfach nur abtippen und starten oder in Ihre eigenen Programme einbauen oder sich zu eigenen Kreationen animieren lassen, ist wie immer Ihre eigene Entscheidung. Die letzte Programmzeile ist wie üblich eine Endlosschleife, um das Zerstören des Bildes durch die Ready-Meldung zu verhindern. Wenn Sie sich also sattgesehen haben, müssen Sie nur zweimal die ESC-Taste betätigen. Haben Sie das Listing in Ihr eigenes Programm eingebaut, so lassen Sie diese Zeile einfach weg oder ersetzen Sie durch ein RETURN, wenn Sie den Grafikgag als Unterprogramm verwenden. Wenn Sie die Teile nur anschauen wollen, so setzen Sie den Rechner jeweils mit CTRL-SHIFT-ESC zurück, bevor Sie eines der Bilder mit RUN "name" starten.

1. Dieses Bild bringt Sie in den Genuß des amerikanischen Sternenbanners.
2. Hier stehen weitere 14 Flaggen zu Ihrer Verfügung. Auf Tastendruck wird die jeweils nächste angezeigt.
3. Dieses Programm konstruiert ein Backgammon-Brett mit regelgerechtem Steinaufbau.
4. Hier können Sie beliebige Texte kreisförmig darstellen.
5. Programm Nr. 5 erzeugt etwas, das unschwer als Sanduhr zu erkennen sein wird.



6. Dieses Bild stellt eine Handgranate dar, was wohl damit zusammenhängt, daß ich momentan beim Bund bin.
7. Dieses Motiv ist dann schon wieder friedlicher und zeigt einen Wasserfall in Bewegung.

Christoph Schillo

Teil 1

```
1 ' Grafikgags Nr.11 Teil 1
2 '
100 MODE 1:INK 0,13:INK 1,26:INK 2,6:INK
3,2
110 p=1:FOR a=1 TO 13:ORIGIN 0,0,0,640,4
30-a*30,430-a*30-30:CLG p+1:p=ABS(p-1):N
EXT
120 WINDOW 1,19,1,13:PAPER 3:CLS
130 ORIGIN 0,0,0,640,400,0
140 FOR c=0 TO 1:d=c/2:FOR b=1+d TO 5-d:
FOR a=1+d TO 6-d:GOSUB 160:NEXT a,b,c
150 GOTO 170
160 PLOT a*50-32,420-b*44,1:DRAWR 8,20:D
RAWR 8,-20:DRAWR -16,12:DRAWR 16,0:DRAWR
-16,-12:RETURN
170 GOTO 170
```

Teil 2

```
1 ' Grafikgags Nr.11 Teil 2
2 '
100 FOR b=0 TO 13:READ d,e,f,a$
110 INK 0,13:BORDER 13:INK 1,d:INK 2,e:I
NK 3,f:MODE 1
120 FOR a=1 TO 3:WINDOW 1,40,a*8-7,a*8:P
APER a:CLS:NEXT:WINDOW 1,40,1,25:PAPER 0
:LOCATE 20-LEN(a$)/2,25:PRINT a$:
130 WHILE INKEY$="" :WEND:NEXT
140 DATA 0,6,15,Bundesrepublik Deutschla
nd,2,26,2,Argentinien,6,15,9,Bolivien,2,
0,26,Estland,9,24,2,Gabun,9,26,6,Iran,24
,9,6,Litauen,6,26,2,Luxemburg,6,26,1,Nie
derlande,6,26,6,Oesterreich,9,26,2,Sierr
a Leone,2,24,9,Sudan,6,26,9,Ungarn,0,26,
6,Volta
150 GOTO 150
```

Teil 3

```
1 ' Grafikgags Nr. 11 Teil 3
2 '
100 MODE 1:INK 0,0:BORDER 0:INK 1,6:INK
2,26:INK 3,9
110 p=1:FOR a=0 TO 1:FOR b=0 TO 5:FOR c=
0 TO 1:PEN p+1:WINDOW a*20+b*3+2,a*20+b*
3+4,7+c*10,14+c*10:PRINT STRING$(24,CHR$
(207));p=ABS(p-1+c):NEXT c,b,a
120 FOR b=0 TO 1:FOR c=0 TO 6:FOR a=-24
TO 24 STEP 2:PLOT 48*c+16+b*320,32,0:DRA
```

```

W a+48*c+16+b*320,160:DRAW 48*c+16+b*320
,288:NEXT a,c,b:a$=SPACE$(40):PAPER 3
130 WINDOW 1,40,1,25:LOCATE 1,6:PRINT a$
:FOR a=7 TO 24:LOCATE 1,a:PRINT " ":LOCAT
E 20,a:PRINT " ":LOCATE 40,a:PRINT " ";:N
EXT:PRINT a$;:FOR a=0 TO 1:FOR b=0 TO 4
140 p=ABS(p-1):c=310-b*4:PLOT 4+b*2+a*32
0,4+b*2,p:DRAWR c,0:DRAWR 0,c:DRAWR -c,0
:DRAWR 0,-c:NEXT b,a
150 DIM r(18,2):b=0:FOR a=0 TO PI STEP 0
.175:r(b,1)=SIN(a)*12:r(b,2)=COS(a)*12:b
=b+1:NEXT:PAPER 0:WINDOW 1,40,1,25
160 FOR a=1 TO 4:READ b,c,d:FOR e=-1 TO
1 STEP 2:FOR f=1 TO d:FOR g=0 TO 18:PLOT
r(g,1)+b,160+e*(r(g,2)+(6-f)*26),c+1:DR
AWR -2*r(g,1),0:NEXT g,f:c=ABS(c-1):NEXT
e,a:DATA 38,0,5,232,1,3,360,1,5,600,0,2
170 GOTO 170

```

Teil 4

```

1 ' Grafikgags Nr. 11 Teil 4
2 '
100 MODE 1
110 MODE 1:DEG:INK 0,0:BORDER 0:INK 1,25
:INK 2,15:INK 3,6
120 a$="Christoph Schillo":r=150:vx=1.25
:vy=vy:p=1:x=320:y=200:GOSUB 160
130 a$="proudly presents":r=100:p=2:vx=1
.3:vy=1.3:GOSUB 160
140 a$="a demonstration":p=3:r=50:vx=1.2
:vy=1.2:GOSUB 160
150 GOTO 190
160 LOCATE 1,25:PRINT a$;:FOR b=16 TO 0
STEP -2:FOR a=0 TO LEN(a$)*16 STEP 2:z=T
EST(a+2,b):IF TEST(a,b) THEN PLOT a,b,0:
PLOT SIN(a*vx)*(r+b*vy)+x,COS(a*vx)*(r+b
*vy)+y,p:IF z THEN DRAW SIN((a+2)*vx)*(r
+b*vy)+x,COS((a+2)*vx)*(r+b*vy)+y
170 NEXT a,b:LOCATE 1,25:RETURN
180 DATA 24,15,6,15,6,24,6,24,15
190 FOR b=1 TO 3:FOR a=1 TO 3:READ c:INK
a,c:NEXT:FOR d=1 TO 100:NEXT d,b:RESTOR
E:GOTO 190

```

Teil 5

```

1 ' Grafikgags Nr.11 Teil 5
2 '
100 MODE 1:INK 0,26:BORDER 26:INK 1,3:IN
K 2,15:INK 3,23
110 SYMBOL 255,60,94,191,191,191,189,90,
60:FOR a=2 TO 24:LOCATE 12,a:PRINT CHR$(
255);SPACE$(14);CHR$(255):NEXT
120 FOR a=PI/2 TO 2.5*PI STEP 0.03:PLOT
240-SIN(a)*50,64*(a-PI/2),3:DRAWR 2*SIN(
a)*50+120,0:NEXT
130 PAPER 1:LOCATE 11,1:PRINT SPACE$(18)
:LOCATE 11,25:PRINT SPACE$(18):PAPER 0
140 FOR a=1.5*PI TO 2*PI STEP 0.03:PLOT
248-SIN(a)*50,64*(a-PI/2),2:DRAWR 2*SIN(
a)*50+104,0:NEXT
150 FOR a=PI/2 TO 2.5*PI STEP 0.1:x=256+
12*(a-PI/2):PLOT x,32-SIN(a)*10:DRAW x,1
8:NEXT
160 FOR a=40 TO 200 STEP 8:PLOT 296+RND*
8,a,-2*(RND>0.5):NEXT
170 GOTO 170

```

Teil 6

```

1 ' Grafikgags Nr.11 Teil 6
2 '
100 MODE 1:INK 0,13:INK 1,0:INK 2,9:INK
3,6:BORDER 13
110 FOR a=0 TO PI/2 STEP 0.01:x=64*COS(a
):y=100*SIN(a):PLOT 320+x,200+y,2:DRAWR
-2*x,0:PLOT 320+x,200-y,2:DRAWR -2*x,0:N
EXT:c=100:FOR b=0 TO 64 STEP 32:GOSUB 16
0:NEXT:b=64:FOR c=0 TO 100 STEP 32:GOSUB
160:NEXT
120 LOCATE 18,6:PRINT CHR$(214):WINDOW 1
9,22,6,7:PAPER 1:CLS:PEN 3:PRINT " o":PAP
ER 0:PEN 1
130 FOR a=0 TO PI/2 STEP 0.03:PLOT SIN(a
)*25+200,64*a+102,1:DRAWR 16,0:PLOT 275-
COS(a)*50,64*a+202:DRAWR 16,0:NEXT
140 FOR a=0 TO 2*PI STEP 0.1:PLOT SIN(a)
*20+264,COS(a)*20+316:NEXT
150 GOTO 170
160 PLOT 320,200+100-b-c,1:FOR a=0 TO 2*
PI+0.1 STEP 0.3:DRAW SIN(a)*b+320,COS(a)
*c+200:NEXT:RETURN
170 GOTO 170

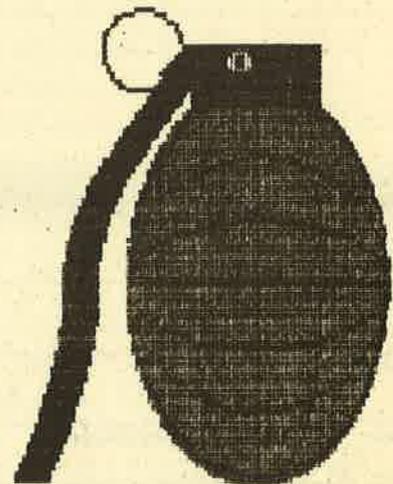
```

Teil 7

```

1 ' Grafikgags Nr.11 Teil 7
2 '
100 MODE 0:INK 0,12:BORDER 12:INK 14,1:I
NK 15,23:RANDOMIZE 1000
110 FOR a=0 TO 640 STEP 4:FOR b=0 TO 32.
STEP 4:PLOT a,b,INT(RND*14+1):DRAWR 0,2:
NEXT b,a
120 FOR a=240 TO 400 STEP 4:l=36
130 PLOT a,l:FOR b=1 TO 13:x=RND*16:DRAW
R 0,x,b:l=l+x:NEXT:IF l<400 THEN 130 ELS
E NEXT
140 WINDOW 1,40,1,4:PAPER 15:CLS:x=20:FO
R a=0 TO 236 STEP 4:PLOT a,336:DRAWR 0,x
,0:PLOT 640-a,336:DRAWR 0,x:x=x+INT(RND*
5)-2:NEXT
150 FOR a=0 TO 400:FOR b=13 TO 1 STEP -1
:r=INT(RND*13+1):INK r,5:INK b,26:CALL
&BD19:INK b,1:INK r,11:NEXT b,a
160 GOTO 160

```



Fingerschonend



"Fingerschonend" ist ein besonderer Service für unsere Leser. Wer die abgedruckten Listings nicht eintippen will, kann sie direkt auf Cassette oder Diskette bestellen, die es zu jedem Heft gibt. Zudem ist diese "fingerschonende" Einrichtung eine preiswerte Angelegenheit: 15.- DM kostet die Cassette und 25.- DM die Diskette. Wer unser Angebot nutzen will, kann den untenstehenden Bestellschein für seine Anforderung verwenden. Wir liefern umgehend per Vorauskasse (versandkostenfrei) oder per Nachnahme (+ 5.70 DM Porto + Versandkosten). Einfacher und preiswerter kann man kaum eine Programmsammlung aufbauen. Und wer sich sowieso zu jedem Heft die Cassette kaufen will, der kann gleich ein Cassetten-Abo machen, da ist das Ganze noch etwas billiger!

Software-Bestellschein

Ich bestelle aus dem CPC-Programmservice folgende Software:

Anzahl	Bestell-Nr.	Einzel-Preis	Ich wünsche folgende Bezahlung:
_____	_____	_____	<input type="checkbox"/> Nachnahme (+ 5,70 DM Porto + Versandkosten) <input type="checkbox"/> Vorauskasse (keine Versandkosten) Bei Vorauskasse bitte Scheck beilegen oder auf Postscheckkonto Karlsruhe 43423-756 überweisen
_____	_____	_____	
_____	_____	_____	
_____	_____	_____	
_____	_____	_____	
_____	_____	_____	

Cass.-Jahres-Abo 150.- DM Cass.-Halbjahres-Abo 75.- DM

Name des Bestellers

Anschrift - Straße

PLZ/Ort

Telefon

Datum/Unterschrift

Coupon ausschneiden, auf Postkarte kleben und einsenden:
Verlag Rätz-Eberle, Postfach 1640, 7518 Bretten

CPC-Magazin 12/85

Darts, Hexmonitor, Sprites, Kalender 464, Kalender 664/6128, Senso, Sasem (nur 464), Software-Uhr, Compressor, Expander, Player's Dream, Killer, Sichere Kennungszeile, Grafik Gags, Variablendump (nur 464).

CPC-Magazin 1/86

Grafik Gags, Extended Basic Teil 1, Breitschrift (nur 464), Examiner (nur 464), Datenverwaltung, Discdoctor, Showdown 464, Showdown 664/6128, Lotto.BAS (nur mit Laufwerk), CPC-Orgel.

CPC-Magazin 2/86

Eingabe einer Funktion per INPUT, Busy-Test, DATAGEN, BAS, PROGHELP, Zeichenvergrößerung, Jump Over, Extended Basic 2, 7 Grafikgags, Director, DIN-Tast., Hex-Tast., Pro-Safe 2.0, Pingo, Pingo-editor, Etikett.BAS, List#8".

CPC-Magazin 3/86

Discmon, Discrsx,
Demo 1, Demo 2,
Mini-Monitor, Sieben
auf einen Streich,
Calc, Mathe CPC,
Painter, Screener,
Periodensystem.

CPC-Magazin 4/86

Sieben auf einen
Streich, Statistik,
Tape cleaner, String-
suche, Unzialschrift,
Input, Baudcopy,
HI-Dump, Fehler-
routine, Sepp im
Paternoster, Puzzle.

CPC-Magazin 5/86

Bücherdatei, Sieben
auf einen Streich,
FQuader, Window,
XBOS, Trickfilm-
grafik + Demo (läuft
auf 464 und 664 mit
vortex-Speicherer-
weiterung), Sort
(nur 464), Elektra-
CAD, Life, Zentus.

CPC-Magazin 6/86

Asso, Sieben auf
einen Streich, Scroll-
bremse (464), Scroll-
bremse (664/6128)
Notizblock, Super-
grafik, Copy?? Right!!
V.2.0, Hello (464 +
Vortexlaufwerk),
Puzzle (mouth), MI-
NIBOS, Listings zum
Floppykurs, CAT-
Routine, Steinschlag.

CPC-Magazin 8-9/86

Sieben auf einen
Streich (Teil 9),
Blinkender Cursor
und Tastenclck,
Musikgraph,
RSXINFO,
Basic-Compiler,
vortex.Com,
Mini-Movie,
Neues Puzzlebild
(Hamster),
Jolly Jumper.

CPC-Magazin 7/86

Minigolf, Centibug,
3D-Processor,
Digitalisierer,
Sieben auf einen
Streich (Teil 8),
Neues Puzzlebild
(Puzface),
Bos.Dat.Bas (464 +
vortexspeicher-
erweiterung).

CPC-Magazin 10/86

Längenausdehnung,
Thermometer,
Examiner,
Sieben auf einen
Streich (Teil 10),
Quader malen,
Symbol-Definition,
Windows,
Disassembler,
Neues Puzzlebild
(Puzpsy), Fastroutine,
Utilities für die
vortex-Floppy,
Pyramide, HighTerm.

CPC-Magazin 11/86

Blumenspiel,
Sieben auf einen
Streich (Teil 11),
Schach-Archiv,
Mini-Texter,
Window Creator,
Neues Puzzlebild
(Madonna),
Funktionstasten für
den vortex-Monitor,
Catsuch, Forth-
Compiler, Tennis.

Schach-Archiv: das Programm für den Schachspieler

Damit können Sie Schachpartien analysieren, nachspielen und für Ihre Sammlung archivieren

464

664

6128

Mit diesem Programm kann man Schachpartien analysieren, nachspielen, archivieren und sich so eine eigene Partien-Sammlung aufbauen. Die Partien werden auf der Diskette unter "User 1" abgespeichert, damit man eine bessere Übersicht hat, falls noch andere, nicht zum Schachprogramm gehörende Files auf der Diskette sind. Wenn Sie das nicht wünschen, können Sie die User-Kommandos im Programm in !User,0 umändern. Das Programm wurde auf dem CPC 464 mit DDI-1 Floppy geschrieben, läuft aber auch auf dem CPC 664/6128 oder mit der vortex-Floppy.

Zur Bedienung des Programms

Diskette einlegen und RUN "schach" eingeben. Auf die Frage "Generator oder Hauptprogramm laden (G/H)" mit dem jeweiligen Buchstaben antworten. Wurde das Hauptprogramm gewählt, meldet sich dieses nach ein paar Sekunden mit der Frage "Partien laden (J/N)". Wenn Sie keine eigenen Partien laden wollen, geben Sie "n" ein. Sie können dann aus 3 fest im Programm gespeicherten Demos wählen. Falls Sie eine mit dem Generator eingegebene eigene Partie laden wollen, drücken Sie "j". Sie werden daraufhin gefragt, ob die richtige Diskette mit der zu ladenden Partie eingelegt ist. Wenn das der Fall ist, drücken Sie eine beliebige Taste. Danach fragt Sie das Programm nach dem Namen der Partie, die Sie laden wollen. Wenn Sie nur ENTER drücken, erscheint das Directory der Diskette. Haben Sie den richtigen Namen eingegeben, wird die Partie geladen.

Bei der Frage "Tastendruck oder automatisch" haben Sie die Wahl, nach jedem Zug eine Taste zu drücken oder die Züge automatisch mit einem Abstand von ca. 1,5 Sekunden ablaufen zu lassen. Danach werden das Schachbrett, die Zuganzeige, die Namen der Spieler und das Datum der Partie auf dem Bildschirm angezeigt. Rechts werden die Züge notiert. Am Ende einer Partie wartet das Programm auf einen Tastendruck und kehrt dann zur Auswahl zurück.

Haben Sie zu Beginn den Generator gewählt, wird er zur Eingabe der Partien geladen. Zuerst wird das Directory der Diskette gezeigt, damit Sie nicht versehentlich denselben Dateinamen zweimal verwenden. Dann müssen Sie den Namen eingeben, unter dem die Partie später gespeichert werden soll, danach den Namen der Spieler (zuerst Schwarz, dann Weiß) und das Jahr, in dem die Partie stattfand. (Verwenden Sie niemals Kommas! Der Name der Partie darf nicht länger als 8 Zeichen sein! Vergewissern Sie sich, daß die richtige Diskette im Laufwerk liegt!) Sind diese Eingaben gemacht, können Sie anschließend die einzel-

nen Züge eingeben. Bei normalen Zügen steht zuerst das Ausgangsfeld und dann das Bestimmungsfeld (z.B. e2e4). Auch wenn eine andere Figur geschlagen wird, dürfen Sie die Felder nur so eingeben (nur Kleinbuchstaben, keine Spaces). Danach fragt das Programm, ob die Eingabe richtig war. Wenn ja, müssen Sie "j" und ENTER eingeben, wenn nicht, "n" und ENTER. Der Zug kann dann noch einmal eingegeben werden.

Folgende Sondereingaben sind zu beachten:

Rochade: Bei der kurzen Rochade (egal, welche Seite) ist kr einzugeben, bei der langen lr.

Schach: Hinter den Zug müssen Sie einfach ein s anhängen (z.B. h1e1s).

En-passant (Schlagen im Vorbeigehen): Hinter den Zug kommt ein v (z.B. e5f6v).

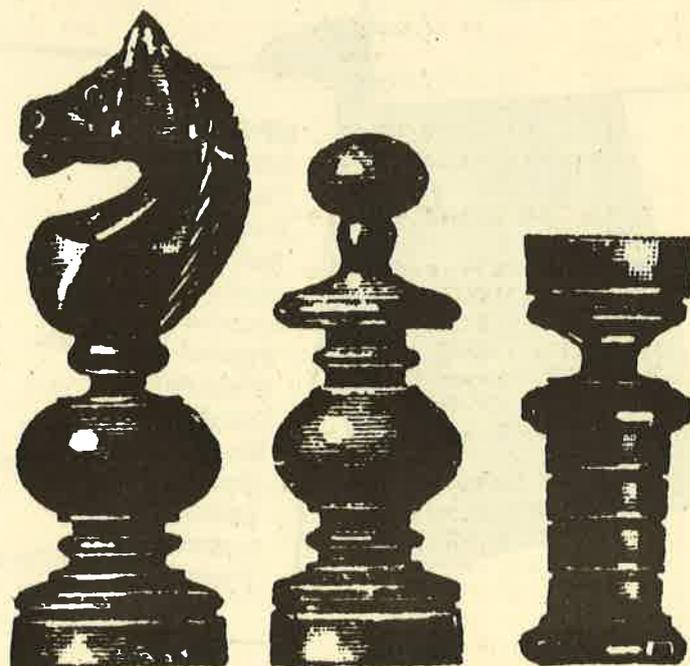
Aufgabe: Je nachdem, ob Schwarz oder Weiß aufgibt: sa oder wa.

Remis: An den letzten Zug müssen Sie hier ein r anhängen (z.B. a8a7r).

Matt: An den letzten Zug kommt ein m (z.B. a8a7m).

Bauernumwandlung: Es kann nur in eine Dame umgewandelt werden. An den Zug ist ein d anzuhängen (z.B. e7e8d).

Am Ende müssen Sie nur noch x eingeben. Die Partie liegt nun auf der Diskette vor und kann vom Hauptprogramm aus geladen werden. Wollen Sie anschließend den



Generator-Teil verlassen, müssen Sie "n" drücken, worauf der Computer zurückgesetzt wird. Für einen erneuten Start ist wieder RUN "schach" erforderlich.

Hinweise zum Abtippen

Zuerst müssen Sie Listing 1 sorgfältig eintippen und mit GOTO 210 abspeichern, danach Listing 2, das Sie mit GOTO 290 speichern. Anschließend folgt Listing 3, das Hauptprogramm. Wenn hier alles richtig ist, muß dieses mit GOTO 2090 abgespeichert werden.

Wolfgang Volmering
Oliver Otten

Starter

```

1 'SCHACH.GO
2 '(c) W.Volmering Aug.1986
3 '
100 INK 0,0:INK 1,18:INK 2,6:INK 3,26
110 BORDER 0:PAPER 0:MODE 0:PEN 6
120 INPUT "Generator oder
      Hauptprogramm laden
      (G/H)";g$
130 IF g$="G" OR g$="g" THEN g=1:CLS:GOT
0 160
140 IF g$="h" OR g$="H" THEN g=2:CLS:GOT
0 160
150 GOTO 100
160 PEN 2:LOCATE 2,3:PRINT CHR$(24);" Sc
hach - Archiv ";CHR$(24):PEN 1
170 PRINT:PRINT:PRINT:PRINT:PRINT
"      Ladevorgang -

      bitte warten"
180 PEN 3:LOCATE 5,18:PRINT CHR$(164);"W
OOL-Soft"
190 IF g=1 THEN RUN "schach.gen" ELSE RU
N "schach.prg"
200 END
210 SAVE"schach.go"

```

Editor

```

1 ' *****
2 ' **      Schach - Archiv      **
3 ' **      von                  **
4 ' **      Wolfgang Volmering  **
5 ' **      und                  **
6 ' **      Oliver Otten        **
7 ' **                          **
8 ' **      (C) WOOL-Soft       **
9 ' *****
10 '
100 BORDER 0:INK 0,0:INK 1,2:PAPER 0:MOD
E 1:PEN 1
110 PRINT"Partie-Eingabe Generator ";
CHR$(164);"WOOL-Soft"
120 PRINT STRING$(40,208)
130 :USER,1:CAT
140 INPUT "Name Partie";name$:IF LEN(nam
e$)>8 THEN PRINT CHR$(7):PRINT:PRINT:PRI
NT"Falsche Eingabe ! Bitte nochmal.":GOT
0 140
150 PRINT:PRINT:INPUT "Name Schwarz";a$
160 OPENOUT name$:PRINT#9,a$
170 PRINT:PRINT:INPUT "Name Weiss";a$
180 PRINT#9,a$
190 PRINT:PRINT:INPUT "Datum der Partie"
;a$
200 PRINT#9,a$

```

```

210 INPUT ";a$:INPUT "Eingabe richtig (
J/N)";r$:IF r$="j" OR r$="J" THEN PRINT#
9,a$:IF a$="x" THEN 230 ELSE 210
220 PRINT"Zug nochmal eingeben":GOTO 210
230 CLOSEOUT:USER,0
240 PRINT:PRINT:PRINT"Partie ist gespeic
hert !"
250 PRINT:PRINT:INPUT "Nochmal ";ja$
260 ja$=LOWER$(ja$)
270 IF ja$="j" THEN RUN
280 CALL 0
290 SAVE "schach.gen"

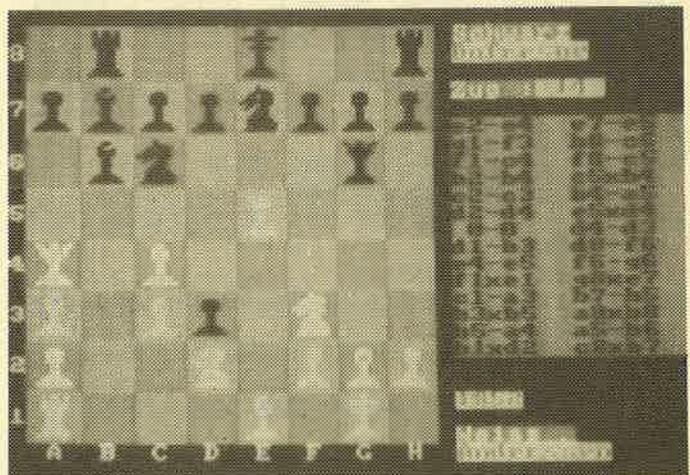
```

Hauptprogramm

```

1 '*****
2 '**      Schach - Archiv V1.0  **
3 '**      von                  **
4 '**      Wolfgang Volmering   **
5 '**      und                  **
6 '**      Oliver Otten        **
7 '**                          **
8 '**      (C) WOOL-Soft       **
9 '*****
10 '
100 INK 0,0:INK 1,13:INK 2,2:INK 3,24:BO
RDER 0:fa=1:wv=3
110 GOSUB 1780
120 PAPER 0:MODE 1:PEN 2
130 RESTORE 140:FOR i=1 TO 6:FOR z=1 TO
9:READ a:b$(i)=b$(i)+CHR$(a):NEXT z:NEXT
i
140 DATA 200,201,200,200,202,200,203,204
,205
150 DATA 206,207,208,209,211,212,203,214
,205
160 DATA 200,213,200,200,210,200,203,204
,205
170 DATA 216,217,218,219,220,221,222,214
,223
180 DATA 224,207,225,200,202,200,203,204
,205
190 DATA 226,199,215,209,202,212,203,204
,205
200 FOR i=1 TO 6:FOR j=0 TO 2:FOR k=1 TO
3
210 a$=MID$(b$(i),j*3+k,1):a$(i)=a$(i)+C
HR$(ASC(a$)):NEXT
220 a$(i)=a$(i)+CHR$(10)+CHR$(8)+CHR$(8)
+CHR$(8):NEXT:NEXT

```



```

230 DIM b(8,8)
240 FOR i=7 TO 1 STEP -2:FOR j=1 TO 7 ST
EP 2:b(i,j)=1:NEXT:NEXT
250 FOR i=8 TO 2 STEP -2:FOR j=2 TO 8 ST
EP 2:b(i,j)=1:NEXT:NEXT
260 DIM a(8,8)
270 GOSUB 1610
280 DIM p(20),p$(20),p1$(20),ja$(20):p=0
:d=0
290 p=p+1:READ p$(p),p1$(p),ja$(p):d=d+3
300 READ a$:d=d+1:IF a$="e" OR a$="x" TH
EN 320
310 GOTO 300
320 IF a$="x" THEN 340
330 p(p)=d:GOTO 290
340 RESTORE 1710:PAPER 0
350 CLS:LOCATE 5,1:PRINT"- Auswahl -
";CHR$(164);" WOOL-Soft"
360 LOCATE 1,3
370 PRINT:PRINT:PRINT"Partien laden (j/n
)"
380 PRINT:PRINT:INPUT " ";z$:z$=LOWER$(z$
):IF z$="n" OR z$="j" THEN GOTO 390: ELS
E PRINT CHR$(7):PRINT:PRINT"Falsch
e Eingabe ! Bitte nochmal.":GOTO 370
390 IF z$="n" THEN PRINT:PRINT:GOTO 430
400 PRINT:PRINT:PRINT"Richtige Diskette
eingelegt ?":CALL &BB06:PRINT:PRINT:PRIN
T"Name der Partie":PRINT:PRINT:INPUT " ";
name$:IF LEN(name$)>8 THEN PRINT CHR$(7)
:PRINT:PRINT:PRINT"Falsche Eingabe ! Bit
te nochmal.":GOTO 400
410 IF name$="" THEN :USER,1:CAT:USER,0
:GOTO 400
420 :USER,1:dir=1:OPENIN name$:CLS:GOTO
450
430 FOR i=1 TO p:PRINT i:TAB(4)p$(i)TAB(
18)p1$(i)TAB(35);ja$(i):NEXT
440 PRINT:PRINT:INPUT " ";w:IF w<1 OR w>p
THEN 440
450 PRINT:PRINT "Tastendruck oder automa
tisch (T/A)":PRINT:PRINT:INPUT " ";taste$
460 CLS:IF dir=1 THEN 490
470 CLS:IF w=1 THEN 490
480 FOR i=1 TO p(w-1):READ a$:NEXT
490 FOR j=1 TO 22 STEP 3:FOR i=2 TO 23 S
TEP 3
500 IF fa=2 THEN fa=1 ELSE fa=2
510 PAPER#1,fa:WINDOW #1,1,i+2,j,j+2:CLS
#1:NEXT:IF fa=1 THEN fa=2 ELSE fa=1
520 NEXT
530 PEN 3
540 e=9:FOR i=2 TO 23 STEP 3:e=e-1:e$=ST
R$(e):LOCATE 1,i:PRINT CHR$(8)+e$
550 NEXT
560 LOCATE 3,25:PRINT"A B C D E F
G H"
570 LOCATE 1,1
580 FOR y=7 TO 8:FOR x=1 TO 8
590 GOSUB 1520:NEXT:NEXT
600 FOR y=1 TO 2:FOR x=1 TO 8:GOSUB 1520
:NEXT:NEXT
610 PAPER 3:PEN 2
620 LOCATE 27,1 :PRINT CHR$(24);"Schwarz
";CHR$(24)
630 LOCATE 27,24:PRINT CHR$(24);"Weiss
";CHR$(24)
640 LOCATE 27,4 :PRINT CHR$(24);"Zug :";
CHR$(24);" 1"
650 WINDOW#2,27,40,6,19:PEN#2,0:PAPER#2,
2:CLS#2
660 IF dir=1 THEN INPUT#9,a$ ELSE READ a
$
670 a$=LEFT$(a$,13):LOCATE 27,2:PRINT a$
680 IF dir=1 THEN INPUT#9,a$ ELSE READ a
$
690 a$=LEFT$(a$,13):LOCATE 27,25:PRINT a
$
700 IF dir=1 THEN INPUT#9,a$ ELSE READ a
$
710 LOCATE 27,22:PRINT a$
720 zu=1:z1=-1:aw=0
730 z1=z1+1:IF z1=2 THEN z1=0:zu=zu+1:LO
CATE 32,4:PRINT zu
740 IF dir=1 THEN INPUT#9,a$ ELSE READ a
$
750 IF LEN(a$)<>4 THEN 940
760 b$=LEFT$(a$,2):GOSUB 1590:x=x2:y=y2
770 b2$=b$
780 b$=RIGHT$(a$,2):GOSUB 1590
790 o$="-":IF a(x2,y2)<>0 THEN o$="x"
800 IF sa=1 THEN b$=b$+"+":sa=0
810 IF sa=2 THEN b$=b$+"ep":sa=0
820 a$=b2$+o$+b$
830 IF z1=0 THEN PRINT#2,a$;: ELSE PRINT
#2,TAB(8);a$;
840 FOR i=1 TO 4:GOSUB 1520:GOSUB 1490:N
EXT
850 a1=a(x,y):a(x,y)=0:a(x2,y2)=a1
860 x=x2:y=y2:FOR i=1 TO 4:GOSUB 1490:GO
SUB 1520:NEXT
870 IF as=1 THEN as=0:RETURN
880 IF aw=4 THEN GOSUB 1290:aw=1
890 IF aw=3 THEN aw=0:RETURN
900 IF aw=2 THEN 1670:
910 GOSUB 1600
920 IF aw=1 THEN aw=0
930 GOTO 730
940 IF a$="e" THEN 1670
950 IF a$="sa" THEN 1110
960 IF a$="wa" THEN 1130
970 IF a$="kr" THEN 1170
980 IF a$="lr" THEN 1220
990 b$=RIGHT$(a$,1):a$=LEFT$(a$,4)
1000 IF b$="s" THEN aw=1:sa=1:GOTO 760
1010 IF b$="m" THEN aw=2:as=1:GOSUB 760:
PRINT#2:PRINT#2:PRINT#2," Matt !":GOTO 8
80
1020 IF b$="r" THEN aw=2:as=1:GOSUB 760:
PRINT#2:PRINT#2:PRINT#2," Remis !":GOTO
880
1030 IF b$="p" THEN aw=2:as=1:GOSUB 760:
PRINT#2:PRINT#2:PRINT#2," Patt !":GOTO 8
80
1040 IF b$="w" THEN aw=2:as=1:GOSUB 760
PRINT#2:PRINT#2:PRINT#2,"Dritte Mal"
1050 IF b$="w" THEN PRINT#2,"Gleiche":PR
INT#2,"Stellung !":GOTO 880
1060 IF b$="z" THEN aw=2:as=1:PRINT#2:PR
INT#2:PRINT#2," 50 Zuege"
1070 IF b$="z" THEN PRINT#2," ohne":PR
INT#2," Bauer !":GOTO 880
1080 IF b$="v" THEN aw=4:sa=2:GOTO 760
1090 IF b$="d" OR b$="t" OR b$="h" OR b$
="l" THEN f$=b$:GOTO 1320
1100 CLS:PRINT"Datafehler !!!":STOP
1110 a$="Schwarz"
1120 GOTO 1140
1130 a$="Weiss"

```

```

1140 PRINT#2,a$;
1150 PRINT #2,"gibt auf !"
1160 GOTO 1670
1170 y=1:t=0:IF z1=1 THEN y=8:t=6
1180 a(5,y)=0:a(8,y)=0:a(6,y)=8-t:a(7,y)
=12-t:x=5:GOSUB 1490:x=8:GOSUB 1490
1190 x=6:GOSUB 1520:x=7:GOSUB 1520
1200 IF z1=0 THEN PRINT#2,"0-0";GOTO 1
270
1210 PRINT#2,TAB(8)"0-0":GOTO 1270
1220 y=1:t=0:IF z1=1 THEN y=8:t=6
1230 a(1,y)=0:a(5,y)=0:a(3,y)=12-t:a(4,y)
)=8-t
1240 x=1:GOSUB 1490:x=5:GOSUB 1490:x=3:G
OSUB 1520:x=4:GOSUB 1520
1250 IF z1=0 THEN PRINT#2,"0-0-0";GOTO
1270
1260 PRINT#2,TAB(8)"0-0-0"
1270 GOSUB 1600
1280 GOTO 730
1290 IF a1=7 THEN y=y-1
1300 IF a1=1 THEN y=y+1
1310 a(x,y)=0:GOSUB 1490:RETURN
1320 aw=3:GOSUB 760
1330 IF (y2=8) AND (f$="d") THEN a(x2,8)
=11
1340 IF (y2=1) AND (f$="d") THEN a(x2,1)
=5
1350 IF (y2=8) AND (f$="t") THEN a(x2,8)
=8
1360 IF (y2=1) AND (f$="t") THEN a(x2,1)
=2
1370 IF (y2=8) AND (f$="h") THEN a(x2,8)
=10
1380 IF (y2=1) AND (f$="h") THEN a(x2,1)
=4
1390 IF (y2=8) AND (f$="l") THEN a(x2,8)
=9
1400 IF (y2=1) AND (f$="l") THEN a(x2,1)
=3
1410 x=x2:y=y2:GOSUB 1520
1420 IF f$="d" THEN PRINT#2,"Dame-";
1430 IF f$="t" THEN PRINT#2,"Turn-";
1440 IF f$="h" THEN PRINT#2,"Springer-";
1450 IF f$="l" THEN PRINT#2,"Laeufer-";
1460 PRINT#2,"Tausch"
1470 GOSUB 1600:GOTO 730
1480 GOTO 730
1490 IF b(x,y)=1 THEN 1510
1500 wv=2:GOSUB 1650:PAPER#1,2:CLS#1:wv=
3:RETURN
1510 wv=2:GOSUB 1650:PAPER#1,1:CLS#1:wv=
3:RETURN
1520 f=a(x,y)
1530 GOSUB 1650
1540 PEN#1,3:IF f>6 AND b(x,y)=0 THEN PA
PER#1,2:PRINT#1,a$(f-6):RETURN
1550 IF f>6 AND b(x,y)=1 THEN PAPER#1,1:
PRINT#1,a$(f-6):RETURN
1560 PEN#1,0:IF b(x,y)=1 THEN PAPER#1,1:
PRINT#1,a$(f):RETURN
1570 PAPER#1,2:PRINT#1,a$(f):RETURN

1580 END
1590 b1$=LEFT$(b$,1):x2=ASC(b1$)-96:y2=V
AL(RIGHT$(b$,1)):RETURN
1600 IF taste$="T" OR taste$="t" THEN CA
LL &BB06 ELSE FOR i=1 TO 1500:NEXT i:RET
URN
1610 a(1,1)=8:a(2,1)=10:a(3,1)=9:a(4,1)=
11:a(5,1)=12:a(6,1)=9:a(7,1)=10:a(8,1)=8
1620 FOR i=1 TO 8:a(i,8)=a(i,1)-6:NEXT
1630 FOR x=1 TO 8:a(x,2)=7:a(x,7)=1:NEXT
1640 RETURN
1650 x1=(x-1)*3+2:y1=(8-y)*3+1:WINDOW#1,
x1,x1+wv,y1,y1+wv:RETURN
1660 REM IF y=8 THEN LOCATE ((x-1)*3+2),
1:RETURN
1670 CALL &BB06
1680 FOR i=1 TO 8:FOR j=1 TO 8:a(i,j)=0:
NEXT:NEXT
1690 dir=0:CLOSEIN:USER,0
1700 GOSUB 1610:GOTO 340
1710 DATA Herzog/Graf,Morphy,1858
1720 DATA e2e4,e7e5,g1f3,d7d6,d2d4,c8g4,
d4e5,g4f3,d1f3,d6e5,f1c4,g8f6,f3b3,d8e7,
b1c3,c7c6,c1g5
1730 DATA b7b5,c3b5,c6b5,c4b5s,b8d7,1r,a
8d8,d1d7,d8d7,h1d1,e7e6,b5d7s,f6d7,b3b8s
,d7b8,d1d8m,e
1740 DATA Dufresne,Anderssen,1852
1750 DATA e2e4,e7e5,g1f3,b8c6,f1c4,f8c5,
b2b4,c5b4,c2c3,b4a5,d2d4,e5d4,kr,d4d3,d1
b3,d8f6,e4e5,f6g6,f1e1,g8e7,c1a3,b7b5,b3
b5,a8b8,b5a4,a5b6,b1d2,c8b7,d2e4,g6f5,c4
d3,f5h5,e4f6s,g7f6,e5f6,h8g8,a1d1,h5f3,e
1e7s,c6e7,a4d7s,e8d7,d3f5s,d7e8,f5d7s,e8
d8,a3e7m,e
1760 DATA Kieseritzky,Anderssen,1851
1770 DATA e2e4,e7e5,f2f4,e5f4,f1c4,d8h4s
,e1f1,b7b5,c4b5,g8f6,g1f3,h4h6,d2d3,f6h5
,f3h4,h6g5,h4f5,c7c6,g2g4,h5f6,h1g1,c6b5
,h2h4,g5g6,h4h5,g6g5,d1f3,f6g8,c1f4,g5f6
,b1c3,f8c5,c3d5,f6b2,f4d6,c5g1,f1e2,b8a6
,f5g7s,e8d8,f3f6s,g8f6,d6e7m,x
1780 SYMBOL AFTER 198
1790 RESTORE 1800:FOR i=199 TO 226:READ
a1,a2,a3,a4,a5,a6,a7,a8:SYMBOL i,a1,a2,a
3,a4,a5,a6,a7,a8:NEXT
1800 DATA 0,60,60,60,255,255,60,60
1810 DATA 0,0,0,0,0,0,0,0
1820 DATA 0,0,0,0,60,255,255,255
1830 DATA 255,255,60,60,60,60,60,60
1840 DATA 0,0,3,3,15,10,0,0
1850 DATA 60,255,255,255,255,170,0,0
1860 DATA 0,0,192,192,240,160,0,0
1870 DATA 0,0,3,3,3,3,3,3
1880 DATA 0,0,60,60,60,255,255,255
1890 DATA 0,0,192,192,192,192,192,192
1900 DATA 3,0,0,0,0,0,0,0
1910 DATA 255,60,60,60,60,60,60,60
1920 DATA 255,255,255,255,255,255,25,25
5
1930 DATA 192,0,0,0,0,0,0,0
1940 DATA 0,0,60,60,240,195,255,255
1950 DATA 255,255,255,255,255,170,0,0
1960 DATA 0,0,0,0,240,240,0,0
1970 DATA 0,0,0,0,0,3,3,15
1980 DATA 0,48,48,63,255,207,207,255
1990 DATA 0,0,0,0,192,192,192,192
2000 DATA 63,51,7,12,0,0,0,3
2010 DATA 255,207,15,63,63,255,255,255
2020 DATA 192,192,0,0,0,0,0,192
2030 DATA 3,3,3,3,15,10,0,0
2040 DATA 192,240,240,240,240,160,0,0
2050 DATA 0,0,12,15,3,3,3,3
2060 DATA 0,0,48,240,192,192,192,192
2070 DATA 0,0,0,0,15,15,0,0
2080 RETURN
2090 SAVE "schach.prg"

```

Tip des Monats

Mini-Texter

464

664

6128

Zuerst müssen Sie das Maschinenspracheprogramm eingeben und starten. Wenn alle DATAs fehlerfrei sind, wird die BIN-Datei auf Diskette abgespeichert (Cassette ist ungeeignet). Danach muß noch der Basic-Teil eingegeben und unter dem Namen "MINI" abgespeichert werden. Gestartet wird mit RUN "MINI". Anschließend ist auch der Drucker einzuschalten.



Die Bedienung des Programms erfolgt mittels Menüsteuerung durch den Cursor. Dazu muß man nur den Menüpunkten folgen. Zur Wahl des Farbbereichs dient ebenfalls der Cursor: nach oben bedeutet heller, nach unten dunkler. Beendet wird jeder Modus mit Return. Ein Y-Z-Tausch ist möglich. Sollten die Druckerodes nicht übereinstimmen, reicht es aus, sie im Basic zu ändern. Beim Abspeichern oder beim Laden wird bei Disketten-Dateien automatisch ".TEX" angehängt. DIR zeigt alle diese TEX-Dateien.

Jetzt zum Editor. Mit der Funktion "Umbruch" wird beim Überschreiben des rechten Randes der letzte Wortanfang in die neue Zeile gesetzt, außer das Wort hört mit !"#\$\$%&'()*+, -./ auf.

Blocksatz richtet die Wörter im Arbeitsausschnitt so aus, daß die Leerzeichen am Ende der Zeile auf die Wortzwischenräume verteilt werden. Dieser Teil ist in Basic geschrieben und etwas langsam. Der Blocksatz ist besonders für 2-Spalten-Texte geeignet.

Mit der Funktion "Text löschen" lassen sich der Text im Hintergrund oder der Textspeicher löschen. Wird beides nicht gewünscht, kommt man mit "zurück" wieder ins Editormenü. Der "Mini-Texter" unterstützt 4 Formate:

1. Volle Breite: Es stehen 80 Zeichen pro Zeile zur Verfügung.
2. Ränder: Begrenzter Arbeitsbereich, ideal für Briefe.
3. 2-Spalten-Texte: Zweispartige Texte sind für Schüler- oder Clubzeitungen ideal.
4. Briefformat: Setzt gleich den richtigen Briefkopf. Ideal für kleine Geschäfte und private Anwendungen. Im Programm müssen Sie dazu eigene Daten einsetzen. Da auch dieser Teil in Basic geschrieben ist, läßt er sich gut

Die Funktionen des Editors

Cursorsteuerung (**)
 Return setzt Cursor in die neue Zeile (*)
 DEL löscht links vom Cursor (*)
 CLR löscht rechts vom Cursor (*)
 kein Copycursor
 kein Einfügemodus (auch nicht wählbar)
 TAB verschiebt Text (*)
 COPY verläßt den Editor
 Sonderzeichen auf den Tasten @[] \\
 Funktionstasten lassen sich belegen
 CAPS LOCK Feststellshift
 CONTROL+CAPS LOCK
 Scrollen durch Herausfahren aus dem Bildschirmbereich
 Shift
 Löschen mit Potenzpfeil und Anzahl der zu löschenden Zeilen (**)
 Einfügen (Kopieren) mit Pfundzeichen und Anzahl der einzufügenden (kopierten) Zeilen (**)
 kein ständiges Piepsen bei Unzulässigkeiten

CONTROL-Tastenbelegung:

Q – Überschreibt Zeile mit Leerzeichen und löscht deren Inhalt (*).
 W – Schaltet Umbruchmodus ein.
 E – Schaltet Umbruchmodus aus.
 R – Gibt in der ersten Bildschirmzeile die x- und y-Position des Cursors sowie den linken und den rechten Rand aus.
 T – Setzt Cursor auf Position x 1 / y1 (**).
 A – Unterstreichen ein (gibt auch ein Leerzeichen aus).
 S – Unterstreichen aus (gibt auch ein Leerzeichen aus).
 Z – Setzt linke Randbegrenzung auf momentane Cursorstelle.
 X – Setzt linke Randbegrenzung auf 1.

(*) Befehl nur in der momentanen Zeile gültig, die durch die Ränder begrenzt ist.

(**) Befehl nur im momentanen Arbeitsbereich gültig (zwischen den Begrenzungen).

mit eigenen Adreßverwaltungen koppeln. Wird bei Ort, Name und Straße keine Eingabe gemacht, werden dafür die Standardwerte eingesetzt.

5. Eigene Formate: Hier lassen sich eigene Formate (3-Spalten-Texte, schmales Papier etc.) erfinden.

Die Länge der Texte ist auf max. 57 Zeilen (ca. DIN A4) begrenzt. Zum Einrücken von Texten läßt sich der linke Rand mit Control-Z auf die Cursorposition verschieben. Mit Control-X ist der linke Rand wieder auf 1.

Peter Klar

MC-Generator

```

1 ' MINITEXT.LDR
2 ' (c) Aug.1986, Peter Klar
3 '
4 ' Erzeugt MINITEXT.MC
5 ' fuer MINITEXT.BAS
6 '
100 MEMORY &7999
110 MODE 1:INK 2,20,26
120 PEN 2:PRINT TAB(15);"Mini-Texter"
130 PRINT:PRINT" (c) 1986 by Peter Klar
":PRINT
140 PEN 1:PRINT" Bitte warten, ich lese
die Datas":PRINT
150 adr=&8000
160 FOR a=1000 TO 1720 STEP 10
170 PRINT a:
180 s=0:FOR b=0 TO 15
190 READ a$:by=VAL("&"+a$):POKE adr,by:a
dr=adr+1:s=s+by
200 NEXT:READ a$:IF VAL("&"+a$)<>s THEN
PRINT"Error":END ELSE PRINT"OK";CHR$(13)
;
210 NEXT
220 PRINT:PRINT"Bitte Diskette einlegen"
230 IF INKEY$="" THEN 230
240 SAVE"minitext.mc",b,&8000,&48F,0
250 '
1000 DATA C3,45,84,C3,64,84,C3,61,84,3E,
FF,32,0D,86,3A,00,071B
1010 DATA 86,CD,33,84,11,40,86,19,22,09,
86,3A,01,86,3D,CD,0576
1020 DATA 33,84,EB,2A,09,86,19,22,0B,86,
EB,2A,02,86,26,00,04EA
1030 DATA 2D,19,22,03,86,3A,0D,86,FE,FF,
20,21,3E,02,CD,0E,0517
1040 DATA BC,2A,09,86,06,08,C5,06,FA,7E,
FE,20,20,02,3E,09,054D
1050 DATA CD,5A,BB,23,10,F3,C1,10,ED,97,
32,0D,86,2A,01,86,06D3
1060 DATA CD,75,BB,3E,18,CD,5A,BB,2A,03,
86,7E,CD,5A,BB,3E,0786
1070 DATA 18,CD,5A,BB,CD,06,BB,CD,1B,BB,
32,05,86,CD,82,80,07B7
1080 DATA 18,8C,FE,7F,30,08,FE,20,38,04,
FE,5E,20,60,21,AE,065E
1090 DATA 80,56,3A,05,86,BA,20,0D,23,7E,
32,A3,80,23,7E,32,054B
1100 DATA A4,80,C3,00,00,7A,FE,00,C8,23,
23,23,18,E3,10,01,059C
1110 DATA 81,7F,2F,81,E0,61,81,09,63,81,
0D,F9,81,F3,8E,81,07E8
1120 DATA F2,60,82,F0,7B,82,F1,04,82,5E,
FE,82,A3,13,83,11,0860
1130 DATA 42,83,17,66,83,05,6D,83,12,85,
83,14,74,83,01,1B,04FB
1140 DATA 84,13,1F,84,FC,09,80,1A,27,84,

```

```

18,2C,84,00,2A,01,0477
1150 DATA 86,CD,75,BB,3A,05,86,CD,5A,BB,
2A,03,86,77,C3,91,07A8
1160 DATA 81,3A,08,86,57,3A,02,86,BA,D0,
2A,03,86,EB,2A,03,05B7
1170 DATA 86,23,3A,02,86,4F,3A,08,86,91,
4F,06,00,ED,B0,2A,052F
1180 DATA 0B,86,3A,08,86,3D,5F,16,00,19,
36,20,C3,39,82,3A,0432
1190 DATA 02,86,57,3A,07,86,BA,D0,3A,02,
86,3D,32,02,86,5F,0548
1200 DATA 3A,08,86,93,4F,06,00,2A,03,86,
EB,1B,2A,03,86,ED,0509
1210 DATA B0,2A,0B,86,3A,08,86,3D,5F,16,
00,19,36,20,C3,39,0450
1220 DATA 82,E1,C9,3A,08,86,57,3A,02,86,
3D,BA,D0,2A,0B,86,068F
1230 DATA 3A,08,86,3D,5F,16,00,19,54,5D,
2B,ED,4B,02,86,3A,0469
1240 DATA 08,86,91,4F,06,00,ED,B8,23,36,
20,C3,39,82,CD,53,0630
1250 DATA 84,3A,02,86,3C,32,02,86,57,3A,
08,86,BA,D0,3A,06,0525
1260 DATA 86,FE,07,20,57,2A,03,86,7E,FE,
30,38,4F,06,01,2B,051A
1270 DATA 7E,FE,20,28,03,04,18,F7,3A,07,
86,80,4F,3A,08,86,0538
1280 DATA B9,38,39,3A,07,86,C6,4F,23,E5,
2A,0B,86,5F,16,00,053E
1290 DATA 19,EB,E1,3A,07,86,80,32,02,86,
7E,12,36,20,13,23,0502
1300 DATA 10,F8,CD,39,82,18,20,3A,06,86,
FE,07,C0,11,50,00,05B4
1310 DATA 2A,0B,86,19,22,0B,86,18,40,CD,
53,84,3A,07,86,32,047C
1320 DATA 02,86,18,03,CD,53,84,3A,01,86,
3C,32,01,86,FE,1A,0515
1330 DATA 38,D5,CD,5B,83,3E,19,32,01,86,
3A,00,86,FE,20,D0,0676
1340 DATA 3C,32,00,86,97,06,01,CD,4D,BC,
21,19,01,CD,75,BB,05A0
1350 DATA 2A,0B,86,11,50,00,19,18,71,2A,
01,86,3A,07,86,67,039D
1360 DATA CD,75,BB,3A,07,86,57,3A,08,86,
92,3C,47,2A,0B,86,05B3
1370 DATA 3A,07,86,5F,16,00,1B,19,7E,CD,
5A,BB,23,10,F9,C9,05C5
1380 DATA CD,53,84,3A,07,86,57,3A,02,86,
14,BA,38,05,3D,32,04FE
1390 DATA 02,86,C9,3A,08,86,32,02,86,18,
03,CD,53,84,3A,01,04CD
1400 DATA 86,FE,02,38,05,3D,32,01,86,C9,
3A,00,86,FE,00,C8,0608
1410 DATA 3D,32,00,86,97,06,00,CD,4D,BC,
21,01,01,CD,75,BB,0588
1420 DATA 2A,0B,86,11,50,00,37,3F,ED,52,
06,50,7E,FE,20,20,04E3
1430 DATA 02,3E,09,CD,5A,BB,23,10,F3,E1,
C3,0E,80,3A,07,86,064A
1440 DATA 3D,5F,16,00,19,22,0F,86,2A,00,
86,7C,85,47,3E,3A,03F2
1450 DATA 90,47,C5,2A,0F,86,54,5D,01,50,
00,3A,0E,86,FE,00,0529
1460 DATA 28,06,37,3F,ED,42,18,01,09,22,
0F,86,3A,07,86,3D,03B0
1470 DATA 47,3A,08,86,90,4F,06,00,ED,B0,
C1,10,D5,C9,CD,2C,06F9
1480 DATA 83,C5,2A,0B,86,97,32,0E,86,CD,
BD,82,C1,10,F2,E1,0810

```

```

1490 DATA C3,09,80,CD,2C,83,C5,21,10,98,
3E,01,32,0E,86,CD,0628
1500 DATA BD,82,C1,10,F1,CD,5B,83,E1,C3,
09,80,CD,15,84,CD,090C
1510 DATA 09,BB,FE,30,38,F9,FE,3A,30,F5,
D6,30,20,02,E1,C9,0852
1520 DATA 47,C9,2A,0B,86,3A,07,86,5F,16,
00,19,3A,08,86,93,047B
1530 DATA 3C,2B,47,36,20,23,10,FB,C3,39,
82,21,10,98,06,50,04CF
1540 DATA 36,20,23,10,FB,C9,CD,15,84,32,
06,86,C9,97,32,06,0609
1550 DATA 86,C3,15,84,97,32,00,86,3C,32,
01,86,3A,07,86,32,051F
1560 DATA 02,86,C3,09,80,21,01,01,CD,75,
BB,CD,15,84,3E,78,0810
1570 DATA CD,5A,BB,2A,02,86,26,00,CD,01,
84,3E,79,CD,5A,BB,06A5
1580 DATA 2A,01,86,3A,00,86,85,6F,26,00,
CD,01,84,3E,6C,CD,0554
1590 DATA 5A,BB,2A,07,86,26,00,CD,01,84,
3E,72,CD,5A,BB,2A,0600
1600 DATA 08,86,26,00,CD,01,84,21,81,84,
3A,06,86,FE,07,20,0517
1610 DATA 03,21,86,84,06,05,7E,CD,5A,BB,
23,10,F9,CD,09,BB,0656
1620 DATA FE,00,28,F9,CD,15,84,21,01,01,
CD,75,BB,06,46,3E,062F
1630 DATA 20,CD,5A,BB,10,FB,3E,0D,CD,5A,
BB,2A,09,86,C3,AA,0760
1640 DATA 82,3E,20,CD,5A,BB,DF,8B,84,3E,
20,CD,5A,BB,CD,5A,0817
1650 DATA BB,CD,5A,BB,C9,3E,07,CD,5A,BB,
C9,3E,D7,18,02,3E,07C3
1660 DATA D5,32,05,86,C3,EE,80,3A,02,86,
18,02,3E,01,32,07,0517
1670 DATA 86,18,E2,D5,C5,5F,6F,26,00,16,
00,29,29,19,29,29,04E1
1680 DATA 29,29,C1,D1,C9,21,40,86,36,20,
11,41,86,01,C0,12,0595
1690 DATA ED,B0,C9,2A,01,86,CD,75,BB,2A,
03,86,7E,CD,5A,BB,0827
1700 DATA C9,97,18,02,3E,FF,DD,6E,00,DD,
66,01,4E,06,00,23,05BD
1710 DATA 5E,23,56,DD,6E,02,DD,66,03,FE,
00,28,01,EB,ED,B0,0719
1720 DATA C9,4F,46,46,20,20,4F,4E,20,20,
20,44,EF,FC,00,00,0510
1730 'Zeile 1720 fuer CPC 464:
1740 'DATA C9,4F,46,46,20,20,4F,4E,20,20
,20,79,EE,FC,00,00,0544
1750 'Zeile 1720 fuer CPC 664:
1760 'DATA C9,4F,46,46,20,20,4F,4E,20,20
,20,49,EF,FC,00,00,0515

```

Hauptprogramm

```

1 ' MINITEXT.BAS
2 ' (c) Aug. 1986, Peter Klar
3 '
1000 WIDTH 255
1010 SYMBOL AFTER 63
1020 MEMORY &7FFF
1030 KEY DEF 68,1
1040 KEY DEF 26,1,&40,&7E
1050 KEY DEF 17,1,&7B,&5B
1060 KEY DEF 19,1,&7D,&5D
1070 KEY DEF 22,1,&7C,&5C

```

```

1080 SYMBOL 91,&DB,&3C,&66,&66,&7E,&66,&
66
1090 SYMBOL 92,&66,&3C,&66,&66,&66,&66,&
3C
1100 SYMBOL 93,&66,&42,&66,&66,&66,&66,&
3C
1110 SYMBOL 123,&CC,0,&78,&C,&7C,&CC,&7E
1120 SYMBOL 124,&66,0,&3C,&66,&66,&66,&3
C
1130 SYMBOL 125,&66,0,&66,&66,&66,&66,&3
E
1140 SYMBOL 126,&3C,&66,&66,&6C,&66,&66,
&6C
1150 SYMBOL 64,&1C,&20,&1C,&22,&22,&1C,2
,&1C
1160 MODE 2
1170 REM Vorbelegung:
1180 bor=10:schrift=25:papier=1
1190 begin=&8000:schnitt=&8600:bspei=&86
40
1200 LOAD"minitext.mc",begin
1210 IF PEEK(&BB4F)=&78 THEN POKE &848B,
&79:POKE &848C,&EE : '--- 464 ---
1220 IF PEEK(&BB4F)=&70 THEN POKE &848B,
&49:POKE &848C,&EF : '--- 664 ---
1230 IF PEEK(&BB4F)=&74 THEN POKE &848B,
&44:POKE &848C,&EF : '--- 6128 ---
1240 CALL begin
1250 POKE schnitt,0:POKE schnitt+1,1:POK
E schnitt+2,1:POKE schnitt+7,1:POKE schn
itt+8,80:POKE schnitt+6,0
1260 ON ERROR GOTO 1290
1270 a$="":GOSUB 1380
1280 GOTO 1460
1290 ON ERROR GOTO 1290
1300 CLS:PRINT:PRINT CHR$(24)" Fehler";E
RR:"in Zeile";ERL
1310 PRINT CHR$(24):PRINT" Abbrechen"
1320 PRINT" Fortfahren"
1330 PRINT" Wiederholen"
1340 a$=LOWER$(INKEY$)
1350 IF a$="a"THEN RESUME 1460
1360 IF a$="f"THEN RESUME NEXT
1370 IF a$="w"THEN RESUME ELSE 1340
1380 WINDOW #0,23,53,6,19:WINDOW #1,22,5
4,5,20:WINDOW SWAP 0,1
1390 FOR a=2 TO 15:LOCATE 1,a:PRINT CHR$
(143):LOCATE 33,a:PRINT CHR$(143);:NEXT
1400 LOCATE 1,1:PRINT CHR$(24);"
Mini-Texter " :CHR$(24)
1410 LOCATE 1,16:PRINT CHR$(24);"
(c) by Peter Klar " :CHR$(24);:WI
NDOW SWAP 0,1
1420 CLS
1430 LOCATE 15-LEN(a$)/2,3:PRINT a$
1440 LOCATE 15-LEN(a$)/2,4:PRINT STRING$
(LEN(a$),"=")
1450 RETURN
1460 BORDER bor:INK 1,schrift:INK 0,papi
er:a$="Hauptmenue":GOSUB 1420
1470 an=5:a$(1)="Editor":a$(2)="Diskette
":a$(3)="Drucken"
1480 a$(4)="Farbe":a$(5)=CHR$(10)+"Ende"
1490 GOSUB 3150
1500 ON a GOTO 1510,1810,1990,2230:BORDE
R 0:INK 0,0:INK 1,24:MODE 1:END
1510 a$="Editor":GOSUB 1420
1520 an=6:a$(1)="Editieren":a$(2)="Forma
t":a$(3)="Text loeschen"

```

```

1530 a$(4)="Blocksatz":a$(5)="Umbruch":a
$(6)=CHR$(10)+"Hauptmenue"
1540 GOSUB 3150
1550 ON a GOTO 1560,1580,1730,2610,1780,
1460
1560 CALL begin+9
1570 a$="Editor":GOSUB 1380:GOTO 1520
1580 a$="Textbreite":GOSUB 1420
1590 PRINT " Links: ";PEEK(schnitt+7);TAB(
15);"Rechts";PEEK(schnitt+8)
1600 an=6:a$(1)="Volle Breite":a$(2)="Ra
ender":a$(3)=" 1.Spalte"
1610 a$(4)=" 2.Spalte":a$(5)="Freie Wahl
":a$(6)=CHR$(10)+"Brief"
1620 GOSUB 3150
1630 ON a GOSUB 1640,1650,1660,1670,1680
,2810:POKE schnitt+2,PEEK(schnitt+7):a$=
"Editor":GOSUB 1380:GOTO 1520
1640 POKE schnitt+7,1:POKE schnitt+8,80:
RETURN
1650 POKE schnitt+7,4:POKE schnitt+8,70:
RETURN
1660 POKE schnitt+7,1:POKE schnitt+8,38:
RETURN
1670 POKE schnitt+7,42:POKE schnitt+8,80
:RETURN
1680 a$="Linker Rand":GOSUB 1420
1690 LOCATE 6,7:INPUT "",1:1=INT(ABS(1)):
IF 1<1 OR 1>79 THEN 1680
1700 a$="Rechter Rand":GOSUB 1420
1710 LOCATE 6,7:INPUT "",r:r=INT(ABS(r)):
IF r<2 OR r>80 THEN 1700
1720 IF(r-1)<4 THEN 1680 ELSE POKE schni
tt+7,1:POKE schnitt+8,r:RETURN
1730 a$="Text loeschen":GOSUB 1420
1740 an=3:a$(1)="Text loeschen":a$(2)="B
ild ohne Text":a$(3)=CHR$(10)+"Zurueck"
1750 GOSUB 3140:ON a GOTO 1760,1770,1510
1760 CALL begin:GOTO 1510
1770 MODE 2:a$="Editor":GOSUB 1380:GOTO
1520
1780 a$="Umbruch":GOSUB 1420:LOCATE 28,1
:IF PEEK(schnitt+6)=7 THEN PRINT "An" ELS
E PRINT "Aus"
1790 an=2:a$(1)="Aus":a$(2)="An"
1800 GOSUB 3150:POKE schnitt+6,a*7-7:GOT
O 1510
1810 a$="Diskette":GOSUB 1420
1820 an=4:a$(1)="Abspeichern":a$(2)="Lad
en":a$(3)="Directory":a$(4)=CHR$(10)+"Ha
uptmenue"
1830 GOSUB 3150
1840 ON a GOTO 1850,1900,1950,1460
1850 a$=a$(1):GOSUB 1420
1860 PRINT:INPUT " Name: ",a$
1870 IF a$="" THEN 1810
1880 SAVE a$+".tex",b,schnitt,4870,schni
tt
1890 GOTO 1810
1900 a$=a$(2):GOSUB 1420
1910 PRINT:INPUT " Name: ",a$
1920 IF a$="" THEN 1810
1930 LOAD a$+".tex"
1940 GOTO 1810
1950 MODE 2:a$="*.tex":!DIR,@a$
1960 PRINT:PRINT TAB(20);"<Eine Taste dr
uecken>"
1970 IF INKEY$="" THEN 1970

```

```

1980 a$="Diskette":GOSUB 1380:GOTO 1820
1990 a$="Drucken":GOSUB 1420:IF(INP(&F50
0)AND 127)<>26 THEN PRINT:PRINT " Drucke
r anschliessen !":FOR i=0 TO 999:NEXT:GO
TO 1460
2000 an=4:a$(1)="Drucken":a$(2)="NLQ":a$
(3)="Normal":a$(4)=CHR$(10)+"Hauptmenue"
2010 GOSUB 3150
2020 ON a GOTO 2030,2210,2220,1460
2030 a$="Drucken":GOSUB 1420
2040 PRINT:INPUT "Menge: ",a:a=MIN(INT(AB
S(a)),50):IF a=0 THEN a=1
2050 PRINT#8,CHR$(27)"@"; : 'RESE
T ->Riteman F+
2060 PRINT#8,CHR$(27)"R"CHR$(2); : '2. Z
eichensatz ->Riteman F+
2070 'PRINT#8,CHR$(27)chr$(54); : '2. Z
eichensatz ->Schneider NLQ
2080 PRINT#8,CHR$(13); : 'Clea
r width counter
2090 IF nlq=1 THEN PRINT#8,CHR$(27);"x"
;CHR$(1);:'NLQ ->Riteman F+
2100 'if nlq=1 then PRINT#8,CHR$(27);"I"
;CHR$(3);:'NLQ ->Schneider NLQ
2110 IF nlq=0 THEN PRINT#8,CHR$(27);"x"
;CHR$(0);:'PICA ->Riteman F+
2120 'if nlq=0 then PRINT#8,CHR$(27);"I"
;CHR$(1);:'PICA ->Schneider NLQ
2130 FOR an=1 TO a:FOR i=0 TO 56
2140 a$=SPACE$(80):CALL begin+6,bspei+80
*!,@a$
2150 a=INSTR(a$,CHR$(215)):IF a<>0 THEN
a$=LEFT$(a$,a-1)+CHR$(32)+CHR$(27)+CHR$(
45)+CHR$(1)+RIGHT$(a$,LEN(a$)-a):GOTO 2
150
2160 a=INSTR(a$,CHR$(213)):IF a<>0 THEN
a$=LEFT$(a$,a-1)+CHR$(27)+CHR$(45)+CHR$(
0)+CHR$(32)+RIGHT$(a$,LEN(a$)-a):GOTO 21
60
2170 PRINT#8,a$;
2180 IF INKEY$<>" " THEN i=100
2190 NEXT
2200 NEXT:GOTO 1990
2210 nlq=1 :GOTO 1990 : '--- NLQ on
2220 nlq=0 :GOTO 1990 : '--- NLQ off -> P
ICA
2230 a$="Farbe/Tasten":GOSUB 1420
2240 an=6:a$(1)="Rand":a$(2)="Hintergrun
d":a$(3)="Schrift"
2250 a$(4)="Z-Y tauschen":a$(5)="Funktio
nstaste":a$(6)=CHR$(10)+"Hauptmenue"
2260 GOSUB 3150
2270 ON a GOTO 2280,2360,2440,2520,2570,
1460
2280 a$="Randfarbe":GOSUB 1420
2290 PRINT:PRINT " ";CHR$(240);" Hell
er"
2300 PRINT:PRINT " ";CHR$(241);" Dunk
ler"
2310 LOCATE 10,10:PRINT STR$(bor);" "
2320 a$=INKEY$:IF a$=CHR$(240)AND bor<26
THEN bor=bor+1
2330 IF a$=CHR$(241)AND bor>0 THEN bor=b
or-1
2340 IF a$=CHR$(13)THEN 2230
2350 BORDER bor:GOTO 2310
2360 a$="Hintergrund":GOSUB 1420
2370 PRINT:PRINT " ";CHR$(240);" Hell
er"

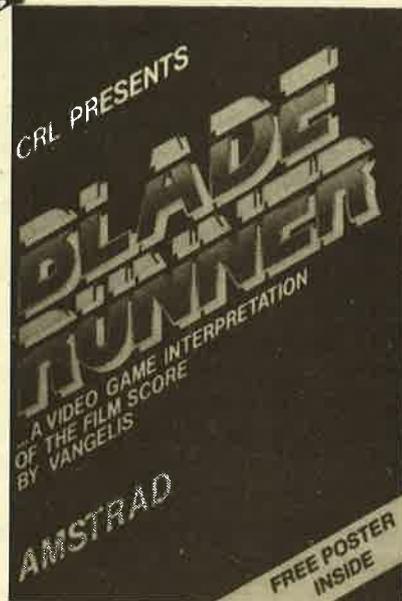
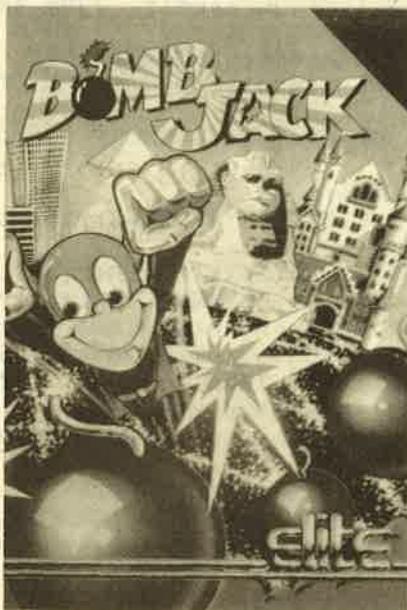
```

```

2380 PRINT:PRINT " ";CHR$(241);" Dunk
ler"
2390 LOCATE 10,10:PRINT STR$(papier);"
"
2400 a$=INKEY$:IF a$=CHR$(240)AND papier
<26 THEN papier=papier+1
2410 IF a$=CHR$(241)AND papier>0 THEN pa
pier=papier-1
2420 IF a$=CHR$(13)THEN 2230
2430 INK 0,papier:GOTO 2390
2440 a$="Schrift":GOSUB 1420
2450 PRINT:PRINT " ";CHR$(240);" Hell
er"
2460 PRINT:PRINT " ";CHR$(241);" Dunk
ler"
2470 LOCATE 10,10:PRINT STR$(schrift);"
"
2480 a$=INKEY$:IF a$=CHR$(240)AND schrif
t<26 THEN schrift=schrift+1
2490 IF a$=CHR$(241)AND schrift>0 THEN s
chrift=schrift-1
2500 IF a$=CHR$(13)THEN 2230
2510 INK 1,schrift:GOTO 2470
2520 a$="Z-Y tauschen":GOSUB 1420
2530 an=2:a$(1)="Deutsch":a$(2)="Amerika
nisch"
2540 GOSUB 3150:ON a GOSUB 2550,2560:GOT
O 2230
2550 KEY DEF 43,1,&7A,&5A:KEY DEF 71,1,&
79,&59:RETURN
2560 KEY DEF 71,1,&7A,&5A:KEY DEF 43,1,&
79,&59:RETURN
2570 a$="Funktionstasten":GOSUB 1420
2580 PRINT:INPUT" Nummer: ",a:a=INT(ABS
(a)):IF a>11 THEN 2570
2590 PRINT:INPUT" Text: ",a$:IF LEN(a
$)>10 THEN PRINT:PRINT" Zu lang !!":FO
R i=0 TO 999:NEXT:GOTO 2570
2600 KEY a,a$:GOTO 2230
2610 a$="Blockatz":GOSUB 1420:lr=PEEK(s
chnitt+7)-1:LOCATE#2,1,1:PRINT#2,SPACE$(
160)
2620 FOR i=0 TO 56
2630 c=0:x=bspei+i*80+lr:a$=SPACE$(PEEK(
schnitt+8)-lr)
2640 CALL begin+6,x,@a$:IF INSTR(a$," ")
=0 OR a$=SPACE$(PEEK(schnitt+8)-lr) THEN
2790
2650 LOCATE#2,1,1:PRINT#2,a$
2660 PRINT:PRINT i;" Blockatz ? (j/n)"
;CHR$(11);
2670 y$=LOWER$(INKEY$):IF y$="n" THEN 27
90
2680 IF y$<>"j" THEN 2670
2690 c=0:WHILE MID$(a$,PEEK(schnitt+8)-c
-lr,1)=" ":c=c+1:WEND:IF c=0 THEN 2790
2700 a$=LEFT$(a$,LEN(a$)-c):IF INSTR(a$,
" ")=0 THEN 2790
2710 x=1:FOR b=1 TO c
2720 a=INSTR(x,a$," ")
2730 IF a=0 THEN x=1:GOTO 2720
2740 WHILE MID$(a$,a,1)=" ":a=a+1:WEND:a
=a-1
2750 x=a+2
2760 a$=LEFT$(a$,a)+" "+RIGHT$(a$,LEN(a$
)-a)
2770 NEXT
2780 CALL begin+3,bspei+i*80+lr,@a$
2790 NEXT:POKE schnitt+6,0
2800 GOTO 1510
2810 REM ***** Briefkopf *****
2820 MODE 2:INPUT "Ort der Briefenstehun
g: ",o$
2830 IF o$="" THEN o$="- Ihr Ort -"
2840 INPUT "Datum: ",d$:IF d$="" THEN RE
TURN
2850 IF INSTR(d$,".")=0 THEN RETURN
2860 IF RIGHT$(d$,1)="." THEN d$=d$+"198
6"
2870 INPUT"Name des Absenders: ",n$
2880 IF n$="" THEN n$="= Ihr Name -"
2890 a$=n$+SPACE$(36-LEN(n$))+o$+", den
"+d$
2900 CALL begin:CALL begin+3,bspei+3,@a$
2910 INPUT"Strasse: ",a$
2920 IF a$="" THEN a$="- Ihre Anschrift
(Strasse) -":b$="- Ihre Anschrift (PLZ W
ohnort) -" ELSE INPUT"Wohnort: ",b$
2930 CALL begin+3,bspei+83,@a$:CALL begi
n+3,bspei+163,@b$
2940 PRINT:PRINT:PRINT"Name des Empfaeng
ers:"
2950 INPUT",a$
2960 REM Hier koennen Sie Daten aus ihre
r eigenen Adressverwaltung holen
2970 INPUT "Anrede : ",b$:c$=a$
2980 INPUT "Strasse : ",d$
2990 INPUT "Wohnort : ",e$
3000 INPUT "Abteilung: ",f$
3010 IF f$="" THEN a=8 ELSE a=7
3020 IF b$<>" " THEN CALL begin+3,bspei+3
+80*a,b$:a=a+1
3030 IF c$<>" " THEN CALL begin+3,bspei+3
+80*a,c$:a=a+1
3040 IF f$<>" " THEN CALL begin+3,bspei+3
+80*a,f$:a=a+1
3050 IF d$<>" " THEN CALL begin+3,bspei+3
+80*a,d$:a=a+1
3060 IF e$<>" " THEN CALL begin+3,bspei+8
0*a+83,e$
3070 PRINT:LINE INPUT "Betreff: ",a$:a$=
a$+" "
3080 CALL begin+3,bspei+3+17*80,a$
3090 LINE INPUT "Anrede: ",a$:IF a$="" T
HEN a$="Sehr geehrte Damen und Herren,"
3100 CALL begin+3,bspei+3+20*80,a$
3110 POKE schnitt,17:POKE schnitt+1,6
3120 POKE schnitt+7,4:POKE schnitt+8,70
3130 RETURN
3140 '*****
*
3150 FOR a=1 TO an
3160 LOCATE 5,5+a:PRINT a$(a)
3170 NEXT:a=1
3180 LOCATE 5,5+a:PRINT CHR$(24);a$(a);S
PACE$(15-LEN(a$(a)));CHR$(24)
3190 a$=INKEY$:IF a$=CHR$(240)THEN GOSUB
3250:a=a-1:IF a=0 THEN a=an
3200 IF a$=CHR$(241)THEN GOSUB 3250:a=a+
1:IF a=an+1 THEN a=1
3210 IF a$=CHR$(13)THEN RETURN
3220 IF VAL(a$)>0 AND VAL(a$)<=an THEN G
OSUB 3250:a=VAL(a$)
3230 IF a$=CHR$(224) THEN a=an:RETURN
3240 GOTO 3180
3250 LOCATE 5,5+a:PRINT a$(a);SPACE$(15-
LEN(a$(a))):RETURN

```

Wir schnüren Ihnen ein Super-Spiele-Paket !



Bestellschein ausschneiden, ausfüllen und einsenden an Verlag.Rätz - Eberle, Postfach 1640, 7518 Bretten

Ja, schicken Sie mir das Super-Spiele-Paket.

Ein Spiel kostet DM 24,-, 2 Spiele kosten DM 45,-, 3 Spiele kosten DM 66,-, 4 Spiele kosten DM 87,-. Hiermit bestelle ich

..... Cassette(n) Lightforce, Cassette(n) Ghosts'n Goblins, Cassette(n) Bomb Jack, Cassete(n) Blade Runner.

Name/Ort/Straße/Datum/Unterschrift

- Ich habe einen Scheck über DM beigelegt und muß daher keine zusätzlichen Porto- und Verpackungskosten bezahlen.
- Ich wünsche den Versand per Nachnahme und muß daher zusätzlich DM 5,70 für Porto und Verpackung bezahlen.



Schnelle Windows mit dem Window-Creator

Damit können Sie Windows erstellen und abspeichern

464

664

6128

Mit diesem Programm lassen sich Windows erstellen, die mit einem Schatten unterlegt sind. Das erzeugte Programm kann entsprechend umnummeriert und in eigene Programme implementiert werden. Die in dem Editor erstellten Windows können entweder in einem File abgelegt oder auf dem Drucker ausgegeben werden. Das versehentliche Beenden des Window-Editors läßt sich mit einer Sicherheitsabfrage abfangen. Windows können nur in MODE 1 erstellt werden.

Nach dem Start des Programms werden die an den Monitor angepaßten Farben eingestellt. Das Titelbild kann mit einem Tastendruck quittiert und die Anzahl der Windows muß eingegeben werden.

Bedienung des Programms

Beim Editor wird der Cursor mit den Pfeiltasten über den Bildschirm bewegt. Zuerst sollte man die linke, obere Ecke, dann die rechte, untere Ecke des zu erstellenden Windows mit der COPY-Taste markieren. Der Editor fragt nun nach der Window-Nummer (0-7). Das erstellte Window wird dann kurz gezeigt, und die restlichen Windows können erstellt werden.

Sind alle Windows erstellt, erscheint das Hauptmenü auf dem Bildschirm, und die erstellten Windows können in einem File auf Cassette/Diskette gespeichert oder auf dem Drucker ausgegeben werden. Die Möglichkeit, das Programm zu verlassen, besteht auch.

Wollen Sie Zeilen ausdrucken, sollten Sie Ihren Drucker bereit machen und eine Taste drücken. Nach dem Druckvorgang kehrt das Programm ins Hauptmenü zurück.

Beim Zeilen-Abspeichern werden sie aufgefordert, den Namen des zu speichernden Files einzugeben. Nun müssen Sie eine Cassette/Diskette einlegen und eine Taste drücken. Nach dem Abspeichern kehrt das Programm automatisch ins Hauptmenü zurück.

Zum Beenden müssen Sie eine Sicherheitsabfrage mit Ja oder Nein beantworten.

Ulrich Döwich

Window Creator

```

10 REM *****
20 REM * -- SCREEN - MAKER -- *
30 REM *
40 REM * (C) JUNI 1986 *
50 REM * VON *
60 REM * ULRICH DOEWICH *
70 REM *****

```

```

80 REM
90 IF PEEK(&BB01)=-&5C GOTO 95
91 IF PEEK(&B294)<>32 THEN SYMBOL AFTER
32:GOSUB 1130 : '--- 464
92 GOTO 100
95 IF PEEK(&B734)<>32 THEN SYMBOL AFTER
32:GOSUB 1130 : '--- 664/6128
100 MODE 1:PRINT"Benutzen Sie einen Grue
n- oder Colour Monitor ( G /, C ) ?"
110 GOSUB 1160
120 IF in$="B" THEN INK 0,13:INK 1,0:INK
2,15:INK 3,26:BORDER 13:PAPER 0:PEN 1:M
ODE 1:GOTO 150
130 IF in$="C" THEN INK 0,1:INK 1,0:INK
2,11:INK 3,26:BORDER 1:PAPER 0:PEN 1:MOD
E 1:GOTO 150
140 GOTO 110
150 n$="":o$="":z=1
160 REM ----- STARTBILD -----
170 WINDOW#1,1,39,2,25:PAPER#1,1:CLS#1
180 WINDOW#2,2,40,1,24:PAPER#2,2:CLS#2
190 p=9:o=5:PEN 1:a$="**** SCREEN -- MAK
ER ****":GOSUB 1050
200 p=16:o=9:a$=CHR$(164)+" Juni 1986":G
OSUB 1050
210 p=20:o=13:a$="von":GOSUB 1050
220 p=15:o=16:a$="Ulrich D[wich":GOSUB 1
050
230 p=13:o=22:a$="<<<<< TASTE >>>>":GOS
UB 1050
240 CALL &BB06:MODE 1
250 REM --- WIEVIELE WINDOWS SOLLEN ERST
ELLT WERDEN ---
260 WINDOW#1,1,39,2,25:PAPER#1,1:CLS#1:W
INDOW#2,2,40,1,24:PAPER#2,2:CLS#2
270 p=3:o=3:a$="Wieviele Windows wollen
Sie erstellen":GOSUB 1050:a$="( max. 5 )
?":p=3:o=6:GOSUB 1050
280 GOSUB 1160 :en=VAL(in$)
290 IF en<1 OR en>5 THEN PRINT CHR$(7):G
OTO 280
295 a$=in$:p=16:o=6:GOSUB 1050
300 REM --- ERSTELLUNG DES WINDOWS ---
310 PAPER 0:PEN 1:MODE 1:x=1:y=1:p=0:DIM
w1x(en):DIM wrx(en):DIM w1y(en):DIM wry
(en):DIM w1x1(en):DIM wrx1(en):DIM w1y1(
en):DIM wry1(en):DIM fe(en)
320 FOR du=0 TO en-1
330 IF INKEY(9)=0 THEN 440
340 IF INKEY(0)=0 THEN y=y-1
350 IF INKEY(8)=0 THEN x=x-1
360 IF INKEY(1)=0 THEN x=x+1
370 IF INKEY(2)=0 THEN y=y+1
380 IF y<1 THEN y=1
390 IF y>24 THEN y=24
400 IF x<2 THEN x=2
410 IF x>40 THEN x=40
420 LOCATE x,y:PRINT CHR$(143):FOR t=0 T

```

```

0 20:NEXT t:LOCATE x,y:PRINT " "
430 GOTO 330
440 IF p=0 THEN ly=y:lx=x:PRINT CHR$(7):
LOCATE x,y:PEN 3:PRINT CHR$(143):x=x+1:y
=y+1:p=p+1:PEN 1:CALL &BB00:GOTO 330
450 IF p=1 THEN ry=y:rx=x:PRINT CHR$(7):
LOCATE x,y:PEN 3:PRINT CHR$(143):CALL &B
B00
460 MODE 1:PEN 3
470 CALL &BB00:a$="WELCHE WINDOW NUMBER
(0-4): ":p=1:o=2:GOSUB 1050
480 GOSUB 1160
490 IF in$>"4" OR in$<"0" THEN PRINT CHR
$(7):GOTO 480
500 f=VAL(in$):a$=in$:p=29:o=2:GOSUB 105
0
510 FOR g=0 TO 1000:NEXT g
520 MODE 1
530 wlx(du)=lx-1:wrx(du)=rx-1:wly(du)=ly
+1:wry(du)=ry+1:wlx1(du)=lx:wrx1(du)=rx:
wly1(du)=ly:wry1(du)=ry:fe(du)=f
540 n$=n$+STR$(z)+"WINDOW#"+STR$(f)+", "+
STR$(lx-1)+", "+STR$(rx-1)+", "+STR$(ly+1)
+", "+STR$(ry+1)
550 n$=n$+":PAPER#"+STR$(f)+", 1:CLS#"+ST
R$(f)+CHR$(13)+CHR$(10)
560 z=z+1:o$=o$+STR$(z)+"WINDOW#"+STR$(f
)+", "+STR$(lx)+", "+STR$(rx)+", "+STR$(ly)
+", "+STR$(ry)
570 o$=o$+":PAPER#"+STR$(f)+", 2:CLS#"+ST
R$(f)+CHR$(13)+CHR$(10)
580 FOR ud=0 TO du
590 WINDOW#fe(ud)+3,wlx(ud),wrx(ud),wly(
ud),wry(ud):PAPER#fe(ud)+3,1:CLS#fe(ud)+
3
600 WINDOW#fe(ud)+3,wlx1(ud),wrx1(ud),wly
1(ud),wry1(ud):PAPER#fe(ud)+3,2:CLS#fe(
ud)+3
610 NEXT ud
620 FOR t=0 TO 1500:NEXT t
630 x=x+1:y=y+1:p=0:z=z+1:NEXT du
640 REM ----- MENU -----
650 MODE 1
660 WINDOW#1,1,39,2,25:PAPER#1,1:CLS#1:W
INDOW#2,2,40,1,24:PAPER#2,2:CLS#2
670 a$="1. -- Zeilen ausdrucken":p=3:o=6
:GOSUB 1050
680 a$="2. -- Zeilen abspeichern":p=3:o=
9:GOSUB 1050
690 a$="3. -- Beenden":p=3:o=12:GOSUB 10
50
700 a$="BITTE WAEHLEN SIE !":p=3:o=20:GO
SUB 1050
710 GOSUB 1160
720 IF in$="1" THEN 840
730 IF in$="2" THEN 910
740 IF in$="3" THEN 770
750 GOTO 710
760 REM ----- BEENDEN -----
770 MODE 1:PAPER 0:PEN 3
780 a$="SIND SIE SICH SICHER ( J / N ) ?
":p=1:o=2:GOSUB 1050
790 GOSUB 1160
800 IF in$="J" THEN MODE 1:END
810 IF in$="N" THEN 650

```

```

820 GOTO 790
830 REM ----- AUDRUCKEN -----
840 MODE 1:PAPER 0:PEN 3
850 a$="DRUCKER BEREITMACHEN. <TASTE>":p
=1:o=2:GOSUB 1050
860 CALL &BB06
870 PRINT#8,n$
880 PRINT#8,o$
890 MODE 1:GOTO 650
900 REM ----- ABSPEICHERN -----
910 MODE 1:PAPER 0:PEN 3
920 a$="NAME DES PROGRAMMES ":p=1:o=2:G
OSUB 1050
930 LOCATE 1,25:INPUT"",e$
940 CLS
950 a$="CASSETTE / DISKETTE EINLEGEN. <T
ASTE>":p=1:o=2:GOSUB 1050
960 CALL &BB06:PRINT:PRINT:PRINT
970 OPENOUT e$
980 PRINT#9,n$
990 PRINT#9,o$
1000 CLOSEOUT
1010 CLS:a$="Ok. ":p=10:o=10:GOSUB 1050
1020 FOR t=0 TO 1000:NEXT t
1030 MODE 1:GOTO 650
1040 REM ----- VERGROESSERUNG -----
1050 LOCATE p,o:PRINT CHR$(22)CHR$(1):LO
CATE p,o
1051 FOR v=0 TO 1
1052 LOCATE p+v,o-v :PEN 1+2*v
1060 FOR a=1 TO LEN(a$):b=ASC(MID$(a$,
a,1)):adr=(b-32)*8+1+HIMEM
1070 FOR s=0 TO 7:s(s)=PEEK(adr+s):N
EXT s
1080 SYMBOL 254,s(0),s(0),s(1),s(1),
s(2),s(2),s(3),s(3)
1090 SYMBOL 255,s(4),s(4),s(5),s(5),
s(6),s(6),s(7),s(7)
1100 PRINT CHR$(254)CHR$(10)CHR$(8)C
HR$(255)CHR$(11);
1104 NEXT
1105 NEXT
1110 PRINT CHR$(22)CHR$(0):RETURN
1120 REM ----- SYMBOLS -----
1130 SYMBOL 91,&66,0,&3C,&66,&66,&66,&3C
,0
1140 RETURN
1150 REM ----- TASTATURABFRAGE -----
1160 in$=UPPER$(INKEY$):IF in$="" THEN 1
160 ELSE RETURN

```

CPC-Hardware

AMX-MOUSE 239.-

Elektronische Produkte für
CPC 464/664 (für 6128 auf Anfrage)

Sprachsynthesizer (Cas.) 89.-
Sprachsynthesizer (ROM) 139.-
Lightpen (Cas.) 69.-
Lightpen (ROM) 98.-
64K-Speichererweiterung 149.-
256K-Speichererweiterung 299.-
256K-RAM-Disk 299.-
Druckerkabel 49.-

Panasonic-Drucker

KX-P 1080 699.-
KX-P 1091 879.-
KX-P 1092 1149.-
CITIZEN 120 D 699.-
OKIDATA Okimate 20 699.-

Wir führen alle vortex-Produkte
sowie Software und Bücher des
MAT-Vorlags.
Wegen Versand per Nachnahme
gewähren wir bei vortex-Produkten
2% Skonto.

VORTEX SP-256 288.-
VORTEX SP-512 375.-

CPC-SOFTWARE

TASWORD 454 59.-
TASWORD 464 D (Disk) 89.-
TASPRINT 34.-
LASER BASIC 49.-
C-BASIC-COMPILER 198.-
Weitere Anwenderprogramme in unserer Liste CPC 2/86.
Fordern Sie unsere aktuelle Preisliste
CPC 2/86 an. Oder bestellen Sie gleich per
Nachnahme bei:

U. Kunz Junge Halden 3
D-7500 Karlsruhe 41

Telefonische Auftragsbestätigung bis 20 Uhr: 07 21 / 48 26 76.

Für unsere Puzzle-Freunde: Madonna beim Luftholen!

Unser neues Puzzle-Bild zeigt die bekannte Popsängerin Madonna beim Luftholen

464

664

6128

Das Puzzlebild dieses Monats zeigt die Sängerin Madonna beim Luftholen. Tippen Sie das hier abgedruckte Listing ein, speichern Sie es danach mit SAVE "PUZZ-MAD.LDR" und geben dann RUN ein. Es steht nun unter dem Namen "PUZZ-MAD.PIC" im Speicher. Tritt in den DATA-Zeilen ein Fehler auf, so wird dieser zusammen mit der Zeilennummer angezeigt. Überprüfen Sie diese Zeile Ihres Programms dann nochmals anhand des abgedruckten Listings und speichern Sie es nach dem Verbessern wieder mit SAVE "PUZZ-MAD.LDR" ab. Um dann aber auch puzzeln zu können, benötigen Sie zusätzlich die Listings aus Heft 4/86 und 6/86, die zum Verschieben der Puzzle-Teile auf dem Bildschirm dienen.

Christoph Schillo

MC-Generator

```

1 ' MC-Generator fuer
2 ' Puzzlerweiterung Teil 6
3 ' Madonna (PUZZ-MAD.PIC)
4 ' (c) Sep. 1986, Schillo
5 '
100 MEMORY 19999
110 pc=20000
120 MODE 1:BORDER 26:INK 0,26:INK 1,0:INK
K 2,6:INK 3,17
130 FOR z=10000 TO 10160 STEP 10:LOCATE
10,12:PRINT"Zeile";z;:c=0:READ x$,s
140 FOR a=0 TO LEN(x$)/2-1:a$=MID$(x$,a*
2+1,2):POKE pc,VAL("&"+a$):c=c+VAL("&"+a
$):pc=pc+1:NEXT:IF s<>c THEN PRINT"Beric
htigen!":END
150 PRINT"Ok":NEXT
160 FOR a=0 TO 1546 STEP 64:FOR y=0 TO 7
:FOR x=0 TO 1:FOR z=0 TO 3:POKE (49152+x
*80+y*2048+z+za),PEEK(20000+a+z+y*4+x*32
):NEXT z,x,y:za=za+4:z1=z1+1:IF z1=5 THE
N z1=0:za=za-20+160
170 NEXT
180 PRINT"Achtung, abspeichern!"
190 SAVE"Puzz-mad.pic",b,20000,1605,0
10000 DATA "0000001100000011000010000000
0000000000A000000004000000000000000000
000000000000000004000000010800000000000010
800000000000000000000000000000000000000000
0070000010F0", 5074
10010 DATA "000030F00000070C00000060000000
C01100000803300108077001000FFFFFFFFFFFFF

```

```

FFFFFFFFFFFFFFFFFFFF11FFFFFFFF0077FFFF0033
FFEE8000FF00C000000000E00000000000000007700
0000FF880000FFCC0000FFEC8000FFF40000FFFF
FFFFFFFFFFFF", 12571
10020 DATA "FFFFFFFFFFFFFFFFFFFFFFFFFCC11
FFFF0000FFFF000077FF000077FF002077FF0060
77FF00C077FF008000370080001D000001090000
0107FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFF", 18069
10030 DATA "FFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFF05FFFFFFFF0060000020D00000000000000000
0000000000000000000000001080000020000000C0
0000200000000000000000000000000000000000
0000000000000000000000001011FF001011FF0010
00FF00100033", 6500
10040 DATA "000000000000000000000000000000
00000000000000000000000000000000000000000
0011001111FF000077FF00000000FFF00000FCE0
0000F8800000E000000040000000000000000000
000000000000000000044000000880000033000011
EE00FFFF8800", 3471
10050 DATA "FFFF00002200000000000000000000
01080000010100000103000001060000030C0000
0300000000000000000000000000000000000000
0000000000000000000000000000000000000000
0000073BFFFF0C33FFFF0811FFFF000077FF0000
00FF00000077", 3016

```



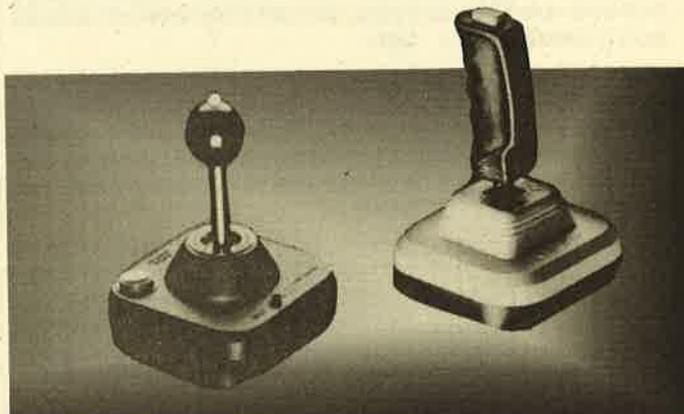
Der Super-Ypsilon-Stecker mit allen Schikanen

Das unten abgebildete Ding ist ein wahrer Zauberkasten: ein Ypsilonstecker mit Dauerfeuer und Wechselimpuls

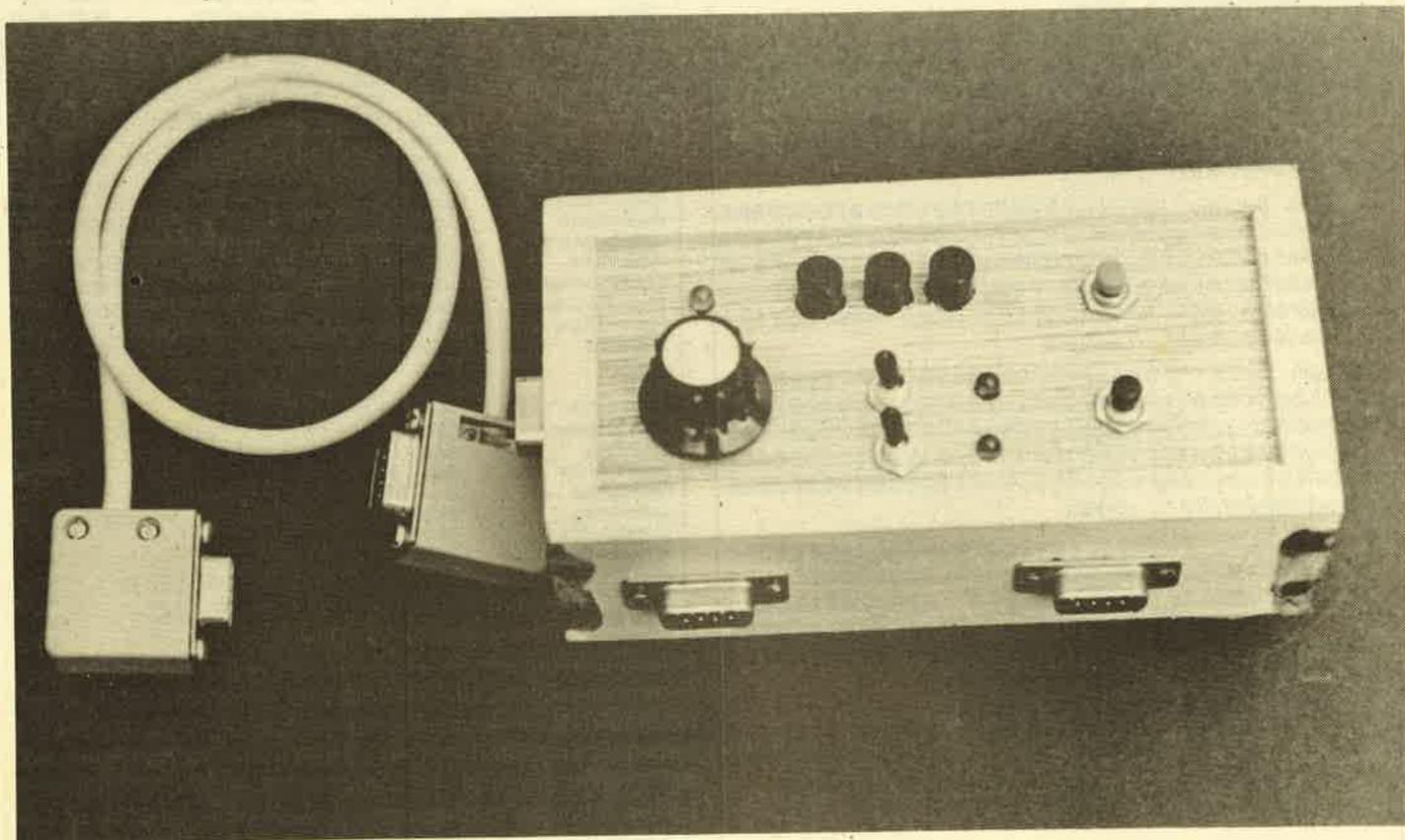
Der Schneider CPC wird zwar überwiegend professionell genutzt, aber welcher Computer-Freak möchte nicht auch mal gerne eine Runde auf seinem Computer spielen? Solange diese Spiele nur für eine Person geschrieben sind, gibt es keine Probleme. Aber ein großer Teil ist für zwei Spieler gedacht, und hier zeigt sich ein Schwachpunkt des CPCs: Er besitzt nur einen Joystickport.

Für dieses Problem gibt es zwei Lösungsmöglichkeiten. Entweder man kauft sich zwei Schneider-Joysticks, von denen der erste eine zweite Joystickbuchse besitzt. Nachteilig ist hierbei, daß die Kabel nicht lang genug sind und so nur auf engstem Raum gespielt werden kann. Die zweite Möglichkeit ist ein Ypsilonstecker mit Autofeuer und Wechselimpuls. Das Gerät wird am Joystickport des Schneider CPC angeschlossen. Durch das längere Verbindungskabel kann man sich auch weiter entfernt vom Computer und Bildschirm hinsetzen. Ein 8adriges Kabel mit 9poligen Steckern stellt die Verbindung zum Computer her. An der linken Seite des Gerätes befinden sich die beiden Eingänge für Joystick 1 und 2.

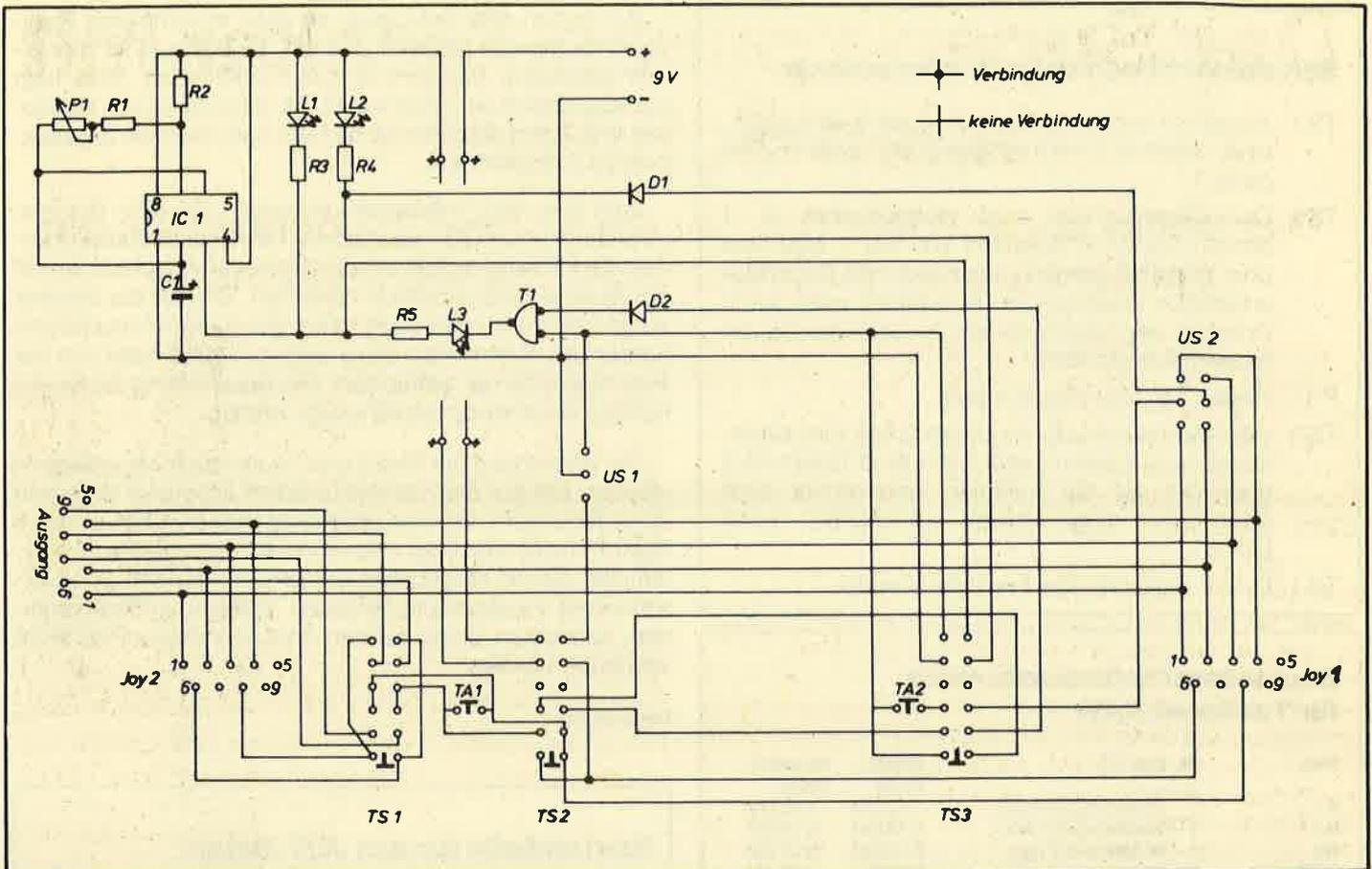
Für diejenigen unter Ihnen, die gerne auch mal die sogenannten "Ballerspiele" absolvieren, sind zwei Autofeuer-



funktionen eingebaut: Bei einer Stellung von Schalter US1 hat man durchgehend Autofeuer, wenn die Taste TS2 gedrückt wird. Wird Schalter US1 umgeschaltet, so hat man nur Autofeuer, wenn der Schußknopf am Joystick gedrückt wird. Die Geschwindigkeit läßt sich durch das Potentiometer P1 regulieren und wird durch die Leuchtdiode L1 angezeigt. Taste TS1 bewirkt, daß Joystick 1 auf den Joystickeingang 2 und Joystick 2 auf den Joystickeingang 1 des Computers gelegt wird.



Der fertig montierte Ypsilon-Switch und das Verbindungskabel zum CPC. Mit ihm ist zusätzlich noch regulierbares Dauerfeuer für beide Joysticks möglich.



Sportspiele erfreuen sich in letzter Zeit immer größerer Beliebtheit. Spiele wie z.B. Summergames oder Wintersports fehlen in kaum einer Spielesammlung. Doch sie haben den Nachteil, daß sie einen Joystick benötigen, der hohen Belastungen ausgesetzt werden kann. Solche Joy-

sticks sind aber sehr teuer. Um dieser Anschaffung aus dem Weg zu gehen, habe ich einen Wechselimpulsgeber eingebaut, den man durch TS2 und TS3 aktiviert.

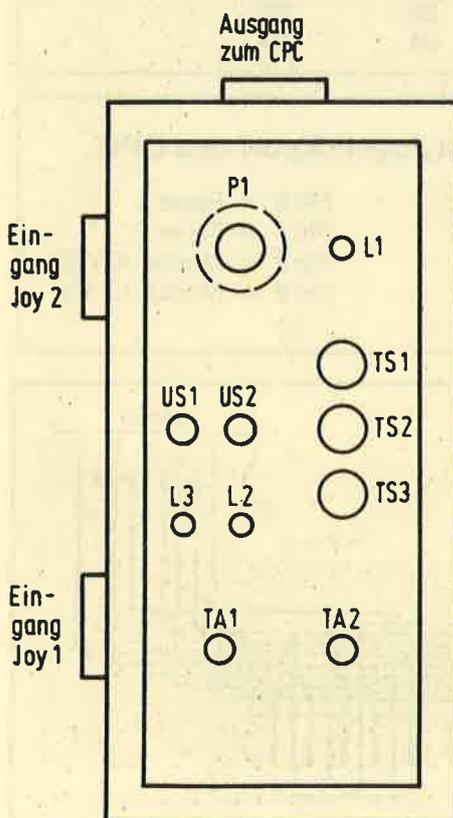
Je nachdem, wie der Schalter US2 liegt, wird bei gedrücktem Taster TA2 entweder die "Links-Rechts"- oder die "Oben-Unten"-Funktion übernommen. Der Taster TA1 ersetzt dabei den Schußknopf am Joystick. Die Geschwindigkeit läßt sich wieder mit Potentiometer P1 regulieren.

Die verschiedenen Funktionen probieren Sie am besten aus, wenn sich kein Programm in Ihrem CPC befindet.

Beim Bau des Gerätes ist unbedingt darauf zu achten, daß die Dioden laut Schaltplan eingebaut werden und kein Strom zum Computer fließen kann. Der Ypsilonstecker wird mit einer 9-V-Block-Batterie versorgt, während der Computer mit nur 5 V gespeist wird. Der Computer könnte also Schaden nehmen, wenn die Dioden falsch herum eingesetzt werden. Auch könnte es sein, daß sich der Kondensator entlädt, wenn man zuviel Lötfett genommen hat. Einige Lötfette sind leitend. Deshalb sollte man die Platine nach dem Bau noch mit Alkohol (mit einem kleinen Pinsel) reinigen. Der Kondensator muß sich nach langer Lagerung langsam aufladen, so daß die Leuchtdioden anfangs ganz unregelmäßig aufleuchten. Das Potentiometer P1 muß dabei voll aufgedreht sein.

Der Transistor ist im Schaltplan der Einfachheit halber wie alle Bauteile in der Draufsicht gezeichnet. Die Leuchtdioden dürfen nicht verkehrt herum eingebaut werden, da sie nur in einer Richtung Strom durchlassen. Der Widerstand R1 könnte auch durch ein Trimpotentiometer mit 10K Ω ersetzt werden, das zum Schluß auf den besten Wert getrimmt werden müßte.

H. Wollenhaupt



Schema des fertig montierten Y-Schalters in der Draufsicht. P1 ist der Regler für die Geschwindigkeit des Dauerfeuers.

Schaltfunktionen beim Ypsilonstecker

- TS 1 Joystick 1 wird umgelegt auf Joystick-Eingang 2 bzw. Joystick 2 wird umgelegt auf Joystick-Eingang 1.
- TS 2 Dauerfeuerfunktion wird eingeschaltet. (L 1 blinkt.) Durch Umschalten von US 1 wird jetzt über Joystick-Eingang 1 entweder die Dauerfeuerfunktion übernommen (L 3 blinkt) oder durch Drücken der Feuertaste am Joystick der Dauerfeuerimpuls aktiviert.
- P 1 Regelt den Dauerfeuerimpuls.
- TS 3 Wechselimpuls (z.B. für Sportspiele) wird eingeschaltet. (L 2 blinkt zusätzlich.) Wird Taster TA 2 gedrückt, ist die Funktion links-rechts oder oben-unten eingeschaltet (umschaltbar durch US 2).
- TA 1 Dieser Taster hat die Funktion "Feuer".

Bauteilliste/Selbstkostenliste für Ypsilonstecker

Pos.	St.	Bauteil	Einzel-Preis	Gesamt-Preis
P 1	1	Potentiometer 1M Ω	1.40 DM	1.40 DM
R 1	1	Widerstand 3,7 k Ω	0.10 DM	0.10 DM
R 2	1	Widerstand 27 k Ω	0.10 DM	0.10 DM
R3/R4/R5	3	Widerstand 390 Ω	0.10 DM	0.30 DM
IC 1	1	555 NE (Timer)	1.25 DM	1.25 DM
C 1	1	Kondensator 1 μ F	0.25 DM	0.25 DM
T 1	1	Transistor BC 238 A	0.30 DM	0.30 DM
D1/D2	2	Diode 1 N 4448	0.25 DM	0.50 DM
L 1	1	LED grün 5 mm	0.20 DM	0.20 DM
L2/L3	2	LED rot 5 mm	0.20 DM	0.40 DM
TA1/TA2	2	Taster 1polig	0.80 DM	1.60 DM
US1	1	Umschalter 1xUM	1.50 DM	1.50 DM
US2	1	Umschalter 2xUM	1.60 DM	1.60 DM
TS1/TS2/TS3	1	Tastensatz 3 Tasten Einzelausl. je 4xUM	6.60 DM	6.60 DM
	1	Batterieanschluß 9V	0.30 DM	0.30 DM
	1	IC-Sockel 8polig	0.35 DM	0.35 DM
	2	Steckerleisten 9polig	2.70 DM	5.40 DM
	1	Buchsenleiste 9polig	3.25 DM	3.25 DM
	1	Drehknopf	1.60 DM	1.60 DM
	1	Platine ca. 50x30 mm	0.20 DM	0.20 DM
	1	Gehäuse 150x80x50 mm	3.00 DM	3.00 DM
			-----	-----
			30.20 DM	30.20 DM
			-----	-----

Bauanleitung für Y-Adapter

Die erste Schwierigkeit beim Selbstbau eines Y-Adapters ist die falsche Pinbelegung für den Joystickport im Handbuch des CPC 464. Eine richtige Belegung finden Sie in unserem abgedruckten Schaltplan. Auch eine komplette und richtige Wertetabelle für den JOY-Befehl haben wir abgedruckt, die zumindest im Handbuch des CPC 464 nicht zu finden ist. Mit Hilfe dieser Tabelle und der folgenden Basic-Zeile kann man die einwandfreie Funktion des Y-Adapters testen:

10 PRINT JOY (0), JOY (1) :GOTO 10

Wir haben den Y-Adapter an allen erreichbaren Zwei-Joystick-Spielen getestet und mit "Exploding Fist" nur eines gefunden, bei dem er nicht funktioniert. Dies liegt wahrscheinlich am Spiel selbst, da ihm auch durch wahlloses Verbinden aller Pins keine Reaktion über den Joystickport zu entlocken war.

Zum Bau des Y-Adapters benötigt man eine Buchse, zwei Stecker und 16 Kabelstücke. Die Stecker dürften in jedem Elektronikgeschäft erhältlich sein. Im Anschlußplan ist die Buchse spiegelbildlich nummeriert. Die auf die Stecker aufgedruckte Numerierung ist bei genauerem Hinsehen zu entdecken, indem man diese von vorne und nicht von der Anschlußseite her betrachtet! Die Beschriftung ist für die richtige Verbindung der Pins sehr wichtig.

Die Verbindung der Pins 5 und 7 kann auch weggelassen werden. Die zur Zeit handelsüblichen Joysticks benutzen diese Pins nicht. Da man aber nicht wissen kann, ob es in Zukunft nicht Joysticks oder andere Zusatzgeräte zu kaufen gibt, die die Möglichkeit der CPC-Computer, zwei verschiedene Feuerknöpfe für jeden Joystick zu unterscheiden, ausnutzen, sollte auf den Anschluß dieser Pins nicht verzichtet werden.

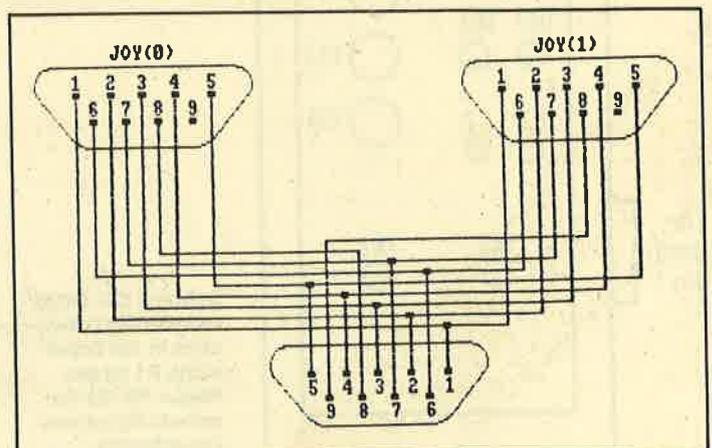
Peter Kluthen

Wertetabelle für den JOY-Befehl

Funktion	JOY (0)	JOY (1)
oben	1	1
unten	2	2
links	4	4
rechts	8	8
Feuer 1	16	16
Feuer 2	32	32
frei	64	64

Pinbelegung am Joystickport des CPC

Pin 1 = oben	Pin 6 = Feuer 1
Pin 2 = unten	Pin 7 = Feuer 2
Pin 3 = links	Pin 8 = Masse JOY (0)
Pin 4 = rechts	Pin 9 = Masse JOY (1)
Pin 5 = frei	



ABO

Abo-Bestellschein

Ich möchte das CPC-Magazin in Zukunft regelmäßig zugeschickt bekommen. Die Abodauer beträgt 12/6 Ausgaben und kann bis spätestens 4 Wochen vor Aboende wieder gekündigt werden. Der Abonnementspreis beträgt 66.- DM einschließlich Mehrwertsteuer und Versandkosten. Für Bestellungen aus dem europäischen Ausland wird es aber nur ein wenig teurer: Hier kostet das Abo 75.- DM. Ab sofort gibt es auch ein Kombi-Abo: CPC-Magazin + Software-Cassette zum Heft.

- Ich wünsche ein Jahresabo mit 12 Ausgaben
- Ich wünsche ein Halbjahresabo mit 6 Ausgaben zum halben Preis (33.- DM/37.50 DM)
- Ich wünsche 12 Ausgaben + Cassette (216.-/225.- DM)
- Ich wünsche 6 Ausgaben + Cassette (108.-/112.50 DM)

.....
Name/Vorname

.....
Straße

.....
PLZ

.....
Ort

Ich bezahle wie folgt:

Ich bestelle ab Ausgabe:

- Scheck liegt bei
- Vorauskasse auf Postscheckkonto Karlsruhe
Nr. 43423-756

Mir ist bekannt, daß ich diese Bestellung innerhalb 8 Tagen widerrufen kann und bestätige dies mit meiner Unterschrift. (Dieses Widerrufsrecht ist per Gesetz vorgeschrieben.)

.....
Datum/Unterschrift

Diesen Bestellschein ausschneiden oder fotokopieren und an das CPC-Magazin, Postfach 1640, 7518 Bretten schicken.

CPC **Magazin**
Für alle Schneider Computer

CASSETTENABOS
in unserer Rubrik
»Fingerschonend«

Funktionstastenbelegung für den vortex-Monitor

Mit diesem Programm wird die Eingabe beim vortex-Monitor erheblich vereinfacht. Insgesamt 12 Tasten sind mit Eingabesequenzen belegt.

Mit der Speichererweiterung von vortex bekommt man einen RAM-Monitor; das VDOS 2.0 enthält zusätzlich einen Diskettenmonitor. Da die Eingabe jedoch recht umständlich ist, lag es nahe, die Bedienung über die Belegung der Funktionstasten komfortabler zu gestalten. Da die vortex-Programme im ROM stehen, also nicht verändert werden können, blieb als einzige Eingriffsmöglichkeit die Unterbrechungsebene. Das Programm "XMONKEYS.BIN" erzeugt einen Frame Ticker, also ein Hintergrundprogramm, das 50mal in der Sekunde aufgerufen wird. Dadurch stehen auf dem Ziffernblock folgende Tastenfunktionen zur Verfügung:

Taste	Wirkung
Null	Pufferadresse-&100, D, ENTER
Punkt	D, ENTER
kl. ENTER	L, ENTER
Vier	Sektor-1, S, ENTER
Fünf	Sektor+1, S, ENTER
Sieben	Track-1, T, ENTER
Acht	Track+1, T, ENTER
Drei und Sechs	R, D, ENTER
Eins	R, D, ENTER, Sektor-1, S, ENTER
Zwei	R, D, ENTER, Sektor+1, S, ENTER
Neun	R, Pufferadresse+&200, Sektor+1
Copy	W, Pufferadresse+&200, Sektor+1

Weiterhin wird die Statuszeile des Diskettenmonitors verlängert. Neu sind BL für die aktuelle Blocknummer und R für die sogenannte Resultphase des zuletzt gelesenen Sektors (näheres im Floppybuch von Data Becker). Von besonderer Bedeutung sind wohl die Tasten Null und Punkt (vorwärts und rückwärts im Speicher blättern) sowie Eins und Zwei (fortlaufend Sektoren lesen und anzeigen). Für spezielle Anwendungen sind Neun und Copy gedacht. Damit ist es möglich, Disketten zu kopieren. Sinnvoll kann das z.B. sein, wenn man Disketten mit einzelnen fehlerhaften Sektoren retten will. Dazu müssen bei jedem Schritt erneut die Anfangswerte für Track, Sektor und Puffer eingestellt werden. Die Pufferadresse sollte möglichst niedrig liegen, wenn der Speicher frei ist, z.B. bei &200. Wenn "XMONKEYS.BIN" bei &A200 betrieben wird, können jeweils 40 KByte des Disketteninhalts geladen werden. Dazu muß man nur die Taste Neun lange genug gedrückt halten. Es besteht eine Absicherung, so daß die Pufferadresse unterhalb des Bereiches von "XMONKEYS.BIN" bleibt und der Ladevorgang dann abgebrochen wird. Wenn die Anfangswerte wiederhergestellt sind, kann mit Copy der Speicherinhalt auf eine neue Diskette übertragen werden. Um ein versehentliches Schreiben von Sektoren auszuschalten, wurde diese Funktion auf die räumlich getrennte Copytaste gelegt.

In der Basic-Umgebung verrichten die Tasten ihre gewohnte Funktion. Erst wenn der vortex-Monitor eingeschaltet wird, trägt "XMONKEYS" eigene Werte in die Tastaturtabelle ein. Die ursprünglichen Werte werden gespeichert und hinterher wiederhergestellt. Als Test, ob der vortex-Monitor aktiv ist, dient die momentane ROM-Nummer. Sobald ROM 6 (Speichererweiterung) oder 5 (VDOS) aktiv sind, wird auch "XMONKEYS.BIN" aktiv. Wenn die ROMs bei anderen Nummern betrieben werden (die Information bekommt man mit dem Befehl ROMS), müssen die zugehörigen Werte in DATA-Zeile 200 (jeweils nach dem FE) geändert werden.

Nützlicher Maschinensprachetrick

Zur weiteren Funktionsweise des Programms soll hier nur noch ein nützlicher Maschinensprachetrick weitergegeben werden. Im Firmwarehandbuch und ähnlichen Zusammenstellungen der Einsprungadressen für die Betriebssystemroutinen wird als Aussprungbedingung auch angegeben, welche Register jeweils "zerstört" werden. Es lohnt sich aber, im ROM-Listing nachzuschlagen, welche Werte in diesen Registern tatsächlich zurückbleiben. Häufig sind es Systemadressen, die sonst kaum verwendet werden können, da sie bei den 3 CPC-Typen so unterschiedlich liegen. "XMONKEYS" bekommt auf diesem Wege die Adresse der Tastaturtabelle, so daß die Tastaturabfrage über den direkten Speicherzugriff wesentlich schneller und einfacher wird.

"XMONKEYS.BIN" wird auf die übliche Weise mit einem Basic-Ladeprogramm erzeugt. Sobald alles richtig abgetippt ist und bei RUN keine Fehlermeldungen mehr erscheinen, kann der SAVE-Befehl aus Zeile 70 eingegeben werden. Dies ist dann aber erst eine vorläufige Version. Sobald diese getestet ist und zufriedenstellend arbeitet, kann im zweiten Durchlauf der erste Wert in DATA-Zeile 160 in 57 geändert (das W für Sektor schreiben) und die endgültige Version erzeugt werden. Da ja auf keinen Fall durch Fehler im Programm Sektoren der Diskette überschrieben werden dürfen, wurde diese Sicherheitsstufe eingebaut.

Auch die Anwendung des Programms unterscheidet sich nicht von den vielen Hilfsprogrammen, die sonst im CPC-Magazin veröffentlicht werden. Da "XMONKEYS.BIN" verschiebbar ist, kann man sich eine geeignete Ladeadresse aussuchen (wie üblich oberhalb &4000) und braucht dann nur die zwei Befehle LOAD "XMONKEYS.BIN", ladeadresse: CALL ladeadresse - und der vortex-Monitor kann "fliegen".

Gerhard Knapieski

Datei suchen und Speicherplatz feststellen

464

664

6128

Wer stolzer Besitzer einer vortex-Station ist, den werden wohl folgende Tips interessieren. Da wäre zum einen die Einstellung des Userbereichs. Beim Schneider-DOS wird dieser mit POKE &A701, u eingestellt, beim vortex-DOS mit POKE &A705, u. Dabei gilt $0 \leq u \leq 225$ für die Usernummer. Hat man nun aus Versehen ein File gelöscht, so läßt sich dieses mit folgender Sequenz wieder retten.

```
:ERA, "filename.typ" File wird versehentlich gelöscht
POKE &A705, &E5 "Usernummer" des gelöschten Files
LOAD "filename.typ"
POKE &A705, 0 normaler Userbereich
SAVE "filename.typ"
```

Wollen Sie mehrere Files retten, so müssen Sie diese auf einer anderen Diskette abspeichern, da sonst die Einträge der noch gelöschten Files im Directory zerstört werden, weil solche mit der Usernummer &E5 vom DOS nach den CP/M-Regeln als freie Einträge behandelt und beim Speichern eines Files überschrieben werden.

Zum Umschalten der Laufwerke dient beim Schneider-DOS POKE &A700, n und beim vortex-DOS POKE &A706, n mit den Werten $n = 0 = A$ und $n = 1 = B$. Natürlich kann man genauso mit PEEK das momentan aktive Laufwerk feststellen.

Ein weiteres Problem taucht auf, wenn man aus einem Programm heraus feststellen möchte, ob eine Datei überhaupt auf Diskette vorhanden ist. Dieses Problem wird durch Listing 1 gelöst. Natürlich kann die Ausgabe auch auf den Drucker umgeleitet werden. Dazu sei noch folgendes vermerkt: Jeder Eintrag im Directory besteht aus 14 Bytes. Byte 0 ist die Usernummer, 225 bedeutet Ende des Directories, Byte 1 bis 8 sind der Dateiname, Byte 9 bis 11 der Dateityp und Byte 12 und 13 stehen für die Größenangabe in Kilobytes, und zwar mit dem Lowbyte in 12 und dem Highbyte in 13.

Wenn man innerhalb eines Programms erfahren will, wieviel Speicherplatz überhaupt noch zur Verfügung steht, so kann man Listing 2 leicht modifiziert als Unterprogramm einbauen.

Kron

Listing 1

```
1 'CATSUCH.BAS (c) KRON 1986
2 '
100 ' Beispiel fuer
110 ' Puffer-Bereich ab &8000
120 '
130 MEMORY &7FFF
140 MODE 2: CLEAR
150 ' nachfolgende Funktion ist Ersatz f
uer -CAT-
160 ' als variable Eingabe gilt Startwert
t fuer Puffer-Bereich
```

```
170 DEF FNa$(x) = CHR$(&11)+CHR$(0)+CHR$(
(x) +CHR$(&CD)+CHR$(&9B)+CHR$(&BC) +CHR$(
&C9)
180 ' nachfolgende Funktion legt Start d
es Puffers fest
190 DEF FNsadr(x)=256*x
200 x=&80 : ' Puffer-Start fuer Director
y = Memory-Wert + 1
210 AS=PEEK(&BB5A) : ' TXT OUTPUT off
220 POKE &BB5A, &C9
230 aa$=FNa$(x) : adra=PEEK(&aa$+2)*256+P
EEK(&aa$+1) : CALL adra
240 POKE &BB5A, AS : ' TXT OUTPUT on
250 ct=0 : mng=0 : ' Anzahl der belegten
Directory-Eintragungen
260 PRINT "&HEX$(FNsadr(x), 4)": Beginn d
es Directories"
270 FOR i=FNsadr(x) TO FNsadr(x)+&7FF ST
EP 14
280 PRINT "&HEX$(i, 4);
290 IF PEEK(i)=&FF GOTO 430 : 'FF=Ende
des Directories // 00 unter AMSDOS
300 mng=mng+1
310 PRINT USING: ### "; PEEK(i);
320 FOR v=1 TO 8
330 PRINT CHR$(1)CHR$(PEEK(i+v) AND
&X1111111);
340 NEXT
350 PRINT ".";
360 PRINT CHR$(1)CHR$(PEEK(i+ 9) AND &
X1111111); : r=PEEK(i+ 9) AND &X10000000
370 PRINT CHR$(1)CHR$(PEEK(i+10) AND &
X1111111); : s=PEEK(i+10) AND &X10000000
380 PRINT CHR$(1)CHR$(PEEK(i+11) AND &
X1111111);
390 PRINT USING " #####K "; PEEK(i+12)
+PEEK(i+13)*256; : 'Kbytes 0-65535
400 IF r THEN r=0: PRINT "R/O "; ELSE P
RINT "R/W ";
410 IF s THEN s=0: PRINT "SYS" ELSE P
RINT "DIR"
420 NEXT
430 PRINT "Ende des Directories"
440 a=FNsadr(x)
450 PRINT: PRINT CHR$(24) " Beginn des Dir
ectories: = "CHR$(24) "&HEX$(a, 4)
460 PRINT: PRINT CHR$(24) " Anzahl der Ein
traege : = "CHR$(24) mng
470 PRINT
```

Listing 2

```
1 ' FREI.BAS (c) KRON 1986
2 '
100 vl=0: OPENOUT "DUMMY": CLOSEOUT
110 startup=PEEK(&BE76)*256+PEEK(&BE75)
120 disk=PEEK(startup+6)
130 btb=startup+&190+disk*23
140 mx=PEEK(startup+&4AC+disk*25)*4
150 mx=mx+560
160 FOR n=0 TO 21
170 wrt=PEEK(btb+n)
180 anz=(wrt=255)*-8
190 IF (anz) OR (wrt=0) GOTO 230
200 FOR m=0 TO 7
210 IF (wrt AND (2^m)) THEN anz=anz+1
220 NEXT
230 vl=vl+anz
240 NEXT
250 vl=vl*4: fr=mx-vl+4
260 PRINT "Freiplatz = "; fr; " KB"
```

PARA.COM mit vortex-Einzellaufwerk

Die Laufwerke von vortex sind bei CPC-Besitzern sehr beliebt. Dies liegt vielleicht auch daran, daß das Diskettenformat mit dem vortex-Programm "PARA" auf einfache Art editiert werden kann und man dadurch die Möglichkeit erhält, anstehenden Datenaustausch mit anderen CP/M-Computern via Diskette zu betreiben.

Leider sind die Besitzer der Einfachlaufwerke in dieser Beziehung etwas benachteiligt. Sie können zwar Disketten lesen, aber dann ist auch schon Schluß. Stellt sich die Frage: Wie speichere ich die eingelesenen Daten ab und wie beschreibe ich die Diskette? Ganz einfach! Zuerst erstellt man einmal eine Systemdiskette für "PARA", formatiert die Diskette neu und überträgt mit MOVCPM und SYSGEN (siehe PARA-Handbuch) ein 38 K-CP/M auf die Systemspuren. Dann kopiert man mit FILECOPY die Files PARA.COM bzw. PARA20.COM, SYS.DAT, DDT.COM und CASCOPY.COM und startet den Debugger.

```
A>DDT<ENTER>
DDT VERS 2.2
-ICASCOPY.COM<ENTER>
-R5F00<ENTER>
NEXT PC
6600 0100
-GO<ENTER>
A>
```

Damit wird CASCOPY an die Adresse &6000-&6600 geladen, wo es von "PARA" nicht zerstört werden kann. Anschließend wird wieder zu CP/M zurückgekehrt. Danach startet man PARA.COM und stellt das gewünschte Format auf Laufwerk A: ein, steckt die Fremddiskette ins Laufwerk und verläßt "PARA" mit Option 7 des Hauptmenüs (zurück ins CP/M mit augenblicklichen Formaten). Nun wird der RAM-Speicher, in dem sich neben "PARA" auch noch das vorher mit DDT geladene CASCOPY befindet, mit

```
A>SAVE 102 CAS.COM<ENTER>
```

auf der Fremdformat-Diskette gesichert. Diese Diskette bekommt nun der Bekannte und kann sie mit seinem Computer lesen und beschreiben. Dieser Freund muß nun folgende Operationen vornehmen:

```
A>DDT CAS.COM<ENTER>
DDT VERS 2.2
NEXT PC
6600 0100
-M600,65FF,0100<ENTER>
-GO<ENTER>
A>SAVE 6 CASCOPY.COM<ENTER>
A>
```

Dadurch wird CASCOPY.COM wieder an seinen ursprünglichen Platz verschoben und lauffähig abgespeichert. Das File CAS.COM kann dann gelöscht werden. Sobald man diese Diskette wieder in Händen hält, ist man in der Lage, Files von der Fremdformat-Diskette auf Cassette und von Cassette auf die Fremdformat-Diskette zu überspielen.

Markus Melchinger

Augen auf beim Computerkauf

Die NEUEN Schneider PCs jetzt ab Lager lieferbar.

PC 1512, 1 Laufwerk, Monochrom-Bildschirm 1999,- PC 1512, 1 Laufwerk, Color-Monitor 2499,-
 PC 1512, 2 Laufwerke, Monochrom-Bildschirm 2499,- PC 1512, 2 Laufwerke, Color-Monitor 2999,-

Hardware CPC: Liste, Stand Oktober 1986

Schneider CPC 464 Keyboard	solange Vorrat	498,-	vortex A-1 S 5.25 Zoll Aufrüstkit F-1 S zur F-1 D	498,-
Schneider CPC 664 Keyboard	solange Vorrat	798,-	vortex A-1 Z 5.25 Zoll Aufrüstkit F-1 Z zur F-1 S	548,-
Schneider CPC 6128 Keyboard (mit Floppy)		848,-	NEU vortex M-1 S 3.5 Zoll Einfachstation 708 KB mit Contr.	998,-
Schneider Grünmonitor GT-65 (für alle CPC)	Sonderpreis	278,-	NEU vortex M-1 D 3.5 Zoll Doppelstation 1.4 MB mit Contr.	1498,-
Schneider Farbmonitor CTM-644 (für alle CPC)		798,-	NEU vortex M-1 X 3.5 Zoll Zweitlaufwerk	758,-
Schneider CPC 464 mit Grünmonitor	solange Vorrat	598,-	NEU vortex F-1 X 5.25 Zoll Zweitlaufwerk	758,-
Schneider CPC 464 mit Farbmonitor CTM 664	solange Vorrat	1198,-	NEU vortex M-1 XRS 3.5 Zoll Zweitlaufwerk mit RS 232	858,-
Schneider CPC 664 mit Grünmonitor	solange Vorrat	948,-	NEU vortex F-2 XRS 5.25 Zoll Zweitlaufwerk mit RS 232	858,-
Schneider CPC 664 mit Farbmonitor	solange Vorrat	1498,-	NEU vortex M-1 Z 3.5 Zoll Einfachstation ohne Controller	628,-
Schneider CPC 6128 mit Grünmonitor		948,-	NEU vortex A-1 S 3.5 Zoll Aufrüstkit M-1 S zur M-1 D	498,-
Schneider 3 Zoll Floppy-Disk, DDI-1	solange Vorrat	548,-	NEU vortex A-1 Z 3.5 Zoll Aufrüstkit M-1 Z zur M-1 D	548,-
Schneider 3 Zoll Floppy-Disk, 2 Laufwerk FD-1	Sonderpreis	558,-	NEU vortex WD-10 3.5 Zoll Winchesterstation 10 MB	2689,-
vortex 5.25 Zoll Floppy-Disk, F-1 S		998,-	NEU vortex WD-20 3.5 Zoll Winchesterstation 20 MB	3299,-
vortex 5.25 Zoll Floppy-Disk, F-1 D		1498,-	Cumana 3 Zoll Zweitlaufwerk ohne Contr.	428,-
vortex 5.25 Zoll Floppy-Disk, F-1 Z		698,-	Cumana 5.25 Zoll Zweitlaufwerk ohne Controller	598,-

Achtung: Bitte geben Sie uns unbedingt Ihren Computertyp an, Sie ersparen sich und uns unnötige Rückfragen!

Wichtiges Zubehör für Ihren CPC

Datenrecorder Sony mit eingebautem Netzteil CPC 664/6128	98,-	Staubschutzhäuben aus welichem Kunstleder, (Schneidergrau)	
3-Zoll-Disketten Panasonic/Maxell CF 2 5 Stück	55,-	- 1 Keyboard 464/664/6128, NLO 401, DDI-1, Monitor Grün/Farbe	17.95
ab 10 Stück je 10.50	10,-	Schutzhäuben für vortex F-1 S/F-1 D, F-1 X/M-1 X	19.95
3.5-Zoll-Disketten Fuji 1 DD 10 Stück	79.80	- Drucker Panasonic 1080/90/92, DMP 2000	19.95
3.5-Zoll-Disketten Fuji 2 DD 135 tpi 10 Stück	89.80	Schutzhaube Rauchglas für Konsole CPC 464/664/6128	24.95
5.25-Zoll-Disketten OS/DD 96 tpi, empfohlen für die vortex-Laufwerke, 10 Stück in PVC-Hartbox	69.95	RS-232-C, serielle Schnittstelle CPC 464/664	148,-
Netzteil MP-1 für den Schneider CPC 464	139,-	Akustikkoppler Dataphon S 21 d	249,-
Netzteil MP-2 für die Schneider CPC 664/6128	159,-	Grafpad II für CPC 464/664, 1280x1024 Bildpunkte, DIN A4	249,-
vortex VHF-Modulator für gestochene und scharfe Bilder	298,-	- für CPC 464/664	278,-
Bei RAM-Erweiterungen von vortex erhebliche Preissenkungen, 1a-Qualität, RAM-Erweiterung SP-256	298,-	- für CPC 6128	298,-
RAM-Erweiterung SP-512	398,-	AMX-Maus, mit Software und deutschem Handbuch	278,-
bits immer Computertyp angeben		Formularaktor zu Drucker NLO 401	79.95
RAM-Erweiterungssatz um 256 KByte	98,-	Joystick, Quickshot II mit Autofire	19.95
RAM-Erweiterung für Joyce auf 512 KByte	148,-	Competition Pro 5000 mit Mikroswitcher	49.95
Bildschirmfilter für GT 64/65	58,-	Diskettenbox für 40 St. 3- bzw. 3.5-Disketten, 1a-Qualität	39.95
Bildschirmfilter für Farbmonitor CTM 644	68,-	wie oben, jedoch für 40 St. 5.25-Disketten	49.95
Monitor-Drehfuß, Neigungswinkel stufenlos einstellbar	39.95	Diskettenbox für 100 St. 5.25-Disketten mit Schloß	34.95
Verlängerungskabel, 1.5 Meter, für CPC 464	29.95		
dito für CPC 664/6128	34.95		

Die Software-Wende...

Star-Writer I, V 2.0	198,-	RH-Disco	59,-	Turbo-Lader Science	169,-
Creator-Star	49.90	Einsteigerpaket	50.-/70.-	Turbo-Graphix Toolbox	225,-
Fibu-Star-Plus	298,-	BYTE ME		Pascal/MT +	174,-
Dateli-Star	88,-	Finanzbuchhaltung	278,-	C-Basic-Compiler 80	174,-
Disksort-Star	49.90	Auftragsbearbeitung 5.0	398,-	Small-C-Entwickl.-Syst.	148,-
Mathe-Star	69.90/98,-	Verkehrsverwaltung	198,-	Multiplan V. 1.06	199,-
Statistic-Star	59.90/79.90	Adreßverwaltung	98,-	WordStar V. 3.0	199,-
Composer-Star	98,-	Demoprogramme auf Anforderung, je Programm 95,- DM, wird bei Rückgabe des Programms wieder erstattet (Bew. gutgeschrieben, (betrifft nur BYTE ME))		dBase II V. 2.41	199,-
Star-Mon	59.90/79.90	Supercopy - das Kopierprogramm der Superlative,		DR GRAPH	199,-
Star u. d. Zaubertab	39.90	CPC 464/664/6128,		DR DRAW	199,-
Star-Games I	39.90	100% Maschinencode, nur 79,-			
Designer-Star	39.90	Spitzenprogramme im Vertrieb von Markt&Technik			
Copy-Star II	29.90/39.90	Turbo-Pascal + Grafik	285,-		
Puzzle	39.90	Turbo-Pascal ohne Grafik	225,-		
Pyramide	39.90	Turbo-Toolbox	225,-		
		Turbo-Tutor dt.	104,-		
		Turbo-Tutor engl.	104,-		
		Turbo-Lader-Grundpak.	138,-		
		Turbo-Lader Business	148,-		

Rund um den Joyce

Schneider PCW 8256 Joyce	1648,-	FIBU-Star Plus	298,-
Schneider PCW 8512 Joyce plus mit 2 Laufwerk 1 MB	2298,-	Joyce-Mailing-System	189,-
3-Zoll-Disketten CF 2-DD für Zweitlaufwerk Joyce	98,-	Dateli-Star	98,-
FD-2 (2. Laufwerk für Joyce 1 MB)	698,-	Business-Star	298,-
RAM-Erweiterungssatz um 256 KByte	148,-	Auftragsbearbeitung	448,-
Bildschirmfilter für Joyce/Joyce plus	89,-	Joyce für Einsteiger	199,-
Parallel- und Seriell-Schnittstelle	148,-	dBase II	298,-
Farbband für Joyce-Drucker	24.95	WordStar	199,-
Grafpad 3, professionelles CAD-System für Joyce PCW 8256 und PCW 8512, komplett mit Software und Handbuch	549,-	Multiplan	199,-
Joyce Schreib-Praxis-Paket, siehe Data Media	89,-	DR GRAPH	199,-
Mica CAD-Programm Joyce	198,-	DR DRAW	199,-
Vereinsverwaltung Joyce	248,-	Star-Mail V 2.0	98,-
Turbo-Adress Joyce	149,-	Star-Base	198,-
Turbo-Pascal ohne Grafik	225,-	Statistic-Star	98,-

Weitere Programme finden Sie in unserer Liste "Joyce", die wir Ihnen gerne zusenden, Gebühr 1.50 DM in Briefmarken.

Druckerparade

Epson LX-80, 100 Zeichen/Sekunde	898,-	Oxidata Einzelblatteinzug für ML 182/ML 192	449,-
Epson FX-85, 160 Zeichen/Sekunde	1248,-	NEC P 6, 200 Zeichen/Sekunde, absolute Spitze	1649,-
Epson FX-105, 160 Zeichen/Sekunde, Breit	1698,-	NEC P 7, 200 Zeichen/Sekunde, Breit	2198,-
Epson LQ-800, 200 Zeichen/Sekunde, 24 Nadeln	1698,-	Star NL-10, 120 Zeichen/Sekunde	898,-
Epson LQ-1000, 300 Zeichen/Sekunde, 24 Nadeln	2198,-	Star NG-10, 120 Zeichen/Sekunde	898,-
Epson HI-80, 4-Farb-Plotter	1298,-	Star SG-15, 120 Zeichen/Sekunde, Breit	1248,-
Panasonic 1080, 100 Zeichen/Sekunde	698,-	Einzelblatteinzug NL-10/NG-10	259,-
Panasonic 1091, 120 Zeichen/Sekunde	849,-	Schneider DMP 2000, Lieferzeit a. A.	678,-
Panasonic 1092, 180 Zeichen/Sekunde	1098,-	Rittermann F-1, 105 Zeichen/Sekunde	848,-
Panasonic 1592, 180 Zeichen/Sekunde, Breit	1598,-	Druckerschlüssel CPC 464/664 ditto CPC 6128, hochwertiges Rundkabel, geschirmt	49.95
Panasonic 1595, 240 Zeichen/Sekunde, Breit	1998,-	Druckerständer, 1a-Qualität, Multiform	59.95
Panasonic 3151, Typenradldrucker	1398,-	Endospapier, weiß, Mikroperforation, 300 Blatt	98,-
Okimate 20, Farbplotter	749,-	dito 1000 Blatt	19.95
Okidata ML 182, 120 Zeichen/Sekunde	798,-	Erersatzfarbbänder für alle Drucker ab Lager lieferbar.	2000 Blatt 49.95
Okidata ML 192, 160 Zeichen/Sekunde	1298,-		
Okidata ML 193, 160 Zeichen/Sekunde, Breit	1498,-		

Alle Geräte mit FTZ und deutschem Handbuch. Technische Unterlagen je Gerät gegen 1.50 DM in Briefmarken. Expressversand möglich in die Schweiz und nach Österreich, zeitliche Abweichung der Bezahlung. Besuchen Sie uns mal in Ravensburg, telefonische Anmeldung erforderlich (Veransprechend) innerhalb der BRD, per Nachnahme zusätzlich Porto.

Geschäftzeiten: Montag - Freitag von 9-12 und von 14-18 Uhr, Mittwochnachmittag geschlossen! Samstags von 8-13 Uhr, kein langer Samstag. Sie finden uns im Schuhhaus Mayer, Z. Stock.

LTE
 schauties
 electronic bauelemente

Bachstraße 52
 D-7980 Ravensburg
 Telefon 07 51 / 2 61 38

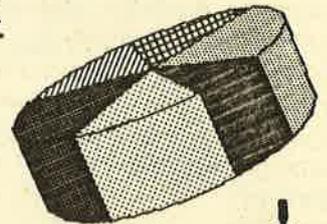
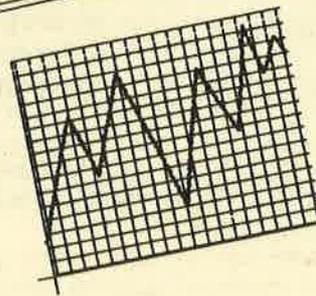


Diesmal gibt es eine ganz besondere "Anwendung des Monats": einen Forth-Compiler für die CPC-Computer 464/664 und 6128. Wer also schon immer mal mit Forth arbeiten wollte, hat jetzt die Gelegenheit dazu.

Sönke Ostertun-Gaekel aus Henstedt-Ulzburg ist der Autor dieses Topprogramms. Er ist 16 Jahre jung und geht in die 10. Klasse am örtlichen Gymnasium. Begonnen hat er die Computerei vor vier Jahren mit einem Sharp MZ 80 K, zuerst mit Basic, später dann mit Pascal und schließlich auch mit Maschinensprache. Vor zwei Jahren bekam er dann einen CPC 464, auf dem er ebenfalls in Pascal und Maschinensprache programmierte. Da sich Sönke aber auch für andere Hochsprachen interessierte, lieh er sich ein Buch über Forth aus, in dem sehr gut beschrieben war, wie die Sprache intern arbeitet. Daraufhin begann er, sich die Sprache selbst zu schreiben.

Sönke bastelt auch gern Schneider-Hardware. So hat er sich z. B. die X- und Y-Tasten ausgetauscht. Nebenbei spielt er auch leidenschaftlich Volleyball und fährt gerne Rad. Und für die 1000.- DM Honorar will er sich ein Schneider-DDI-1-Laufwerk kaufen.

Anwenderprogramm November



RBC-COMPUTERTECHNIK GmbH
Herrn
Karl Müller
Hofstraße 5
2234 Dorfstadt

SEHR GEEHRTEN HERR MÜLLER
MIR FREIEN UNS, DASS SIE VON
UNSEREM ANGEBOT GEBRAUCH MACHEN
MIR BITTEN SIE DAVON, UNS DIE
VOLLSTÄNDIGE LIEFERANSCHRIFT
SCHNELLSTMÖGLICH BEKANNZUGEBEN.
MIT FREUNDLICHEN GRÜßEN



Forth-Compiler

Anwendung des Monats: der CPC-Forth-Compiler

Wer schon immer mal mit Forth arbeiten wollte, hat jetzt die Gelegenheit dazu

464

664

6128

Das Programm wird folgendermaßen erzeugt:

1. Den MC-Generator abtippen und mit SAVE"FORTH.LDR" abspeichern.
2. Das Programm mit RUN starten. Der Maschinencode wird dann unter "FORTH.MC" abgespeichert und der Compiler gestartet.
3. Sobald das Bereitschaftszeichen (ein nach rechts zeigendes Dreieck) erscheint, muß man noch die Befehlsbibliothek abspeichern. Das geschieht mit SAVE"FORTH (ohne Anführungszeichen hinter dem Programmnamen!). Danach können Sie Ihren CPC zurücksetzen. Von Cassette kann man jetzt den Compiler jederzeit mit RUN" neu starten. Sobald das Bereitschaftszeichen erscheint, muß die Befehlsbibliothek mit LOAD"FORTH wieder geladen werden (auch hier ohne 2. Anführungszeichen). Für Diskettenbetrieb wird bei jedem Start ohne den MC-Generator zusätzlich das Listing "Starter" benötigt, da sonst auf Cassette umgeschaltet wird.

Zahleneingaben

Zahlen können zwischen -32768 und 65535 eingegeben werden. Dabei entsprechen sich die Bereiche -32768 bis -1 und 32768 bis 65535. Bei negativen Zahlen muß das Minuszeichen hinter der Zahl eingegeben werden. Die Eingabe muß immer dezimal erfolgen. Die Programme arbeiten mit 3 Stacks.

Stack 1 ist der Routinenstack, für den Programmierer allerdings unzugänglich.

Stack 2 ist der X-Stack, der für Parameterübergaben verwendet wird

Stack 3 ist der Y-Stack, der genau wie der X-Stack arbeitet.

Die Stapelzeiger zeigen immer auf das Low-Byte des letzten Eintrages. Wird eine Zahl eingetragen, so wird der Stackzeiger um 2 decremented, andererseits um 2 incremented, wenn ein Parameter vom Stack geholt wird. Es sollte jedoch darauf geachtet werden, daß die Stacks nicht über- oder unterlaufen, da ein solcher Fehler nicht angezeigt wird. Auf allen 3 Stacks ist jedoch jeweils für 512 Einträge Platz, was normalerweise ausreicht.

Speicheraufteilung

01F2 - 01F3	Zeiger auf letzten Befehl (für BYE & PRGM)
01F4 - 01F9	Systemvariable
01FA - 01FB	Programmzeiger
01FC - 01FD	Namen-Tabellenzeiger

01FE - 01FF	Tabellenzeiger für Adressen
0200 - 79FF	Programm
7A00 - 89FF	Namen-Tabelle
8A00 - 91FF	Adressen-Tabelle
9200 - 95FF	X-Stack
9600 - 99FF	Y-Stack
9A00 - 9DA1	Routinen-Stack
9DA2 - A4DF	Compiler
A4E0 - A5FF	Eingabe-Puffer
A600 - ABFF	Disk-RAM, Symboltabelle etc.

Die Befehle der Programmbibliothek "FORTH" belegen den Speicherbereich &0200 bis einschließlich &0902.

Befehle im Direktmodus

LIST	gibt alle bisher vorhandenen Befehle auf dem Bildschirm aus. Das Tempo kann durch Tastendruck reguliert werden.
#LIST	druckt die Befehlsliste.
SAVE"Name	speichert das Programm auf Cass./Disk.
LOAD"Name	lädt ein solches Programm wieder zurück. Der Name für LOAD und SAVE muß angegeben werden und darf höchstens 8 Buchstaben lang sein.
USE	ruft den USER-Modus auf (siehe dort).
CREATE"Name	erzeugt einen neuen Befehl. Die Eingabe muß direkt hinter dem Namen enden.
DECREATE"Name	löscht die Tabelleneinträge der Routine mit dem angegebenen Namen. Die Routine selbst bleibt erhalten. Der Befehl ist dann nützlich, wenn die Tabellen überzulaufen drohen. Auch hier muß die Eingabe direkt hinter dem Namen abgeschlossen werden.
PRGM	erzeugt ein fertiges Programm. Dazu muß nach Eingabe von PRGM die ENTER-Taste gedrückt werden. Dann kann man den Name eingeben. Das kann irgendein Dateiname mit dem Format filename.typ sein, wobei "filename" höchstens 8 und "typ" höchstens 3 Zeichen lang sein darf. Ein so erzeugtes Programm kann später

von Basic aus mit dem Befehl RUN"filename.typ geladen und gestartet werden. Der letzte Befehl des Hauptprogramms muß als letztes mit CREATE kreiert worden sein. Er darf nicht mit LOAD geladen worden sein (siehe Beispiel).

Befehle im CREATE-Modus

Routinenamen, die eingegeben werden, veranlassen den Compiler, einen Maschinensprache-CALL zu der Adresse der entsprechenden Routine in das Programm einzutragen. Außerdem versteht der Compiler folgende Kommandos:

PRON	schaltet den Drucker ein. Darauf folgende Eingaben im CREATE-Modus werden auf dem Drucker protokolliert.
PROFF	schaltet den Drucker wieder aus.
BYE	vergißt alle Eingaben, die beim letzten Mal im CREATE-Modus gemacht wurden und springt zurück in den Direktmodus.
OK	schließt die Routine ab, kontrolliert alle Schleifen und kehrt in den Direktmodus zurück.
REPEAT UNTIL	UNTIL erwartet ein Flag auf dem Stack. Ist dieses Flag =0, springt das Programm zu REPEAT zurück, ansonsten wird das Programm hinter UNTIL fortgesetzt.
WHILE WEND	WHILE erwartet ein Flag auf dem Stack. Ist dieses Flag =0, läuft das Programm hinter WEND, ansonsten hinter WHILE weiter. Erreicht das Programm den Befehl WEND, springt es zurück zu WHILE. Es darf also nicht vergessen werden, vor WEND ein Flag auf den Stack zu legen.
IF ELSE ENDIF	IF erwartet ein Flag auf dem Stack. Ist dieses Flag =0, läuft das Programm hinter ELSE, sonst hinter IF weiter. Trifft das Programm hinter

IF auf ELSE, springt es zu ENDIF. Der Befehl ELSE kann ausgelassen werden. In diesem Fall wird das Programm bei nichtgesetztem Flag hinter ENDIF fortgesetzt.

Wird OK, UNTIL, WEND, ELSE oder ENDIF eingegeben, kann es passieren, daß "Fehler Nr. ..." ausgegeben wird. Die Fehlernummern haben folgende Bedeutung:

- Nr. 1. Ein IF-Befehl wurde noch nicht durch ENDIF abgeschlossen.
- Nr. 2. Ein IF-ELSE-Befehl wurde noch nicht durch ENDIF abgeschlossen.
- Nr. 3. Ein REPEAT-Befehl wurde noch nicht durch UNTIL abgeschlossen.
- Nr. 5. Ein WHILE-Befehl wurde noch nicht durch WEND abgeschlossen.
- Nr. 6. Es sind schon alle Verzweigungen und Schleifen abgeschlossen.

Die USR-Befehle

	Ein	Aus	Beschreibung
*	n1 n2	n3	n3=n1*n2
+	n1 n2	n3	n3=n1+n2
-	n1 n2	n3	n3=n1-n2
/	n1 n2	n3	n3=n1/n2
<	n1 n2	f	f=-1=n1 kleiner n2, sonst 0
<=	n1 n2	f	f=-1=n1 kleiner/gleich n2, sonst 0
<>	n1 n2	f	f=-1=n1 nicht gleich n2, sonst 0
=	n1 n2	f	f=-1=n1 gleich n2, sonst 0
>	n1 n2	f	f=-1=n1 größer n2, sonst 0
>=	n1 n2	f	f=-1=n1 größer/gleich n2, sonst 0
?	n		n wird positiv ausgegeben.
?\$	n		Das Low-Byte von n wird als Zeichen ausgegeben.
?\$\$	n		Eine Zeichenkette ab der Adresse n wird ausgegeben, bis der Wert 0 gefunden wurde.
?-	n		n wird mit Vorzeichen ausgegeben.
ABS	n1	n2	n2= positiver Wert von n1.
AND	n1 n2	n3	n3=n1 and n2.
BASE	n		n legt die Basis des Zahlensystems für INPUT, ? und ?- fest.
BORDER	n1 n2		Dem Bildschirmrand werden die Farben n1 und n2 zugewiesen.
CHANGE	n1 n2	n2 n1	Die letzten beiden Stackeinträge werden vertauscht.
CLG			Grafikfenster löschen.
CLS			Textfenster löschen.
CODE	n		Der Speicherbereich ab n kann verändert werden.
DEC	n1	n2	n2=n1-1
DEEK	n1	n2	Der Zwei-Byte-Wert ab der Adresse n1 wird auf den Stack übertragen.

Wir haben die Joyce-Software, die Sie suchen!

Lohnbuchhaltung	ab DM 577.-
Hausverwaltung	ab DM 570.-
Faktura/Lager/Datei	ab DM 178.-
Vereinsverwaltung	ab DM 248.-
Adreßverwaltung	ab DM 128.-
Videoarchivverwaltung	ab DM 98.-
Datenbank	ab DM 198.-
Finanzmathematik	ab DM 95.-
Wärmebedarf	ab DM 248.-

Weitere Software auf Anfrage!

Computer-Studio 8058 Erding
Am Rätchenbach 9 · Tel. 0 81 22 / 4 05 29, zw. 14 u. 18 Uhr
Händleranfragen erwünscht!

DOKE	n1 n2	Der Zwei-Byte-Wert n2 wird ab Adresse n1 abgelegt.	NEG	n1	n2	n2=-n1
DRAW	n1 n2	Es wird eine Linie zur absoluten Position n1, n2 gezogen.	NOSIGN			Die Rechenoperationen *,/, MOD und \ werden auf positive Logik umgeschaltet. f2=-1 wenn f1=0, sonst f2=0.
DRAWR	n1 n2	Es wird eine Linie gezogen um n1 und n2, relativ zur momentanen Position des Grafikkursors.	NOT	f1	f2	f2=-1 wenn f1=0, sonst f2=0.
DROP	n	Der letzte Stackeintrag wird gelöscht.	OR	n1 n2	n3	n3=n1 or n2
DUB	n	Der oberste Stackeintrag wird verdoppelt.	OUT	n1 n2		n2 wird auf Port n1 ausgegeben.
EDIT\$	n	Der Text ab Adresse n kann editiert werden.	PAPER	n		Texthintergrundfarbe festlegen.
GPAPER	n	Die Hintergrundfarbe des Grafikfensters wird von n bestimmt.	PEEK	n1	n2	n2 entspricht dem Wert an Adr. n1.
GPEN	n	Der Farbstift für Grafikausgaben wird ausgewählt.	PEN	n		Textschreibstift auswählen.
HEIGHT	n1 n2	Die Höhe des Grafikfensters wird festgelegt: n1=oben, n2=unten.	PLOT	n1 n2		An der absoluten Position n1, n2 einen Punkt setzen.
IN	n1	n2 wird vom Port n1 gelesen.	PLOTR	n1 n2		Wie PLOT, jedoch keine absoluten, sondern relative Koordinaten.
INC	n1	n2=n1+1	POKE	n1 n2		Wert von n2 an Adr. n1 ablegen.
INK	n1 n2 n3	Dem Farbstift n1 werden die Farben n2 und n3 zugewiesen.	POS		n1	Gibt die Spalte des Textcursors zurück.
INPUT	n	Es wird eine Zahleneingabe erwartet. Bei Zahlensystemen, deren Basis größer als 10 ist, müssen die Buchstaben groß eingegeben werden. Das Minuszeichen hat vor der Zahl zu stehen.	ROT	n1 n2 n3	n2 n1 n3	Der oberste Stackeintrag wird zwischen den 3. und 4. geschoben.
INPUT\$	n	Es wird auf einen Tastendruck gewartet. Der entsprechende ASCII-Wert wird auf den Stack gelegt.	SAVE	n1 n2 n3		Ab Adresse n1 steht der Name des abzuspeichernden Files. n2 ist die Startadresse und n3 die Länge.
JOY	n	In n wird der Joystickstatus übergeben.	SCREEN	n		Bildschirmmodus setzen.
KEY\$	n	Wie INPUT\$, es wird jedoch nicht auf einen Tastendruck gewartet.	SGN	n1	n2	n2 enthält das Vorzeichen von n1.
LOAD	n1 n2	Ab Adresse n1 muß der Name des zu ladenden Files stehen. Dieses File wird an Adresse n2 geladen.	SIGN			Die Rechenfunktionen *,/, MOD und \ werden auf vorzeichenbehaftete Zahlen eingerichtet.
LOCATE	n1 n2	Der Textcursor wird an die Position n1, n2 verschoben.	SOUND	n1 n2		Das Soundregister n1 wird mit dem Wert n2 geladen.
MAX	n1 n2	n3=n2 wenn n2>n1, sonst n3=n1	SP.I	n1 n2		Die Blinkzeiten der Farbstifte werden festgelegt.
MIN	n1 n2	n3=n2 wenn n2<n1, sonst n3=n1	SP.K	n1 n2		Bei der Tastatur Wiederholung n1 und Geschwindigkeit n2 setzen.
MOD	n1 n2	n3 enthält den Rest von n1/n2.	SP.W	n1 n2		Synchronisationsbyte n1 und Baudrate n2 setzen.
MOVE	n1 n2	Der Grafikkursor wird an die absolute Position n1, n2 verschoben.	ST*EX			n2=333333/Baudrate, n1 soll zwischen 10 und 25 liegen. Je höher die Baudrate, desto kleiner n1.
MOVER	n1 n2	Der Grafikkursor wird um n1 nach rechts (minus=links) und um n2 nach oben (minus=unten) verschoben.	ST*X			Baudrate n1 n2
			ST*Y			1000 25 667
			SYS	n		2000 18 333
			TEST	n1 n2	n3	3600 10 93
						X-Stack und Y-Stack tauschen.
						X-Stack auswählen.
						Y-Stack auswählen.
						Routine an Adr. n aufrufen.
						n3 enthält die Farbstiftnummer der Grafikposition n1, n2.

TESTR	n1 n2	n3	Wie TEST, jedoch wird die Position relativ zur Grafik-cursorposition gemessen.
TIME	n1 n2		Uhrzeit setzen; n1 enthält den höheren Wert, n2 den niedrigeren.
TIME?		n2 n1	Uhrzeit abfragen; n1 enthält den höheren Wert, n2 den niedrigeren.
VPOS		n	Die Zeile, in der der Text-cursor steht, wird erfragt.
WIDTH	n1 n2		Breite des Grafikfensters festlegen; n1=links, n2=rechts.
XOR	n1 n2	n3	n3=n1 xor n2
XPOS		n	X-Position des Grafik-cursors erfragen.
YPOS		n	Y-Position des Grafik-cursors erfragen.
ZERO	n1 n2		Nullpunkt für Grafikbefehle setzen.
	n1 n2	n3 n4	n3=n1/n2 n4=n1 mod n2
@			Den USER-Modus verlassen.

Der Sound-Befehl

Er hat die Form SOUND n1 n2, wobei das Soundregister n1 mit dem Wert n2 geladen wird.

	Sound-Register	n2-Wert
0	Low-Byte Tonperiode Kanal A	(8 Bit)
1	High-Byte Tonperiode Kanal A	(4 Bit)
2	Low-Byte Tonperiode Kanal B	(8 Bit)
3	High-Byte Tonperiode Kanal B	(4 Bit)
4	Low-Byte Tonperiode Kanal C	(8 Bit)
5	High-Byte Tonperiode Kanal C	(4 Bit)
6	Rauschen, mittlere Periodendauer	(5 Bit)
7	Kontrollregister	(8 Bit)
	Bit 7: Datenrichtung Port B: 1=out, 0=in	
	Bit 6: Datenrichtung Port A: 1=out, 0=in	
	Bit 5: Rauschen Kanal C: 1=aus, 0=an	
	Bit 4: Rauschen Kanal B: 1=aus, 0=an	
	Bit 3: Rauschen Kanal A: 1=aus, 0=an	
	Bit 2: Ton Kanal C: 1=aus, 0=an	
	Bit 1: Ton Kanal B: 1=aus, 0=an	
	Bit 0: Ton Kanal A: 1=aus, 0=an	
8	Lautstärke Kanal A	
	Bit 0 bis 3: analog (ansteigend)	(4 Bit)
	Bit 4: Hardwarehüllkurve: 1=an, 0=aus	(5 Bit)
9	Lautstärke Kanal B (siehe Reg. 8)	
10	Lautstärke Kanal C (siehe Reg. 8)	
11/12	Periodendauer Hardwarehüllkurve	
11	Low-Byte	(8 Bit)
12	High-Byte	(8 Bit)
13	Hardwarehüllkurve (s. u.)	(4 Bit)
14	Datenregister Port A (siehe Reg. 7, Bit 7)	(8 Bit)
15	Datenregister Port B (siehe Reg. 7, Bit 6)	(8 Bit)
	Port B ist nicht herausgeführt.	

Hardwarehüllkurven (Reg. 13)

Bit 3: continue
 Bit 2: attack
 Bit 1: alternate
 Bit 0: hold

0 0 ? ?
 0 1 ? ?
 1 0 0 0
 1 0 0 1
 1 0 1 0
 1 0 1 1
 1 1 0 0
 1 1 0 1
 1 1 1 0
 1 1 1 1

Der CODE-Befehl

Dieser Befehl gibt den Wert der Adresse aus, auf die zugegriffen wird. Danach kann man einen neuen Wert eingeben (hexadezimal), aber erst nach <ENTER> wird der neue Wert gespeichert. Nach <ESC> wird der Wert dieser Speicherstelle nicht mehr verändert, und das Programm verläßt die CODE-Routine.

Mit dieser Routine kann man direkt Maschinencode in das Programm einfügen. Dabei muß folgendes beachtet werden:

1. Der Programmzeiger (Adresse 506 bzw. &1FA) muß anschließend hinter die neue Routine zeigen.
2. Die Routinenadresse wird in die Adreßtabelle eingetragen. Der Zeiger der Adreßtabelle steht bei Adresse 510 bzw. &1FE und muß um 2 erhöht werden.
3. Der Name wird in die Namentabelle eingetragen. Beim letzten Zeichen muß Bit 7 gesetzt sein. Der Tabellenzeiger (Adr. 508 bzw. &1FC) muß ebenfalls korrigiert werden.
4. Das IX-Register zeigt immer auf den gerade benutzten Stack, das IY-Register auf den anderen. Die Register zeigen auf das Low-Byte des letzten Eintrages.
5. Die Routine muß mit RET (&C9) abgeschlossen werden.
6. Einen Parameter vom Stack holen und in HL speichern:


```
LD L, (IX+00)
LD H, (IX+01)
INC IX
INC IX
```
7. Einen Parameter aus HL auf den Stack legen:


```
DEC IX
DEC IX
LD (IX+00), L
LD (IX+01), H
```

Noch ein paar Tips

1. Hat man mit CREATE einen Befehl erzeugt, der nicht zufriedenstellend läuft, kann man ihn folgendermaßen vollständig (nicht nur die Tabelleneinträge) löschen:


```
DECREATE "Name"
USE
```

506
498
DEEK
DOKE
@

Programmzähler
zurücksetzen

Es muß der zuletzt kreierte Befehl sein, der gelöscht werden soll.

- Der Befehlsname eines neuen Befehls darf keine Verlängerung eines schon vorhandenen Befehlsnamen sein. Es darf also kein Befehl mehr mit ?, @ oder ähnlichem anfangen. Andersherum ist es jedoch erlaubt, z. B.:
1. Befehl MOVER
2. Befehl MOVE

Das ist auch der Grund, warum die Befehle SCREEN und ZERO nicht MODE und ORIGIN heißen.

- Die Parameterangaben bei der Befehlsliste sind von links nach rechts zu lesen. Die links stehenden Parameter sind also zuerst einzugeben bzw. auszulesen. Ansonsten arbeitet der Stack nach dem LIFO-Prinzip (Last In, First Out).

Beispielprogramm

CREATE"dez-hex
REPEAT
10
base

Basis dezimal
festlegen

input
13
10
?\$
?\$
16
base
DUB
?
13
10
?\$
?\$
NOT

Zahl eingeben

Cursor
zum Beginn der nächsten Zeile
Basis hexadezimal festlegen
Stackeintrag duplizieren
Zahl ausgeben

Cursor
zum Beginn der nächsten Zeile
Flag setzen, falls Zahl=0 war
Wiederholung, falls Flag=0

UNTIL

OK

Dieses kurze Programm wandelt solange eingegebene Dezimalzahlen in Hexdezimalzahlen um, bis man 0 eingibt. Wenn Sie das Programm unter Basic ausprobieren wollen, müssen Sie noch folgendes nach der Eingabe von OK tun:

PRGM <ENTER>
DEZ-HEX.SUB <ENTER> warten
 <CTRL><SHIFT><ESC>
RUN"DEZ-HEX.SUB" <ENTER>

Sönke Ostertun-Gaekel

MC-Generator

```

1  ' *****
2  ' **** MC-Generator fuer      ****
3  ' ****                        ****
4  ' **** FORTH-COMPILER        ****
5  ' ****      (c) August 1986  ****
6  ' **** Soenke Ostertun-Gaekel ****
7  ' **** Virchowring           107 ****
8  ' **** 2359 Henstedt-Ulzburg 1 ****
9  ' **** Telefon : 04193 / 1399 ****
10 ' *****
11 '
100 MEMORY &4FFF
110 FOR i=31232 TO 37375:POKE i,0:NEXT
120 adr=&5010:z1=1010:z2=1570:GOSUB 220
130 adr=&7A00:z1=1590:z2=1680:GOSUB 220
140 adr=&8A00:z1=1700:z2=1750:GOSUB 220
150 adr=&9DA0:z1=1770:z2=2340:GOSUB 220
160 adr=&5000:z1=2360:z2=2360:GOSUB 220
170 IF PEEK(&BB4F)=&70 THEN PRINT"664-An
passung ":PRINT:GOSUB 2380
180 IF PEEK(&BB4F)=&74 THEN PRINT"6128-A
npassung":PRINT:GOSUB 2440
190 SAVE"forth.mc",b,&9DA2,&73E,&A041
200 CALL &5000
210 END
220 FOR z%=z1 TO z2 STEP 10:READ ze$,sum
%
230   FOR i%=0 TO (LEN(ze$)/2)-1
240     ax=VAL("&"+MID$(ze$,i%*2+1,2))
250     sum%=sum%-ax:POKE adr,ax:adr=adr+1
260   NEXT i%
270   IF sum%<>0 THEN PRINT"DATA-Fehler
in Zeile";z%:END
280 PRINT z%;"ok";CHR$(11)
290 NEXT
    
```

```

300 RETURN
1000 ' PRG
1010 DATA 03092E7BAAB8ADD6E00DD6601DD23DD
237EB7C8E5CD5ABBE12318F5DD7E00DD23,4008
1020 DATA DD23C35ABB0038313932DD6E00DD66
01DD23DD23E52146031147030113003600,2607
1030 DATA EDB0E1015903110A00CD9808CD3D03
020B7CB520F121460306147EC5E5CD5ABB,3149
1040 DATA E1C12310F5C9DD6E00DD6601DD23DD
237CE68028BFE53E2DCD5ABBE17C2F677D,4237
1050 DATA 2F6F2318AFCD09BDD2BDD2BDD7700
DD360100C9CD06BBDD2BDD2BDD7700DD36,3679
1060 DATA 0100C92120AB3600CD3ABD211FAB23
7EFE2138FAFE2D28F6C93E3FCD5ABBCD5A,3621
1070 DATA BCD5ABB00CD9D02E5C1CD070330EA
6F2600CD070330170B110A00CD4E0838D9,2989
1080 DATA CD070330D4856F30032428CD18E4E5
2120AB7E23B72806FE2D20F71802E1C9E1,3413
1090 DATA 110000EBB7ED527CE68028ADC90A03
FE403802D607D630FE0AC9CDBF02DD2BDD,3870
1100 DATA 2BDD7500DD7401C9CD24BBDD2BDD2B
DD7500DD7401C9DD6E00DD6601DD23DD23,3920
1110 DATA C33ABD7BC630FE3AD8C607C9000000
0000000000000000000000000000000000,1793
1120 DATA DD7E00DD23DD2332410232D3023212
03C9DD2BDD2BDD7E02DD7700DD7E03DD77,3418
1130 DATA 01C9DD23DD23C9DD6E00DD6601DD7E
02DD7700DD7E03DD7701DD7E04DD7702DD,3864
1140 DATA 7E05DD7703DD7504DD7405C9DD6E00
DD6601DD7E02DD7700DD7E03DD7701DD75,3732
1150 DATA 02DD7403C9DD7CFE96D8DDE5FDE5DD
E1FDE1C90000DD7CFE96D018EE000DD5E,5099
1160 DATA 00DD5601DD6E02DD6603DD23DD23C9
CDD803CD2408DD7500DD7401C9CDD803CD,3859
1170 DATA 260818F1CDD803CD2A0818E9CDD803
CD8B0818E1CDD803CD8B08EB18D8CDD803,3947
1180 DATA CD8B08DD2BDD2BDD7500DD7401DD73
02DD7203C9DD6E00DD6601CD4208DD7500,3705
    
```

```

1190 DATA DD7401C9DD7E00DD8B6013E0020023E
FFDD7700DD7701C9CDD8037CA2677DA3DD,3907
1200 DATA 7700DD7401C9CDD8037CB2677DB318
EFCDD8037CAA677DAB18E5CDD803CDDA08,4279
1210 DATA FFFF3E0028023EFFDD7700DD7701C9
CDD803CDDA083D28023EFF18ECCDD803CD,3976
1220 DATA DA08FE0120023EFF18DECCDD803CDDA
082FFEFE20023E0018CFCCDD803CDDA08FE,3924
1230 DATA 0120023E0018C1CDD803CDDA083D3E
FF28023E0018B2DD6E00DD66017CE680C8,3190
1240 DATA CD4208DD7500DD7401C9DD8E00DD66
01CDED08DD7700FE0120023E00DD7701C9,3494
1250 DATA CDD803CDDA08E680C8DD7EFEDD7700
DD7EFFDD7701C9CDD803CDDA08FE01CDD,5021
1260 DATA 7EFEDD7700DD7EFFDD7701C9214E08
2202042E95220A04221204221B04C9212A,2663
1270 DATA 082202042E8B18ECDD3400C0DD3401
C9DD3500C0DD3501C9DD7E00DD23DD23E6,3464
1280 DATA 03C30EBCDD7E00DD23DD23C390BBDD
7E00DD23DD23C396BBDD4E00DD4602DD23,4024
1290 DATA DD23DD23DD23C338BCDD4E00DD4602
DD7E04DD23DD23DD23DD23DD23DD23C332,3931
1300 DATA BCDD6E00DD6602C375BBDD78BBDD2B
DD2BDD7400DD360100C9CD78BB6518EEDD,4293
1310 DATA 6E00DD6601DD23DD23E9CDD803DD23
DD2373C9CDD803DD23DD23732372C9DD4E,4131
1320 DATA 02DD4603DD7E00DD23DD23DD23DD23
ED79C9DD6E00DD66015E2356DD7300DD72,3767
1330 DATA 01C9DD6E00DD66015E160018EEDD4E
00DD4601ED78DD7700DD360100C9DD7E02,3349
1340 DATA DD4E00DD23DD23DD23DD23C334BD06
041187B1DD7E00DD23121310F7C9060411,3229
1350 DATA 8AB11ADD2BDD77001B10F7C9DD6E00
DD6602DD23DD23DD23DD2322E9B4C9DD6E,4095
1360 DATA 00DD6601DD7E02DD23DD23DD23DD23
C368BCDD6E00DD6602DD23DD23DD23DD23,3864
1370 DATA C33EBCDD6E04DD660506081100C0DD
E5CD8CBCDDE13E02DD6E02DD6603DD5E00,3798
1380 DATA DD5601CDF507DDE5CD98BCCD8FBCDD
E1C900000000DD6E00DD6601DD5E02DD56,4222
1390 DATA 03C9CDAF06C3C3BBBCDAF06C3EDBBDD
AF06C3F9BBBCDAF06CDF3BBDD2BDD2BDD77,5078
1400 DATA 00DD360100C9CDAF06C3C0BBBCDAF06
C3EABBCDAF06C3F6BBBCDAF06CDF0BB18DA,4713
1410 DATA DD7E00DD23DD23C3DEBBDD7E00DD23
DD23C3E4BBBCDAF06C3C9BBBCDAF06C3CFBB,4876
1420 DATA CDAF06C3D2BBBCDC6BBDD2BDD2BDD73
00DD7201C9CDC6BBDD2BDD2BDD7500DD74,4762
1430 DATA 01C97CCD41077DF50707070707CD4A07
F1E60FFE0A3802C607C630C35ABBD630FE,3534
1440 DATA 0A3802D607C625CB25CB25CB25B56F
C96E7DCD41073E08CD5ABBCD5ABBCD06BB,3686
1450 DATA FE0DC8FEFC37C8FE3038F2FE5E3802
D620FE4730E8FE3A3804FE4138E0CD5707,4355
1460 DATA 18CFDD6E00DD6601DD23DD233E0ACD
5ABB3E0DCD5ABBCD3C073E20CD5ABBE5CD,3791
1470 DATA 6A077DE1D877CD4107237DE60720EA
18DBFD2BFD2BDD7E00DD7700DD7E01FD77,4012
1480 DATA 01DD23DD23C9DD2BDD2BFD7E00DD77
00FD7E01DD7701FD23FD23C9DD23DD23DD,4192
1490 DATA 23DD23DD23DD23C9DD6E02DD6603DD
5E00DD5601DDE5D50608CD77BC3E02E1CD,3969
1500 DATA 83BCCD7ABCDDDE1C3F90719C9B7ED52
C9CD3408CD4E08D27808C97CAA47EBCD3F,4623
1510 DATA 08EBC33F087CB7F0AF956F9C95BC67
37C0FE01C97CB728057AB737C0EBB5C87A,4432
1520 DATA B37D6B62C8FE033810378F30FD29D8
87300219D8FE8020F5C9FE01C829C97CB7,4084
1530 DATA FA8308B0FA420837C9EE80B5C07837
8FC9CD9208DA7808C9CD34087AB3C8C5EB,4507
1540 DATA 06017CB720097ABD3805652E000609
7B957A9C3805042930F63F3F78444D2100,2263
1550 DATA 003D2003181729F5781F47791F4F7B
917A983805577B915F2CF13D20E937C1C9,2995
1560 DATA 7CAA7CF2E508879FD83CC9BA20F97D
9320F5C97C879FD8B5C8AF3CC9DD7E003D,4744
1570 DATA DD77003CC0DD3501C9000000000000
00,1068
1580 ' NMT
1590 DATA 3F24A43FA43FADBF4B4559A4494E50
5554A4494E5055D44A4FD945444954A442,3300
1600 DATA 4153C54455C244524FD0524FD44348
414E47C553542AD853542AD953542A45D8,3397
1610 DATA ABADAAAFDC4D4FC44E45C74E4FD441
4EC44FD2584FD23EBD3CBD3CBEBCBE41,4363
1620 DATA 42D35347CE4D41D84D49CE6E6F7369
67EE736967EE494EC34445C35343524545,3742
1630 DATA CE434CD35045CE50415045D2424F52
4445D2494ECB4C4F434154C5504FD35650,3451
1640 DATA 4FD35359D3504F4BC5444F4BC54F55
D4504545CB444545CB49CE534F554EC454,3613
1650 DATA 494D45BF54494DC553502ECB53502E
D753502EC9534156C54C4F41C44D4F5645,3245
1660 DATA D2504C4F54D244524157D254455354
D24D4F56C5504C4FD4445241D7544553D4,3545
1670 DATA 475045CE4750415045D2434CC75A45
52CF57494454C84845494748D458504FD3,3330
1680 DATA 59504FD3434F44C55055D44745D400
00,1599
1690 ' ADT
1700 DATA 00021502600224027F028E02140322
0330035A036B037C038103A603BF03CF03,1580
1710 DATA C403E903F603FE03060417040E042E
043E04510460046A04740489049604A404,1990
1720 DATA B304C104D004E404FA041005260538
054205F608520514BC5E05680572058305,2286
1730 DATA 9B05A405B305B905C405CD05D805FC
05EC050706170637062806460658068C06,2266
1740 DATA 7D060208BC06C206C806CE06E006E6
06EC06F206FA060407DBBB0E0714071A07,2663
1750 DATA 20072E079C07CB07E0070000000000
00,696
1760 ' Hauptprogramm
1770 DATA 0000CD92BC2A319E3600CD3ABDE506
FF7E2304B720FAE1110092CD8CBC2AF201,3620
1780 DATA 22F59DED5BFA01D521E99D010E00ED
B0EB110002D5B7ED52EBE17CC1CD98BCCD,4575
1790 DATA 8FBCC32D9E00000000DD21FF95FD21
FF9931A29DC300004C4953D4234C4953D4,3567
1800 DATA 534156C54C4F41C4505247CD5553C5
4352454154C544454352454154C59A9EC2,3421
1810 DATA 9EF69E699FA29D27A189A359A431A2
9D21E0A43600CD889E7EB728F1FE202328,4191
1820 DATA F711F79D0108002BCD5F9E79B728DE
26006929011B9E094E2346C5C90000000C5,2901
1830 DATA E5D51AAEE65F20111AE680231328F3
D1E1B7ED4223444DC9D11AE6801328FA,4165
1840 DATA E10B78B120DAE1C93EF6CD5ABBCD3A
BD3E0DCD5ABB3E0AC35ABB21007A7EB7CA,4218
1850 DATA 2D9EE67FCD5ABB7E23E68028F03E0D
CD5ABB3E0ACD5ABBCD09BBDC06BB000000,3766
1860 DATA 18DB21007A7EB7CA2D9EE67FCD2BDD
7E23E68028F03E0DCD2BDD3E0ACD2BDDCD,3931
1870 DATA 09BBDC06BB00000018DB2E5052472E
544142000000002A319E7E23B7CA2D9EFE,2644
1880 DATA 2220F6E506087EB728052310F91805
36202310FB11EA9EEB0E04EDB0E1E5060C,3173

```

```

1890 DATA 110092CD8C8C3E022AFA0111FA01B7
ED52EB21FA01010000CD98BCCD8FBCE1E5,3878
1900 DATA 0108000911EE9EEB0E04EDB0E1060C
110092CD8C8C3E0221007A110018010000,2297
1910 DATA CD98BCCD8FBCC32D9E2A319E7E23B7
CA2D9EFE2220F6E506087EB728052310F9,3940
1920 DATA 180536202310FB11EA9EEB0E04EDB0
E1E5060C110092CD77BC3E0221FA0111FF,3259
1930 DATA FFC83BCCD7ABCE1E50108000911EE
9EEB0E04EDB0E1060C110092CD77BC3E02,3827
1940 DATA 21007A11FFFFCD83BCCD7ABCC32D9E
0401070E000F01160017001C0001011C01,2265
1950 DATA 18181C0214141C0306060A464F5254
482D434F4D50494C455220204350432034,1568
1960 DATA 36342F3636342F3631323820564552
53494F4E0A202020202020202020A42031,1694
1970 DATA 3938362062792053B26E6B65204F73
74657274756E2D4761656B656C0A1D0101,2600
1980 DATA 0021CF9F06727E23CD5ABB10F90000
00000000000000000000000000000000,1427
1990 DATA 000000000000000000000000000000
FF173600EDB021000222FA01267A22FC01,1808
2000 DATA 268A22FE01C32D9E3EB43600C38A9E
C5E5C129380E29380B2938080938050938,2728
2010 DATA 02C1C9C13E3FCD5ABB0C5ABB35ABB
1A13FE303FD0FE3AC93E3FCD5ABB0C5ABB,4370
2020 DATA 3E2DC922F801EBCDAFA030ECD6306F
2600CDAFA0300ECD8FA0D630856F300124,3762
2030 DATA 28D618ED1B1AFE2DC0110000EBB7ED
527CE68028C33E00C90000E52AFE011100,3336
2040 DATA 8AB7ED52CB1CCB1D444D7CB5E1C811
007ACD5F9E21000078B1C80B090901008A,3273
2050 DATA 097E23666FC9E9DD21FF95FD21FF99
2A319EE5CD88A0E17EB728F423FE40CA2D,4566
2060 DATA 9EFE2138F2FE303804FE3A38142BCD
FAA07CB52806CD26A1C32FA1CDB8A0C32F,4100
2070 DATA A12BCDC3A0FE2DCA2FA1DD2BDD7400
DD2BDD7500C32FA149C6454C53C5454E44,3990
2080 DATA 49C65245504541D4554E5449CC5748
494CC557454EC44259C54FCB50524FCE50,3468
2090 DATA 524F46C62A319EE536003E7DCD5ABB
CD3ABD30F03E0DCD5ABB3E0ACD5ABBE1C9,3907
2100 DATA 7ECD2BDD23B720F83E0DCD2BDD3E0A
CD2BBDE1C93EE532BFA1C93EC932BFA1C9,4268
2110 DATA 2AFA011103A2010B00EBEDB0EBDD2B
DD7400DD2BDD7500DD2BDD360001232322,3217
2120 DATA FA01C9DD7E00DD8B01DD23DD23CAEB
DD7E00FE01202DDD6E01DD6602DD23DD23,4000
2130 DATA DD23D5131313732372E136C323DD2B
DD7400DD2BDD7500DD2BDD360002232322,3147
2140 DATA FA01C9F52163A27ECD5ABB23B720F8
F1C630CD5ABB3E0DCD5ABB3E0ACD5ABBE1,4402
2150 DATA C32D9E4645484C4552204E52E00EB
DD7E00FE0330CD2BDD6E01DD6602DD23DD,3343
2160 DATA 23DD23D5732372E122FA01C9DD2BDD
7400DD2BDD7500DD2BDD36000322FA01C9,3710
2170 DATA EB2103A2010B00EDB0EB000000DD7E
00FE03C243A2DD7E017723DD7E027723DD,3346
2180 DATA 23DD23DD2322FA01C9DD2BDD7400DD
2BDD7500DD2BDD360004EB2103A2010B00,3224
2190 DATA EDB0DD2BDD7200DD2BDD7300DD2BDD
3600051313C354A4DD7E00FE05C243A2EB,3901
2200 DATA D5131313DD6E01DD6602732372E1DD
23DD23DD23DD7E00FE04C243A236C323DD,3717
2210 DATA 7E017723DD7E0277DD23DD23DD2323
22FA01C9E0A10EA26EA28CA2A0A2C9A2F7,4105
2220 DATA A2D7A449A3D4A1DAA1DD7E00FE06C2
43A2DD5E01DD56022AFE017323722322FE,4068

```

```

2230 DATA 012AF60122FC01C348A4010B001177
A1CD5F9E21000078B1C80B2133A309097E,2707
2240 DATA 23666FC9E52AFA01C9E52AFA0122F2
01E12A319E7EB7CA2D9E23FE2220F6ED5B,4088
2250 DATA FC017E122313B7280CE68028F53E3F
CD5ABBC32D9E2B2B1BED53F6011B7EF680,3541
2260 DATA 12DD2137A0DD3600062AFA01DD7501
DD7402CDA4A12A319E7EB728F623FE2138,3491
2270 DATA F7FE303833FE3A302F2BCDC3A0FE2D
CAD2A3545D2AFA0136212373237223EB21,3699
2280 DATA 0EA4010A00EDB0EB22FA01C3D2A3DD
2BDD7400DD2BDD75002B22F401CD6AA328,3729
2290 DATA 06CD84A3C3D2A32AF401CDFAA07CB5
2811EB2AFA0136CD237323722322FA01C3,3939
2300 DATA D2A3CDB8A0C3D2A32AFA0136C92322
FA01C32D9EED53FA01C92A319E7E23B7CA,4323
2310 DATA 2D9EFE2220F6E52AFE0111008AB7ED
52CB1CCB1D444D7CB5E1CA2D9EC5DDE111,4155
2320 DATA 007ACD5F9E78B1CA2D9EC521FE8B09
09EBD5E123230BDDE5E3B7ED4229E3C1ED,4538
2330 DATA B021000022FE93C10B21007A7E23E6
8028FA0B78B120F55D547E23E68028FA7E,3509
2340 DATA 12B72804231318F7ED53FC012AFE01
2B2B22FE01C32D9E2AF20122FA01C32D9E,3181
2350 ' Kopierroutine
2360 DATA 21105011FA01010010EDB0C32D9E00
00,1225
2370 ' 664-Anpassung
2380 POKE &9DAB,&5B:POKE &9E8E,&5B:POKE
&A1B0,&5B
2390 POKE &50B9,&5B:POKE &5151,&5B
2400 POKE &62B,&B4:POKE &62C,&B8
2410 POKE &63A,&B7:POKE &63B,&B8
2420 RETURN
2430 ' 6128-Anpassung
2440 POKE &9DAB,&5E:POKE &9E8E,&5E:POKE
&A1B0,&5E
2450 POKE &50B9,&5E:POKE &5151,&5E
2460 POKE &62B,&B4:POKE &62C,&B8
2470 POKE &63A,&B7:POKE &63B,&B8
2480 RETURN

```

Starter

```

1 'FORTH.GO Starter fuer FORTH.MC
2 '
3 '(c) Aug.1986, S. Ostertun-Gaekel
4 '
100 MEMORY &4FFF
110 LOAD"forth.mc",&9DA2
120 CALL &A041

```

Programmiersprache COMAL

Pascal zu kompliziert? Basic zu "alt"? Logo unzureichend?

Warum nicht gleich COMAL-80

Diese Sprache ist strukturiert, schnell und modern. Auch Computerneulinge können schon nach 14 Tagen fließend programmieren. COMAL-80 wird in verschiedenen Bundesländern für den Schulunterricht empfohlen und vielfach in Universitäten eingesetzt.

Jetzt auch für alle Schnöder CPC Computer

Diskette COMAL-80 Version 1.83 mit Handbuch DM 69,-
COMAL-80 Modul in Vorbereitung

COMALGRUPPE-DEUTSCHLAND

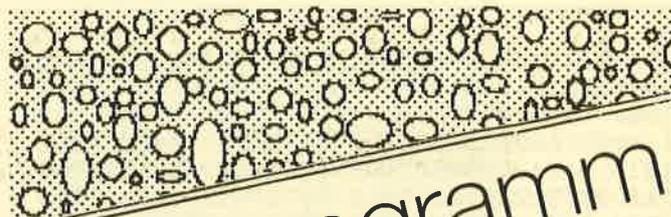
Fa. B. Bolz, 2270 Utersum/Führ, Tel.: 04883/500 Modem 554



Gerade richtig kommt das neue "Spielprogramm des Monats". Ab sofort kann jeder auf seinem heimischen CPC-Computer ein Tennismatch gegen Boris Becker wagen – und vielleicht sogar gewinnen. Das Superspiel "Tennis" macht's möglich.

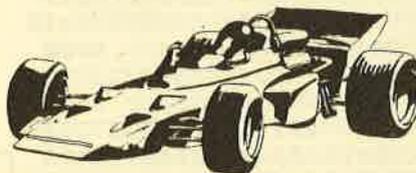
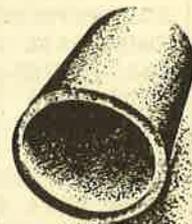
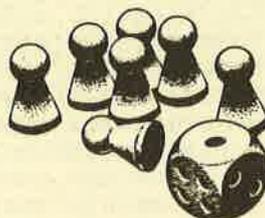
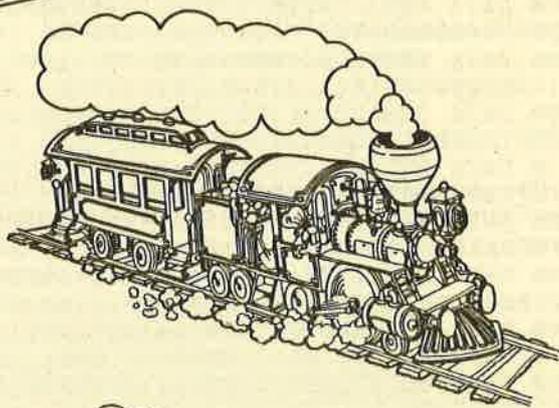
Michael Hägele aus Ravensburg hat uns dieses Programm eingeschickt. Er ist 19 Jahre jung und macht nächstes Jahr sein Abitur. Seinen CPC 464 kaufte er sich im November 1984, obwohl damals alle seine Freunde ihm wegen des großen Softwareangebots zu einem C 64 rieten. Er wollte aber einen Computer mit guten Grafikfähigkeiten und leichter Programmierung – und das war zu der Zeit der gerade erschienene CPC 464.

Seitdem hat ihn die Computerei nicht mehr losgelassen. Alle Neuigkeiten interessieren ihn brennend und jedes neue Schneider-CPC-Magazin erwartet Michael mit Ungeduld. Daneben hat er aber noch eine ganze Reihe anderer Hobbys, so z.B. Surfen, Skifahren, Tischtennis, Fußball, Fahrradfahren, Fotografieren und Gitarre spielen.



Spielprogramm

November




Tennis

Für alle Boris-Becker-Fans: Tennis, unser "Spielpro- gramm des Monats"

Wer wollte nicht schon immer einmal gegen die Großen der Tenniswelt antreten – und gewinnen?

464

664

6128

Da professionelle Programme wie z.B. "Centre Court" von Amsoft doch relativ wenig Action bieten, entschloß ich mich selbst, auch mal ein derartiges Programm zu schreiben. Es läuft auf allen CPC-Computern und wird mit Tastatur gespielt. Man kann sowohl zu zweit als auch allein gegen den Computer antreten. Dabei gelten die internationalen Tennisregeln, wobei immer derjenige aufschlägt, der den letzten Punkt gemacht hat. Das Aufspringen des Balles wird akustisch angezeigt. Zur effektiven Spielweise muß natürlich der Ball auch schräg gespielt werden können, hierzu ist der Abstand zwischen Ball und Schläger entscheidend. Die Richtung wird durch Drücken der jeweiligen Taste, die auch zum Laufen verwendet wird, während der Schlagbewegung (oben oder unten) bestimmt. Man kann natürlich auch harte, lange Bälle (Taste für vorwärts) und kurze, weiche Bälle (Taste für rückwärts) schlagen, sonst wär's ja kein richtiges Tennis!

Die Tastaturbelegung

Rechter Spieler: nach oben (Schlagball nach unten)
(Cursortasten) nach unten (Schlagball nach oben)
nach hinten (weicher Schlag)
nach vorne (harter Schlag)

Linker Spieler: w oben
x unten
a vorne
d hinten

Michael Hägele

```
D, &7E, &04, &21, &00, &00, &11, &00, &00, &CD, &C
3, &BB, &CD, 2427
2060 DATA &FC, &BB, &DD, &7E, &02, &DD, &56, &0
1, &DD, &5E, &00, &21, &F0, &FF, &CD, &C3, &BB, &C
D, &FC, &BB, 3170
2070 DATA &C9, 201
2080 IF HIMEM<>&77DF THEN SYMBOL AFTER 3
2:MEMORY &77DF
2090 dat=0:sz=0:dz=1
2100 FOR adr=30688 TO 30768
2110 READ byte:dat=dat+1
2120 sz=sz+byte
2130 POKE adr,byte
2140 IF dat<20 AND adr<30768 THEN 2180
2150 READ chksum
2160 IF chksum<>sz THEN PRINT"Fehler in
Zeile :";dz
2170 dz=dz+1:sz=0:dat=0
2180 NEXT
2190 CALL &77E0
2200 KEY 135,"pen 1:mode 2:ink 0,0:ink 1
,20"+CHR$(13)
2210 '-----
2220 'Variablendefinition
2230 '-----
2240 DIM b$(5):b$(1)=" 0":b$(2)="15":b$(
3)="30":b$(4)="40":b$(5)=" A"
2250 DEFINT a-k,m-z
2260 x=35:y=200:r=540:s=200:xx=50:yy=167
:xxv=20:yyv=0:x1=x:r1=r:s1=s:y1=y:spiel=
0:bl=1:br=1:sl=0:sr=0:sal=0:sar=0
2270 a$(1)=" fault!":a$(2)=" shoot long
er!":a$(3)=" it's out!":a$(4)=" bad!":
a$(5)=" unfortunately out!"
2280 '-----
2290 'Zeichendefinition
2300 '-----
2310 SYMBOL 230,0,8,28,56,56,17,57,125:'
linker spieler(Grundstellung)
2320 SYMBOL 231,255,244,112,120,44,100,7
0,96
2330 SYMBOL 232,0,0,0,64,160,16,32,192
2340 SYMBOL 233,0,16,56,28,28,136,156,19
0:'rechter spieler(Grundstellung)
2350 SYMBOL 234,255,47,14,30,52,38,98,6
2360 SYMBOL 235,0,0,0,2,5,8,4,3
2370 SYMBOL 237,0,48,112,112,32,248,252,
248'schlagender spieler 1
2380 SYMBOL 238,240,240,112,112,88,200,1
40,192
2390 SYMBOL 239,0,0,0,16,40,73,75,62
```

Tennis

```
1000 '---- TENNIS by -----
1010 '---- Mike Haegele -----
1020 '---- Kellenbachsteige 10 --
1030 '---- 7980 Rv-Taldorf ----
2000 '-----
2010 'RSX initialisieren
2020 '-----
2030 DATA &01, &EE, &77, &21, &EA, &77, &CD, &D
1, &BC, &C9, &FC, &A6, &EE, &77, &F3, &77, &C3, &F
A, &77, &53, 3331
2040 DATA &50, &52, &49, &4E, &D4, &00, &DD, &5
6, &0B, &DD, &5E, &0A, &DD, &66, &09, &DD, &6E, &0
8, &DD, &7E, 2186
2050 DATA &06, &CD, &C0, &BB, &CD, &FC, &BB, &D
```

```

2400 SYMBOL 240,0,12,14,14,4,31,63,31'sc
hlagender spieler r
2410 SYMBOL 241,15,15,14,14,52,19,49,3
2420 SYMBOL 242,0,0,0,8,20,146,210,124
2430 SYMBOL 243,48,40,36,28,6,3,1,0'
aufschlag1 spieler l(koerper 237,238)
2440 SYMBOL 244,12,20,36,56,96,192,128,0
aufschlag1 spieler r(koerper 240,241)
2450 SYMBOL 245,0,0,6,14,15,7,15,14'
aufschlag2 spieler l
2460 SYMBOL 246,14,30,28,24,56,104,80,12
0
2470 SYMBOL 247,24,40,72,112,128,128,0,0
2480 SYMBOL 248,0,0,96,112,240,224,240,1
12'aufschlag2 spieler r
2490 SYMBOL 249,112,120,56,24,28,32,10,3
0
2500 '-----
2510 'Titelbild aufbauen
2520 '-----
2530 SYMBOL 250,24,20,18,14,1,1,0,0
2540 MODE 1:FOR i=336 TO 240 STEP-96:ISP
RINT,40,i,230,232,231,-32
2550 :SPRINT,128,i,239,237,238,-16
2560 :SPRINT,224,i,243,237,238,-16
2570 :SPRINT,320,i,245,247,246,-32
2580 :SPRINT,416,i,230,232,231,-32
2590 :SPRINT,512,i,243,237,238,-16:NEXT
2600 PLOT 32,336:DRAWR 544,0:DRAWR 0,-12
8:DRAWR-544,0:DRAWR 0,128:MOVE 32,280:DR
AWR 544,0
2610 LOCATE 11,8:PRINT CHR$(24)"T E N
N I S"CHR$(24);
2620 FOR I=100 TO 500 STEP 100:PLOT I,33
6:DRAWR 0,-128:NEXT
2630 LOCATE 1,17:INPUT"SPIELER (1-2) ";a
nz:IF anz<1 OR anz>2 THEN 2630
2640 anz=anz-1:PEN 3:LOCATE 3,19:PRINT C
HR$(24)"NAME(5 Buch.)"CHR$(24)" linker S
PIELER";:INPUT NAM$
2650 IF ANZ=1 THEN LOCATE 3,21:PRINT CHR
$(24)"NAME(5 Buch.)"CHR$(24)" rechter SP
IELER";:INPUT nam2$ELSE nam2$="Compu"
2660 PEN 1:IF anz=0 THEN LOCATE 1,23:INP
UT"Level (0-3) ";lev:IF lev<0 OR lev>3 T
HEN 2660 ELSE lev=lev+3.5:lev1=lev/2:PEN
1
2670 INK 1,0:INK 0,9:INK 2,26:INK 3,15:B
ORDER 9
2680 PRINT CHR$(23)CHR$(0)
2690 '-----
2700 'Spielbildaufbau
2710 '-----
2720 MODE 1:WINDOW#5,1,40,25,25
2730 PLOT 20,25,2:DRAWR 600,0:DRAWR-50,2
50:DRAWR-500,0:DRAWR-50,-250
2740 PLOT 152,150:DRAWR 332,0
2750 PLOT 26,50:DRAWR 590,0:PLOT 66,250:
DRAWR 507,0
2760 PLOT 140,50:DRAWR 25,200:PLOT 500,5
0:DRAWR-25,200
2770 PLOT 322,25,1:DRAWR 0,255:PLOT 318,
25:DRAWR 0,255:PLOT 320,25:DRAWR 0,10:PL
OT 320,35,2:DRAWR 0,255
2780 ORIGIN 0,0,3,636,310,285:CLG 1:ORIG
IN 0,0,1,1,1,1:CLG 0:ORIGIN 0,0,1,640,40
0,1:PEN 1:LOCATE 2,7:PRINT" Pentax "CHR$
(24)" Michelin "CHR$(24):LOCATE 22,7:PEN
3:PRINT" Amstrad "CHR$(24):LOCATE 33,7:
PRINT" softy "CHR$(24)
2790 PLOT 142,385,1:DRAWR 355,0:DRAWR 0,
-35:DRAWR-355,0:DRAWR 0,35
2800 WINDOW#3,10,31,2,3:PAPER#3,2:CLS#3
2810 PLOT 3,311,3:FOR i=1 TO 10 STEP 2:P
LOT i,310+i:DRAWR 640-2*i,0:NEXT
2820 PAPER 2:PEN 1:LOCATE 10,2:PRINT" "n
am$:LOCATE 10,3:PRINT" "nam2$:PLOT 286,3
85,1:DRAWR 0,-35:PLOT 326,385:DRAWR 0,-3
5:PLOT 700,0,2:PAPER 2
2830 PRINT CHR$(22):PEN 2:PAPER 0:LOCATE
2,5:PRINT STRING$(38,203):PEN 1:LOCATE
2,5:PRINT STRING$(38,164):PAPER 2:PRINT
CHR$(22)CHR$(0)CHR$(23)CHR$(1)
2840 TAG:SPRINT,x1,y1,230,232,231,-32:SP
RINT,r1,s1,235,233,234,-16:FOR i=1 TO
1000:NEXT:GOSUB 3820:GOTO 4580
2850 '-----
2860 'Hauptroutine
2870 '-----
2880 IF NOT INKEY(0)THEN x=x+1:y=y+8
2890 IF NOT INKEY(2)THEN x=x-1:y=y-8
2900 IF NOT INKEY(8)THEN x=x-8
2910 IF NOT INKEY(1)THEN x=x+8
2920 IF x1=x THEN IF y1=y THEN 2940
2930 :SPRINT,x1,y1,230,232,231,-32:SPRI
NT,x,y,230,232,231,-32:x1=x:y1=y
2940 :SPRINT,r1,s1,235,233,234,-16:SPRI
NT,r,s,235,233,234,-16:r1=r:s1=s
2950 MOVE xx1,yy1:PRINT CHR$(144):MOVE
xx,yy:PRINT CHR$(144):xx1=xx:yy1=yy:xx=
xx+xxv:yy=yy+yv
2960 IF flug=1 THEN 4370
2970 IF r-xx+5<0 THEN 3430
2980 IF xx-x+5<0 THEN 3130
2990 IF anz THEN 3050
3000 IF s>1 THEN IF s>yy THEN s=s-lev
3010 IF s<330 THEN IF s<yy THEN s=s+lev
3020 o=2.9*fall:IF o+x>r THEN r=r+lev1
3030 IF o+x<=r THEN IF r>330 THEN r=r-le
v1
3040 GOTO 2880
3050 IF NOT INKEY(59)THEN r=r-1:s=s+8
3060 IF NOT INKEY(63)THEN r=r+1:s=s-8
3070 IF NOT INKEY(69)THEN r=r-8
3080 IF NOT INKEY(61)THEN r=r+8
3090 GOTO 2880
3100 '-----
3110 'Schlagabfrage Spieler links
3120 '-----
3130 IF ABS(yy-y)<16 AND ABS(yy-y)>=0 TH
EN i=REMAIN(0):yvv=0:fall=90:xxv=20:so=6
ELSE GOTO 4540
3140 :SPRINT,x1,y1,230,232,231,-32

```



SPIELER (1-2) ?

```

3150 :SPRINT,x-16,y,239,237,238,-16
3160 FOR i=1 TO 150:NEXT
3170 IF INKEY(8)=0 THEN fall=60:xxv=xxv-
2:so=0
3180 IF INKEY(1)=0 THEN fall=110:xxv=xxv
+2:so=15
3190 IF INKEY(0)=0 THEN 3220
3200 IF INKEY(2)=0 THEN 3230
3210 IF yyv=0 THEN 3250
3220 w=ABS(yy-y)/2:kw=-w:gw=-w-2:GOTO 32
40
3230 w=ABS(yy-y)/2:kw=w:gw=w+2:GOTO 3240
3240 yyv=INT(RND*(gw-kw+1)+kw)
3250 SOUND 129,0,1,15,0,0,so:SPRINT,x-1
6,y,239,237,238,-16
3260 :SPRINT,x,y,230,232,231,-32:x1=x:y1
=y
3270 flug=0:flug1=0:xx=x+16:AFTER fall,0
GOSUB 4360:GOTO 2920
3280 '-----
3290 'Punkt Spieler links
3300 '-----
3310 i=REMAIN(0):r1=r:s1=s:TAGOFF:bl=bl+
1:IF bl>=5 AND br<4 THEN sl=sl+1:bl=1:br
=1:GOSUB 4590 ELSE IF bl=5 AND br=5 THEN
br=br-1:bl=bl-1
3320 IF bl>5 AND br=4 THEN sl=sl+1:bl=1:
br=1:GOSUB 4590
3330 LOCATE 24+spiel,2:PRINT sl:LOCATE
24+spiel,3:PRINT sr;
3340 IF sl=7 OR ABS(sl-sr)=2 THEN sal=sa
l+1:spiel=spiel+2:sl=0:sr=0
3350 LOCATE 19,2:PRINT b$(bl):LOCATE 19
,3:PRINT b$(br);
3360 IF ABS(sal-sar)=2 OR spiel>=6 THEN
4200
3370 CLS#5:PAPER 0:LOCATE 13,25:PRINT na
m$:a$(fehl):PAPER 2
3380 TAG:MOVE xx1,yy1:PRINT CHR$(144);
3390 ENV 3,10,1,5,70,0,2,10,-1,5:SOUND 1
29,0,500,0,3,0,6:FOR i=1 TO 3000:NEXT:ON
aufschlag+1 GOTO 3820,4010
3400 '-----
3410 'schlagabfrage spieler rechts/Compu
ter
3420 '-----
3430 IF ABS(yy-s)<16 AND ABS(yy-s)>=0 TH
EN i=REMAIN(0):yyv=0:x2=x:xxv=-20:so=7 E
LSE GOTO 4540
3440 :SPRINT,r1,s1,235,233,234,-16
3450 :SPRINT,r+16,s,240,242,241,-32
3460 FOR i=1 TO 150:NEXT
3470 ON anz+1 GOSUB 3640,3700
3480 yyv=INT(RND*(gw-kw+1)+kw)
3490 SOUND 129,0,1,15,0,0,so:SPRINT,r+1
6,s,240,242,241,-32
3500 :SPRINT,r,s,235,233,234,-16:r1=r:s1
=s
3510 flug=0:flug1=0:xx=r-15:AFTER fall,0
GOSUB 4360:GOTO 2920
3520 '-----
3530 'Punkt Spieler rechts
3540 '-----
3550 i=REMAIN(0):r1=r:s1=s:TAGOFF:br=br+
1:IF br>=5 AND bl<4 THEN sr=sr+1:br=1:bl
=1:GOSUB 4590 ELSE IF br=5 AND bl=5 THEN
bl=bl-1:br=br-1
3560 IF br>5 AND bl=4 THEN sr=sr+1:br=1:

```

```

bl=1:GOSUB 4590
3570 LOCATE 24+spiel,2:PRINT sl:LOCATE
24+spiel,3:PRINT sr;
3580 IF sr=7 OR ABS(sl-sr)=2 THEN sar=sa
r+1:spiel=spiel+2:sl=0:sr=0
3590 LOCATE 19,2:PRINT b$(bl):LOCATE 19
,3:PRINT b$(br);
3600 IF ABS(sal-sar)=2 OR spiel>=6 THEN
4200
3610 CLS#5:PAPER 0:LOCATE 13,25:PRINT na
m$:a$(fehl):PAPER 2
3620 TAG:MOVE xx1,yy1:PRINT CHR$(144);
3630 ENV 3,10,1,5,70,0,2,10,-1,5:SOUND 1
29,0,500,0,3,0,6:FOR i=1 TO 3000:NEXT:ON
aufschlag+1 GOTO 3820,4010
3640 IF s>140 AND s<200 THEN gw=4:kw=-4:
GOTO 3680
3650 IF s>230 THEN kw=-6:gw=0
3660 IF s<115 THEN gw=6:kw=0
3670 IF s<=230 AND s>=115 THEN gw=2:kw=-
2
3680 fall=INT(RND*(29)+80):IF fall<65 TH
EN so=0 ELSE IF fall>100 THEN so=15
3690 RETURN
3700 fall=90:IF INKEY(69)=0 THEN fall=11
0:xxv=xxv-2:so=15
3710 IF INKEY(61)=0 THEN fall=60:xxv=xxv
+2:so=0
3720 IF INKEY(63)=0 THEN 3770
3730 IF INKEY(59)=0 THEN 3750
3740 IF yyv=0 THEN 3490
3750 w=ABS(yy-s)/2:kw=-w:gw=-w-2:RETURN
3760 RETURN
3770 w=ABS(yy-s)/2:kw=w:gw=w+2:RETURN
3780 RETURN
3790 '-----
3800 'Aufschlag Spieler links
3810 '-----
3820 r=600:s=150:x=30:y=200:xx=50:xxv=20
:FLUG=0:SPRINT,x1,y1,230,232,231,-32
3830 :SPRINT,r1,s1,235,233,234,-16:SPRI
NT,r,s,235,233,234,-16:r1=r:s1=s
3840 :SPRINT,x-16,y,239,237,238,-16:MOVE
x,y+16:PRINT CHR$(144);
3850 FOR i=1 TO 1000:NEXT
3860 :SPRINT,x-16,y,239,237,238,-16:MOVE
x,y+16:PRINT CHR$(144);
3870 :SPRINT,x-16,y,243,237,238,-16:MOVE
x,y+10:PRINT CHR$(144);
3880 FOR i=1 TO 200:NEXT
3890 :SPRINT,x-16,y,243,237,238,-16:MOVE
x,y+10:PRINT CHR$(144);
3900 SOUND 129,500,2,15,0,0,5
3910 :SPRINT,x,y,245,247,246,-32
3920 FOR i=1 TO 200:NEXT
3930 :SPRINT,x,y,245,247,246,-32
3940 x=x+8:x1=x:y1=y:SPRINT,x1,y1,230,2
32,231,-32
3950 fall=INT(RND*(31)+90)
3960 yyv=INT(RND*(-3))-2
3970 yy=y+10:xx1=xx:yy1=yy:MOVE xx1,yy1:
PRINT CHR$(144):xxv=ABS(xxv):CLS#5:GOTO
4580
3980 '-----
3990 'Aufschlag Spieler rechts
4000 '-----
4010 r1=r:s1=s:r=610:s=200:x=25:y=150:xx
=525:xxv=20:FLUG=0:SPRINT,r1,s1,235,233
,234,-16
4020 :SPRINT,x1,y1,230,232,231,-32:SPRI

```

```

NT, x, y, 230, 232, 231, -32: x1=x: y1=y
4030 !SPRINT, r-16, s, 240, 242, 241, -32: MOVE
r, s+16: PRINT CHR$(144);
4040 FOR i=1 TO 1000: NEXT
4050 !SPRINT, r-16, s, 240, 242, 241, -32: MOVE
r, s+16: PRINT CHR$(144);
4060 !SPRINT, r-16, s, 240, 244, 241, -32: MOVE
r, s+10: PRINT CHR$(144);
4070 FOR i=1 TO 200: NEXT
4080 !SPRINT, r-16, s, 240, 244, 241, -32: MOVE
r, s+10: PRINT CHR$(144);
4090 SOUND 129, 500, 2, 15, 0, 0, 5
4100 !SPRINT, r-32, s, 250, 248, 249, -16
4110 FOR i=1 TO 200: NEXT
4120 !SPRINT, r-32, s, 250, 248, 249, -16
4130 r=r-40: r1=r: s1=s: !SPRINT, r1, s1, 235,
233, 234, -16
4140 fall=INT(RND*(31)+80)
4150 yyv=INT(RND*(-3))
4160 xx=xx-30: yy=y+10: xx1=xx: yy1=yy: MOVE
xx1, yy1: PRINT CHR$(144); : xxv=-ABS(xxv):
CLS#5: GOTO 4580
4170 '-----
4180 'Screen Spielende
4190 '-----
4200 CALL &BB06: PRINT CHR$(23)CHR$(0): CL
S: IF sal>sar THEN winner$=nam$ ELSE winne
r$=nam2$
4210 LOCATE 1, 2: PRINT winner$ " !": PEN 3:
PRINT: PRINT CHR$(24) "You win": PRINT: PRIN
T sal: "sar; CHR$(24)
4220 LOCATE 2, 24: PRINT: PRINT "ONCE MORE (
j/n) ?": PEN 1
4230 LOCATE 15, 3: PRINT STRING$(15, CHR$(2
07)): FOR i=2 TO 10: LOCATE 15, i+2: PRINT S
TRING$(15, CHR$(143)): NEXT
4240 FOR i=1 TO 4: LOCATE 14+i, i+12: PRINT
CHR$(213); STRING$(15-(2*i), CHR$(143)); C
HR$(212): NEXT
4250 FOR i=17 TO 20: LOCATE 19, i: PRINT ST
RING$(7, CHR$(206)): NEXT
4260 FOR i=1 TO 3: LOCATE 19-i, 20+i: PRINT
CHR$(214); STRING$(5+(2*i), CHR$(143)); CH
R$(215): NEXT
4270 PRINT CHR$(23)CHR$(1): FOR i=1 TO 2:
!SPRINT, 248+(i*32), 336, 230, 232, 231, -32: N
EXT: FOR i=3 TO 4: !SPRINT, 248+(i*32), 336,
235, 233, 234, -16: NEXT
4280 LOCATE 16, 8: PRINT CHR$(24) "TENNIS-M
ASTER"
4290 PEN 4: LOCATE 18, 10: PRINT winner$: CH
R$(24)
4300 FOR i=1 TO 2: !SPRINT, 248+(i*32), 200
, 230, 232, 231, -32: NEXT: FOR i=3 TO 4: !SPRI
NT, 248+(i*32), 200, 235, 233, 234, -16: NEXT
4310 a$=LOWER$(INKEY$): IF a$<>"j" AND a$<
>"n" THEN 4310 ELSE IF a$="j" THEN GOTO 22
50
4320 END
4330 '-----
4340 '1. Aufkommen vom Ball
4350 '-----
4360 SOUND 129, 450, 1, 5, 0, 0, 5: flug=1: flug
1=0: fall=fall\2+20: AFTER fall, 0 GOSUB 44
50: RETURN
4370 flug=3: IF flug1=2 THEN 4460 ELSE fe
hl=3: IF (xx<315 AND yy<52) OR (xx<315 AND y
y>254) THEN 3310
4380 IF (xx>315 AND yy<52) OR (xx>315 AND y
y>254) THEN 3550
    
```

```

4390 IF xxv<0 AND xx>315 THEN fehl=2: GOT
O 3310
4400 IF xxv>0 AND xx<315 THEN fehl=2: GOT
O 3550
4410 GOTO 2960
4420 '-----
4430 '2. Aufkommen vom Ball
4440 '-----
4450 SOUND 129, 450, 1, 4, 0, 0, 5: flug=1: flug
1=2: RETURN
4460 fehl=1: IF xx>298 THEN 3310 ELSE 355
0
4470 fehl=5: IF (xx<315 AND yy<52) OR (xx<31
5 AND yy>254) THEN 3310
4480 IF (xx>315 AND yy<52) OR (xx>315 AND y
y>254) THEN 3550
4490 IF xx>625 THEN 3550 ELSE IF xx<15 T
HEN 3310
4500 fehl=4: IF xxv<0 THEN 3550 ELSE 3310
4510 '-----
4520 'Ball hinter Spieler
4530 '-----
4540 IF flug1=2 THEN 4500
4550 IF flug=1 THEN 4470
4560 MOVE xx1, yy1: PRINT CHR$(144); : MOVE
xx, yy: PRINT CHR$(144); : xx1=xx: yy1=yy: xx=
xx+xxv: yy=yy+yyv: FOR i=1 TO 200: NEXT
4570 GOTO 4540
4580 flug=0: flug1=0: AFTER fall, 0 GOSUB 4
360: GOTO 2920
4590 aufschlag=ABS(aufschlag-1): RETURN
    
```

WHS-SOFT Blatt: 2

WHS-SOFT 7441 AICHTAL 1 07127/5414

RAUM : SCHLAFEN RAUM-NR. 113

Projekt : DIN
 Projekt-Nummer : 4701/83
 Datum : 12.04.1986
 Sachbearbeiter : WHS-SOFT

Norm Innentemperatur	°C	20	Hauslängengröße	m	0.52
Norm Außentemperatur	°C	-10	Anzahl der Innentüren		1
Raumvolumen	m ³	54.1	Hohe über Terrain / m		4.0
Gesamt-U-Fläche	m ²	92.8	Höhenkorrektur eSA		1.0
Temp. d. nachstr. Luft	°C	15	Höhenkorrektur eSN		0.0
Abluftüberschuss	m ³	0	Höhenkorrektur eGA		1.0

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
BZ HF n	b	h/L	A		A	IN	DT	DT	nw ns	L	a	aSL					
AF ND	2	1.14	1.28	2.1	-	2.1	2.50	20	276	2	2	10.08	0.6	6.0	A		
AW ND	1	6.64	2.77	18.4		15.2	1.34	30	613								
FB	1	1.87	1.29	2.5		2.5	0.75	5	10								
DE	1	6.64	2.26	21.6		21.6	0.40	26	225								
IW	2	2.26	2.77	18.1		18.1	0.52	5	47								
IW	1	1.76	2.77	4.9		4.9	1.55	-4	-30								

*DIN 2961-1
 WÄRMEBEDARF
 DIN 4701/83
 TK-WERT
 BERECHNUNG
 FÜR FCV, JOYCE
 FCV, CPC
 + C 128
 MIT AUSF.
 HANDBUCH
 WHS HINDERER
 TECHNISCHE
 SOFTWARE
 HOHENZOLLERNSTR. 2
 7447 AICHTAL
 07127/2414*

ang. Durchlassigkeits. (a1)A	1	6.0	Normlüftungswärme	DL/W	1	276	
n.a. Durchlassigkeits. (a1)N	1	0.0	Norm Transmission	DT/W	1	1100	
Raumkennzahl	r	0.9	Krischer-D-Wert	W/m ²	1	0.40	
Lüftungs-WB frei	Q-LFL/W	1	85	antelliger Lüft. WB	DL/DT	1	0.25
Lüftungs-WB RTL	Q-RLT/W	1	0				
Mindest-Luft. WB	DL min/W	1	276	Norm-Wärmebedarf	DN/Watt	1	1376

HÄNDLERKARUSSELL

3414 Hardegsen



Woltermann - Electronic Computer - Datentechnik

Ihr Fachhändler für Schneider-Computer
 ● Gute Beratung ● qualifizierter
 Service ● Große Auswahl an
 Zubehör und Fachliteratur

3414 Hardegsen/Solling · Lange Straße 11
 Telefon 0 55 05 / 16 94

3502 Vellmar

Schneider
 COMPUTER DIVISION
mimpex
 büroelectronic

im Herkules Vellmar, Tel. 05 61 / 82 41 90

4000 Düsseldorf

Joysoft

Humboldtstr. 84
 4000 Düsseldorf
 ☎ 0211/6801403



4322 Sprockhövel

Schneider
 COMPUTER DIVISION

vortex

PiZie-Data

Autorisierter Schneider-Fachhändler

Mittelstraße 61, 4322 Sprockhövel 2, Tel. 023 39 / 71 91

5000 Köln

Joysoft

Berrenratherstr. 159
 5000 Köln 41
 ☎ 02 21 / 41 66 34



5300 Bonn

**MUSIC
 LAND** GMBH

Computer Software, Am Ringwall 4
 5300 Bonn 1, Telefon 02 28 / 66 20 19

5630 Remscheid

Peter Stamm

Software-Versand

Henkelshof 1-3
 5630 Remscheid 11
 ☎ 0 21 91 / 66 30 78

5800 Hagen

SCHNEIDER SOFT- UND HARDWARE

SFK ELEKTRO GMBH
 DELSTERNER STRASSE 23
 5800 HAGEN
 ☎ 0 23 31 / 7 26 08

7000 Stuttgart

GOSUB

Gebrauchtcomputer

Rosenbergstraße 51
 7000 Stuttgart 1
 ☎ 07 11 / 61 39 53 / 47

Wir haben und suchen alles vom 464 bis zum Joyce!

7602 Oberkirch

Schneider
 COMPUTER DIVISION

vortex
 COMPUTERSYSTEME

**RADIO
 ELEKTRO KUDERER GMBH**

Hauptstr. 79 (neben evang. Kirche) 7602 Oberkirch Tel: 0 78 02 / 41 14 Btx: 078 021 400



**Ihr Ansprechpartner
 für das
 Händlerkarussell**

Arno Weiß
 unter der Telefonnummer
 07252/85273 jederzeit
 für Sie erreichbar.

Achtung!

Anzeigenschluß für unsere Dezember-Ausgabe:
 Montag, den 3. November 1986.

Neue Leserfragen

Beantwortet von unserem Spezialisten Andreas Zallmann

Frage: Das Programm "Puzzle" läßt sich einwandfrei laden, der Bildschirm wird aufgebaut, und dann stürzt es ab. Woran kann das liegen und was kann ich dagegen tun?

Antwort: "Puzzle" läuft auf allen drei Schneider-Rechnern und ist mit hundertprozentiger Sicherheit richtig abgedruckt. Dies bestätigten mir schon viele Leser. Ich vermute, daß Sie einen Fehler beim Abtippen gemacht haben. Möglich ist beispielsweise ein falscher CALL- oder POKE-Befehl. Auf derartige Fehler hin sollten Sie Ihr Basic-Listing noch einmal untersuchen.

Frage: Wie kann man auf dem CPC in Mode 2 mehr als zwei Farben anwählen, um z.B. in Texten bestimmte Teile hervorzuheben? Ist es möglich, mehrere Bildschirmmodi gleichzeitig zu benutzen?

Antwort: Normalerweise können Sie in Mode 2 nur zwei Farben darstellen. Es gibt allerdings im Handel diverse Programme, mit deren Hilfe man den Bildschirm in zwei oder mehr Sektoren unterteilen und in diesen jeweils zwei verschiedene Farben wählen kann (bei Mode 2), so daß man 6 oder mehr zur Verfügung hat. Ferner ist es mit Hilfe dieser Programme möglich, den Bildschirm in mehrere Mode-Zonen zu unterteilen.

Frage: Ich wüßte gerne, wo ich BIOS-Quelltexte bekommen kann.

Antwort: Sie könnten die von BIOS belegten Sektoren laden und disassemblieren. Es belegt zusammen mit dem CPC die Sektoren 1-9 der Spur 1. Zum Laden könnten Sie das Diskettensystem verwenden, welches ab Ausgabe 10 des CPC-Magazins abgedruckt wird. Benötigen Sie allerdings ein dokumentiertes Listing, sollten Sie sich an Schneider oder Amstrad wenden.

Frage: Ich habe das Baudcopy-Programm aus dem CPC-Magazin 4/86 abgetippt. Das Programm funktioniert auf meinem 664 mit dem Cassettenlaufwerk einwandfrei. Im Untertitel wird auf die Kopiermöglichkeit von Diskette auf Diskette hingewiesen, die dafür notwendigen Änderungen sind aber nirgends zu finden. Können Sie mir die Umprogrammierung angeben?

Antwort: Die angebliche Kopierfähigkeit von "Baudcopy" mit Disketten ist auf einen Fehler des Layouters zurückzuführen.

Der Satz sollte richtig heißen: "Auch Diskettenlaufwerkbesitzer können das Programm dazu benutzen, um Backup-Kopien von Cassette auf Cassette zu ziehen". Es ist nicht möglich, das Programm so umzuschreiben, daß auch Disketten kopiert werden können, denn Baudcopy ist für den Betrieb mit Disketten überhaupt nicht vorgesehen. Ich verweise Sie deshalb auf das ab Ausgabe 10/86 erscheinende Diskettensystem. Dort werden u.a. auch Kopierprogramme aller Art für Disketten vorgestellt.

Frage: Ich habe das Top-Programm "Elektra Cad" aus dem CPC-Magazin 5/86 abgetippt. Das Programm läuft, nach einigen Operationen unterbricht der CPC jedoch immer mit MEMORY FULL. Was kann ich tun, um dies zu verhindern?

Antwort: Die MEMORY FULL-Meldung wird ausgegeben, wenn der Rechner keinen Speicherplatz mehr zur Verfügung hat. Überprüfen Sie zunächst einmal, ob Sie Zeile 5410 richtig abgetippt haben, weil sich dort das MEMORY-Kommando befindet. Ansonsten rate ich Ihnen, alle REM-Zeilen zu entfernen. Dies ist ohne Probleme möglich, da diese nicht durch GOTO angesprungen werden. Auch von Ihnen hinzugefügte Funktionen und Programmteile sollten Sie weglassen.

Frage: Gibt es eine Möglichkeit, Programme zu laden, ohne die Meldung PRESS PLAY ... zu erhalten?

Antwort: Um Cassettenmeldungen und das Warten auf den Tastendruck zu unterbinden, genügt es, dem Namen ein Ausrufezeichen voranzustellen, z.B.:

SAVE "!BASIC"

Beachten Sie allerdings, daß keine Anzeige mehr erfolgt und sofort geladen und gespeichert wird. Sie müßten also schon vorher PLAY, bzw. REC & PLAY gedrückt haben.

Frage: Als Besitzer eines 464 interessiert es mich, was bei der MERGE-Routine nicht in Ordnung ist. Überall liest man, daß diese fehlerhaft sein soll. Was kann ich dagegen tun, und wie lade ich Unterprogramme mit MERGE nach?

Antwort: Die MERGE-Routine im CPC 464 funktioniert in der Tat nicht richtig. Immer wenn der Charakter für END OF FILE erscheint, bedeutet dies das Ende des Files. Dies ist ja schön und gut, nur leider haben die Programmierer übersehen, daß auch in Strings oder Zeilennummern dieser Wert mal vorkommt. Dann denkt der Rechner, das File sei zu Ende, und da zu lesen versucht wird, folgt die Fehlermeldung END OF FILE. Je kürzer das File ist, desto größer ist die Wahrscheinlichkeit, daß kein "falscher" END-OF-FILE-Character im Programm vorkommt und die Routine einwandfrei arbeitet. Bei manchen Programmen kann es also durchaus funktionieren – aber eben nicht bei allen. Beim CPC 664/6128 ist dieser Fehler behoben.

Auf dem CPC 464 haben Sie zwei Möglichkeiten, dieses Problem zu lösen. Sie können das Unterprogramm mit SAVE "NAME",A als ASCII-Datei abspeichern; dann taucht dieser Fehler nicht auf. Ferner hat Data-Becker im Floppybuch eine Routine vorgestellt, die diesen Fehler behebt.

Jetzt noch einige Tips zur Benutzung der MERGE-Unterroutinen. Das Programm, das nachgeladen wird, überschreibt immer die Programmzeilen, die es selbst beinhaltet. Sollte Ihr Hauptprogramm z.B. bis 12000 gehen, könnten Ihre nachzuladende Unterroutine ab 20000 beginnen. Da nach MERGE das Programm immer wieder mit RUN gestartet wird, sollte in Zeile 1 im nachzuladenden Programm

GOTO 20000 stehen, damit Ihre Routine auch ausgeführt wird. Diese Zeile darf dann natürlich nicht vom Hauptprogramm belegt sein. Auch ist es von Vorteil, alle Unterprogramme ab 20000 in gleichen Schritten zu programmieren. Ist nämlich ein Unterprogramm in Zehnerschritten, ein anderes in Fünferschritten durchnummeriert, bleiben beim Nachladen Teile des alten Programmes stehen, was zu sehr unliebsamen Effekten führen kann. Man kann diesen Umstand natürlich auch bewußt einsetzen.

Frage: Wie sind die Befehle DEFB, DEFW u.ä. beim ASSO-Assembler (CPC-Magazin 6/86) zu ersetzen bzw. zu umgehen?

Antwort: Siehe Anleitung ASSO, CPC-Magazin 6/86. Seite 74, 1. Spalte, Mitte (Pseudobefehl DB). DEFB steht für ein Byte ablegen, DEFW für zwei Bytes ablegen.

Frage: Ich habe mir vor einiger Zeit die Cassettenversion von "Elite" gekauft. Inzwischen habe ich ein Diskettenlaufwerk. Ich möchte wissen, wie ich eine Kopie von "Elite" auf Diskette erstellen kann. "Elite" ist ein tolles Spiel, aber die langen Ladezeiten der Cassettenversion sind ziemlich lästig.

Antwort: Ich weiß nicht, zum wievielten Male ich diese Frage schon beantwortet habe, was aber auf keinen Fall ein Vorwurf sein soll. Sehr viele Personen spielen offensichtlich "Elite" so gerne und so oft, daß der Wunsch nach einer Version auf Diskette immer größer wird. Leider ist "Elite" sehr gut geschützt und so ohne weiteres nicht auf Diskette zu überspielen. Es wird wohl das Beste sein, sich die Diskettenversion, die natürlich ebenfalls gut abgesichert ist, zu kaufen.

Frage: Als Textverarbeitungsprogramm benutze ich "Tasword" (deutsche Version). Die Hilfsseite kenne ich inzwischen auswendig und brauche sie nicht mehr. Wie bekomme ich sie aus dem Programm? Dadurch könnte ich ca. 2000 Zeichen mehr verarbeiten.

Antwort: Sie werden es wohl kaum erreichen, "Tasword" so zu ändern, daß die Hilfsseite verschwindet und stattdessen 2000 Zeichen mehr verarbeitet werden können. Ersteres wäre noch das kleinere Problem. Sie müßten bei den Routinen zum Drucken dieser Hilfsseite einfach ein RET einfügen. Dazu müßten Sie sie freilich erst einmal finden, was bei einem solch umfangreichen Programm wahrlich nicht einfach ist. Noch schwieriger dürfte es sein, Tasword zu bewegen, 2000 Zeichen mehr zu akzeptieren. Das ganze Unterfangen erscheint mir leider undurchführbar.

Frage: Gibt es Public-Domain-Software für den Schneider? Wenn ja, wie kommt man an sie heran?

Antwort: Zur Erklärung für unsere Leser: Public-Domain-Software nennt sich Software, die umsonst oder gegen eine geringe Schutzgebühr erhältlich ist. Manche Hersteller bieten Standardprogramme wie Textverarbeitung oder Tabellenkalkulation als Public-Domain-Software an, um ihre Computer dadurch besser verkaufen zu können. Uns ist hier nur die SIG/M-User-Group-Software bekannt, die verschiedene Inserenten im CPC-Magazin anbieten.

Frage: Vor einiger Zeit kaufte ich mir zum CPC-Magazin 6/86 die Cassette mit den Listings. Besonders interessierte

ich mich für das Kopierprogramm von Michael Grams (S. 85). Leider läuft es auf meinem Schneider 6128 nicht richtig. Zwar werden Programme von Diskette zum Überspielen in den Speicher geladen, aber <CTRL I> zum Abspeichern auf Cassette funktioniert nicht. Für eine Hilfestellung wäre ich dankbar.

Antwort: Mit <CTRL I> stellen Sie auf TAPE.IN, d.h., es wird jetzt von Cassette eingelesen. Um auf Diskette abzuspeichern, müssen Sie die Option <CTRL O> für TAPE .OUT benutzen.

Frage: Wie kann ich ein assembliertes Maschinenprogramm abspeichern, laden und aufrufen?

Antwort: Ein fertig assembliertes Maschinenprogramm abzuspeichern, ist relativ einfach:

SAVE "Name",b,Startadresse, Länge.

Um dieses Programm dann wieder zu laden, ist folgende Befehlsfolge notwendig:

MEMORY Startadresse - 1
LOAD "Name"

Zum Aufruf des Programms ist dann nur noch der Befehl CALL Startadresse erforderlich.

Frage: Ich benutze das Programm "Paintbox". Wenn ich das damit erstellte Bild mit CLS lösche, befindet es sich aber noch im Speicher. Gibt es einen CALL, der mir das Bild wieder in den Bildschirmspeicher holt, damit es wieder sichtbar wird?

Antwort: Den CALL, der das im Speicher befindliche Bild bei "Paintbox" in den Bildschirmspeicher bringt, kenne ich auch nicht. Vielleicht gibt es hier gar keine Routine, die lediglich das Bild kopiert. Das Beste ist, Sie schreiben sich ein eigenes Maschinenprogramm. Es könnte folgendermaßen aussehen:

```
LD BC, 16384      ; 16384 Bytes sollen kopiert werden
LD DE, 49152     ; Kopierziel
LD HL, adr       ; Kopierquelle
LDIR              ; Kopieren
RET               ; und zurück
```

Die Adresse müßten Sie dann noch selbst herausfinden. Ich kann Ihnen bei dieser Frage nicht weiterhelfen, da ich das Programm "Paintbox" nicht besitze.

Bezugsquellenverzeichnis

Was hilft ein Testbericht, wenn keine Bezugsquelle angegeben ist. Deshalb hier die entsprechenden Lieferanschriften. Dies ist allerdings nur eine Auswahl, die meisten Produkte gibt es auch bei anderen Lieferfirmen.

● Iron On Coloring Kit – Joachim Wussow, Pilotstraße 17, 8500 Nürnberg ● Rembrandt – PiZie-Data, Mittelstraße 61, 4322 Sprockhövel ● Laser Genius – Peter West Records, Am Heerdter Hof 15, 4000 Düsseldorf ● Disc Demon – PiZie-Data ● Am drum – Peter West Records und Profisoft GmbH, Sutthausen Straße 50-52, 4500 Osnabrück ● Colossus Chess 4.0 – Joysoft, Humboldtstraße 84, 4000 Düsseldorf ● Disc Mate – PR8-Software-dienst, Postfach, 8702 Eisingen ● Johnny Reb II, Arnheim – Joysoft ● Rescue on Fractacius – Activision Deutschland GmbH, Postfach 76 06 80, 2000 Hamburg 76 ● Glass – Peter West Records ● Mission Omega, Split Personalities, Stainless Steel – Joysoft ● Bounder – Peter West Records.

Neuheit



Mac Lin für den Joyce

Mac Lin, das Programm, das programmiert. Schreiben Sie mit **Mac Lin** Ihre individuellen Programme in einer Sprache, die Sie kennen: Deutsch. Warum wollen Sie immer von der Stange kaufen, wenn das Erstellen eigener individueller Programme so einfach ist, dank **Mac Lin**. **Mac Lin**, der Programmgenerator kostet bei uns 475.- DM.

Unser Angebot:

Schneider Joyce mit Mac Lin 2075.- DM
Schneider Joyce Plus mit Mac Lin 2725.- DM
Händleranfragen erwünscht! Selbstverständlich können Sie alle Geräte aus unserem Lieferprogramm auch mieten.

● NEU ● NEU ● NEU ● NEU ● NEU ● NEU ●
Computer 24 Monate mieten statt kaufen. Jetzt kaufen und in 6 Monaten bezahlen!

Schneider PC 1512
monochrom/Einzelaufwerk
Mietpreis Kaufpreis
92.- DM 1999.- DM

Schneider PC 1512
monochrom/Doppelaufwerk
Schneider PC 1512
Farbmonitor/Einzelaufwerk
Mietpreis Kaufpreis
117.- DM 2499.- DM

Schneider Joyce
Mietpreis Kaufpreis
78.- DM 1625.- DM

Schneider Joyce Plus
Mietpreis Kaufpreis
107.- DM 2275.- DM

Zweitlaufwerk FD-2
für Joyce 648.- DM

RAM-Erweiterung
für Joyce 119.- DM

Aufrüstkit Joyce
komplett 728.- DM

PC-Software

Software für Architekten
Honorarabrechnung 599.- DM

Baukosten mit
Nebenkosten 599.- DM

Leistungs-
verzeichnisse 599.- DM

alle drei Pakete
zusammen 1499.- DM

Videotheken-
verwaltung 570.- DM

Fahrschulverwaltung 570.- DM

Weitere Software zu günstigen Preisen in Vorbereitung.

Ständig die neuesten Spiele und Anwenderprogramme für Schneider CPC lieferbar!

Katalog gegen Rückporto (1.50 DM in Briefmarken). Sämtliche Lieferungen erfolgen zzgl. Porto + Verpackung.

Wo ist welcher User-Club? 100 Club-Adressen für 40.- DM. M. Bakmeier, Grüner Jäger 9, 3180 Wolfsburg 22

Verk. WordStar für CPC 6128 oder Tausch gegen Multiplan. ☎ 02 08 / 87 90 73

Verkaufe verschiedene Original-Programme u. Spiele auf Cass. u. Disk. 40% unter NP. ☎ 0 89 / 6 37 08 08 ab 17 Uhr.

Verkaufe Software und Bücher für CPC 464. Liste kostenlos! ☎ 0 72 43 / 6 66 29 von 17.00 bis 20.00 Uhr

●● Aufsteiger – Systemwechsel?? ●●
Suche Hardware und Anwendersoftware rund um CPC/Joyce/PC1512 – Info 80 Pfg. ● Uwe Ganter, Haarhofstr. 5, 5100 Aachen

Schneider CPC 464. Originale auf Disk und Cassette zu verkaufen oder Tausch ab 15.00 DM – 130 Stück – Georg Heiler, Pariser Str. 177, 6750 Kaiserslautern, ☎ 06 31 / 1 46 67

Verk. Original-Spiele auf Cassette. Liste gegen Rückporto. G. Schmötsch, Chr.-Förster-Str. 32, 2000 Hamburg 20

Verkaufe Original-Software auf Disk.: They sold a Million I u. II je 30 DM, Exploding Fist + Fight. Warrior: 35 DM, alles zusammen 90 DM. Tel.: 0 88 51 / 55 28!

Verk. CPC 6128 Color m. AMX-Mouse, Creator Star, Terminal Star, Elite und weiteren Spielen für DM 1.390.- DM. ☎ 030 / 8 81 94 43

Achtung Joyce-256-User!

RAM-Erweiterung von 256 KByte auf 512 KByte. Nur 100.- DM, ausführliche Anleitung! Für jeden leicht zu realisieren. Bar oder Scheck an OPM-Bakmeier, Grüner Jäger 9, 3180 Wolfsburg 22

Verkaufe CPC 464 mit Grünmonitor, ca. 5 Monate alt, 600.- DM. ☎ 0 73 24 / 82 56

Verkaufe TAIFUN 60.- DM, TASWORD 30.- DM, TASPRT 20.- DM. Alles Originale!!! Sascha Tobler, Wedelstr. 20, 7924 Steinheim

Verkaufe CPC 6128 mit Color-Monitor für 1100.- DM, außerdem Data-Becker-Bücher zum CPC u.a. Literatur. R. Laugisch, 6701 Ruppertsberg

CPC 664 mit Farbmon. und 2. Laufwerk für 1000.- DM, günstige Software und Lit. ☎ 0 42 42 / 14 29

Verkaufe wegen Systemwechsel: Star-Writer und Datei-Star, Software-Paß noch vorhanden. Beide nur für vortex LW 5.25" geeignet. Sehr preisgünstig!!! ☎ 06 21 / 31 78 05

CPC 664, Farbe mit Acrylhaube gegen Gebot zu verkaufen. Rolf Oppermann, Heinrichstr. 27, 3300 Braunschweig

● Verkäufe folgende Originalprg.: ●
Star Writer I 150.- DM, Tank Commander (3") u. Way of the Tiger (3") 40.- DM, Saboteur! u. Wintersports 30.- DM, Spirit u. On the Run 25.- DM, Commando 20.- DM, expl. Fist 15.- DM, Chiller, Soul of a Robot, One Man and his Droid, Formula 1 Sim., Into Oblivion je 5.- DM (+ Porto). Suche deut. Handbuch von Laser Basic u. gute Original-Programme! Karsten Kammler, Heinrich-Hertz-Straße 4, 4280 Borken-Gemen

Verkaufe und tausche CPC-Spiele. Habe viele Programme zum Tauschen. Bitte meldet euch bei Jörg Pien, ☎ 052 61 / 724 53 (nach 15 Uhr)

!! Die Chance !!

Verkaufe wegen Systemwechsel: CPC 6128 mit Colour-Monitor und Zubehör (Basic-Lernkurs, Disketten, Literatur u. Spiele) NP 2450.-, VB 1800.-. * ☎ 022 74 / 21 03 *

Verkaufe CPC 464 + Colour-Monitor für 799 DM (mit vortex Sp-64 + 130 DM). Mark Brendel, ☎ 07 11 / 69 44 18

Hilfe! Wer kann mir 5 1/4" CP/M im vortex-Format ins IBM-Format konvertieren? Ebert, Mainstr. 51, 4100 Duisburg 1

WIEN: CPC 464 + Farbmon. + vortex F1D 5,25" + 192K RAMsp. + 40 Zeitschr. + 10 Bücher + viel Basic + CP/M-Softw. ☎ 02 22 / 3 54 70 04

Der JOYCE kann mehr als nur schreiben!

SUNSHINE-Software,
A.d. Schilde 14,
5270 Gummersbach,
Telefon 0 22 61 / 7 57 52



Wer tauscht The Way of the Tiger oder Yie are Kun Fu gegen das Superspiel Get Dexter (Original, keine Raubkopien)? ☎ 071 58 / 28 83

Ich gebe meine importierten superheißen 3"-Disks von Intim-Soft-International ab. 30 DM V-Scheck o. bar. Uwe Ganter, Haarhofstr. 5, 5100 Aachen

Tausche Software aller Art. Jeder, der mir Programme zuschickt, erhält kostenlos die doppelte Anzahl zurück. Genügend Tauschmaterial vorhanden. Jürgen Eppler, Im Weglanger 11, 7470 Albstadt 2

TASWORD-D + COPY + PRINT, D.3", neu, org., nur 120.-, ☎ 08 21 / 70 65 02

Außendienstreisender sucht selbige zum Austausch von Erfahrungen und Programmen zur Spesenabrechnung per Computer. A. Boebé, ☎ 0 22 35 / 14 10, Bahnhofstr. 42, 5042 Erfstadt 1

Verkaufe wegen Systemwechsel: CPC 464 mit Grünmonitor + vortex 512K-Erweiterung + vortex F1D-Laufwerk mit VDOS 2.0 sowie Programme und Bücher. 1250.- DM VHB. ☎ 0 52 23 / 6 26 28 ab 19 Uhr

Achtung DFÜ-Fans! Interesse an privatem Datentausch im Großraum Köln-Bonn? Dann ruft mal an: 0 22 35 / 14 10 no Box!

Suche Farbmonitor zum 464 unter 500.- DM. Zahle bar oder per NN. Andreas Boebé, Bahnhofstr. 42, 5042 Erfstadt 1, ☎ 0 22 35 / 14 10

Suche für CPC 6128 Programme über Maschinenbau (z.B. Statik, Festigkeitsberechnungen, Zeichnungen etc.) auf 3"- und 5 1/4"-Diskette. Andreas, ☎ 0 61 42 / 6 21 46

Programme für Schneider CPC 464/664/6128. Liste kostenlos! G. Murdfield, Hauptstr. 61, 5377 Dahlem 1

Biete Software in Basic + MCode an. Suche Tauschpartner für Programme aller Art. C. Arun, Hauptstr. 17, 6464 Althenhäu. PS.: Tausche Farb- geg. Grünmonitor

Verkaufe CPC 464 mit Farbmonitor + vortex-Floppy F1-S + vortex-SP 64 + Bücher + Zeitschriften + viele Spiele + Anwenderprogramme. Alles für 1999.- DM abzugeben bei Klaus Buttgerit, Goethestr. 2, 7450 Hechingen, ☎ 074 71 / 67 52

Tausche! Multiplan (Orig.) für CPC 6128, 3", gegen dBase. ☎ 050 42 / 8 13 72

Taifun-Basic-Compiler, 464-Cass.-Version! 60 DM! ☎ 0 23 25 / 7 02 10

LOCKSMITH-Kopierprogramm (Bericht im Juliheft, Orig.-Diskette/Handbuch 55.- DM); für Fotofans: Diaverwaltung (Diskette 30.- DM), Spiele (C): Grand Prix Rallye II und Mindshadow (zus. 45.- DM), Bezahlung gegen bar o. V-Scheck. E. Tönnis, Brinkweg 56, 4358 Haltern 5

Suche dringend Hardcopy-Routine für Grafikdrucker Oki Mikroline 192; Rechner: Schneider 6128. Schröck-Schmidt, Lutz-Kraus-Str. 22, 6114 Groß-Umstadt

*** Original-Programmpakete ***
WordStar u. dBase II, zus. 199.- DM. Star-Division: Datei-Star (neueste Version 2.4), Diskort-Star und Statist-Star zus. nur 115.- DM. Profisoft: Tasword 6128 und Backup 3 zus. nur 65.- DM. Alle 3 Pakete zusammen nur 350.- DM. ☎ 02 34 / 31 39 75

Ich verkaufe Weltmeisterschaftsprg. Wirklich alles über die FWM. Info bei ●●● Ralf Beckmann ●●● Steigerstr. 15, 4750 Unna ●●●

CPC 464 + DD-1 + Comp.-Tisch + Zubehör, nur 1100 DM, ☎ 05 41 / 70 78 57

*** Hallo CPC-Freaks ***

Suche Tauschpartner für CPC 6128. Programme aller Art auf Cass. und Disk. Holger Raitchel, Bei der Linde 22, 8672 Selb

Wir haben die Software für CPC + JOYCE!

SUNSHINE-Software,
A.d. Schilde 14,
5270 Gummersbach,
Telefon 0 22 61 / 7 57 52



*** Wegen Systemwechsel ***
Data-Becker: Maschinenspr., Hardware Erweiter., Intern, 464 Einsteig. CP/M-Trainingsbuch, dazu Chip Spezial Bd. 1+2 mit Prg. Cass zu Bd. 1, zusammen nur 125.- DM ● Schneider: Firmware-Handbuch und Devpac Ass./Diss. zus. nur 99.- DM. ☎ 02 34 / 31 39 75

Original-Software teilw. unter 10 DM. Tolle Spiele u. MIDI-Programm + Interface. Info 80-Pf.-Marke. Heinrich Hildebrandt, Eichenstr. 34, 5470 Andernach. Es lohnt sich!

Abzugeben, neuwertig, wenig gebraucht: DMP 2000, DD11, MP1, FD1, NLQ 401, CPC Int. Jg. 85, 464 Wärmebed. + K-Zahl, 464-Buchhalt. – 3"-Discs, ☎ 0 74 27 / 25 19

Suche Joyce-Software aus Eigenentwicklung mit Kopierrecht. Angebote an M. Bakmeier, Grüner Jäger 9, 3180 Wolfsburg 22

CPC-464-Tauschpartner gesucht! Biete ausreichend Software auf Disk./Cass. Liste an Andreas Votres, Dorumer Weg 44, 2850 Bremerhaven, ☎ 04 71 / 5 12 84. Suche gebrauchte 3"-Disketten!

SUPERCOPY

Das Disketten-Kopierprogramm der Superlative für alle CPCs.

100% MC, alle 43 Tracks, 1 oder 2 Laufwerke, kopiert auch von B auf B oder von B auf A.

Supercopy analysiert alle Tracks und Sektoren:

jede mögl. Sektoranzahl pro Track
Sektorgr. (126-4096 B/S), auch gemischt auf einem Track, Sektoren mit falscher DATA ADR-MARK, Sektoren mit fehlerhafter Sektor-ID, z.B. falsche Track-Nr., Kopf-Nr., Größe

SUPERCOPY erstellt von 99,9% (wir maßen uns nicht an, 100% zu sagen; man weiß nie, was noch kommt) der auf dem Markt befindl. Software ein 1:1 Sicherheits-Duplikat.

Preis: 79.- DM (Vers. per NN + 5.- VK)

Thomas Scholl

Wiflingshauser Str. 83, 7300 Esslingen
Tel: 07 11 / 36 29 83 von 18-20 Uhr
Handleranfragen erwünscht!

Equinox	29.90/45.90
Spindizzy	38.90/49.90
Fairlight	29.90/49.90
Winter Games	35.90/54.90
"V"	29.90/49.90
Yie are Kung Fu	31.90/54.90
Zoids	31.90/45.90
Bomb Jack	/39.90
Elite	59.90/69.90
Way of the Tiger	31.90/42.90
Hexenküche II	27.50/39.90

Versand gegen Nachnahme+Versandkosten.

**Kostenlosen Katalog
S8/86 anfordern!**

Soft- und Hardwareversand

Ulrike Becker · Fasanenweg 2
6690 St. Wendel 8
Hotline täglich ab 17 Uhr: 0 58 56 / 504

Holschuh - Disketten

3.5"

Disketten

Preis auf Anfrage.

10 Disketten 5,25" HC/PC in Plastikbox (transparent oder farbig). Preis auf Anfrage.

5,25" Diskettenkopien auf Anfrage.

Datencassetten

Preis auf Anfrage.



Holschuh
Tapes
Keltenstr. 67
6140 Bensheim
☎ 06251/62665

Profi-Software für Joyce PCW 8256

Adreßverwaltung, 3" Disk 79.-
600 Adressen, Etikettendruck, Serienbriefe mit LogoScript-ASCII-Texten, div. Schriftarten, superschnelles Sortieren/Suchen, da in C programmiert.

Faktura-Control

Plus, 3" Disk 139.-
Fakturieren mit Adreß-Verw., Artikeldatei und Mahnprog., Offene-Posten-Liste, 500 Adressen, 700 Artikel, 2000 Rechnungen, direktes Einblenden der gespeicherten Adressen und Artikel in Rechnung/Gutschr./Liefersch., Lagerbestandsfortschreibung beim Fakturieren, Umsatz- u. Mindestbestandskontrolle.

Baufinanzierung, 3" Disk 189.-

Für Finanzierungsberater/Makler, neues Steuerrecht.

Joyce-Schachuhr, 3" Disk 35.-

Alle Progr. mit ausf. Benutzeranl. in deutsch. Individuelle Programme zum Festpreis.

Hashagen-EDV, ☎ 0 60 73 / 6 19 93.
Eckstraße 11, 61133 Babenhausen 3

●●● Suche Tauschpartner für CPC ●●●

Habe viele Spiele und Anwendungen auf Disk./Tape). Sendet eure Listen an Marcel Wermuth, Gemeindrütistr. 4A, CH-8630 Rüti/ZH, ☎ 055 / 31 65 78

● Disk-Softw. ● 20.-: Marco Polo 1 / je 30.-: Int. Karate, Sam. Fox Strip P., Colour Star + / je 35.-: Way of the Tiger, Spindizzy, Tau Ceti, Sorcery + / je 40.-: Winter Games, Theatre Europe / 45.-: Elite deutsch / alle Progr. nur 1 x in Originalpkg. + Anl. Tel. Bestellung bei Dirk Steitz, ☎ 08 21 / 9 83 62

Verkaufe gebr. Einzelteile für Schneider 5.25"-Zweifloppy (BASF 6106) anschlussfertig an DDI-1 = 260.- DM / 512K-Speichererweiterung (vortex) für 464 = 340.- DM / Maxam im ROM (mit dt. + engl. Anleitung) für 464 = 140.- DM / 1 m Universal-Druckerkabel für 464/664/6128 = 40.- DM / 2,5 m Scartkabel für CPC an FFS = 30.- DM / Monitor-Kabelweiche für zwei Grünmonitore an CPC = 20.- DM / 40 Stück 5.25"-Wendendisketten Colour / NN = 100.- DM. Lieferung portofrei bei Vorauskasse oder per Nachnahme. Zwischenverkauf vorbehalten. Suche Tasword 6128-D zu kaufen od. Tausch. Fickinger, Finkenweg 21, 8901 Diedorf

●●● AXON ●●●

Konstruieren von 3-dimens. Objekten in 9 versch. Axonometrien (Perspektiven). Weiter Mögl.: drehen, verkl., vergr., verschieben, Projektionen auf 3 Ebenen, Ausdruckmöglichkeit und, und, und... Einfache Bedienung, erhältlich auf 3"-Diskette für CPC 464 u. 6128 bei Grunhofer G., Berliner Ring 86, 6780 Pirmasens gegen 70.- DM per Nachnahme oder Verrechnungsscheck

Hardcopy für CP/M-Programme. Ein- u. zweifache Größe als Source in PAS u. ASM. Printprog. mit allen Steuerbefehlen u. zusätzl. manueller Eingabe in PAS sowie COM-Dateien für CPC 6128 und DMP 2000. Disk. 29.- DM, V-Scheck o. NN. W. Bucher, Kapellenstr. 12, 7990 Friedrichshafen 1

* CPC 464/664/6128 - CHEMIE *

Von Chemiker entw. 4 Programmodule: Eigensch. d. Elemente; Atommodelle; Elektronenhüllen n. Bohr o. Pauling; Periodensystem; (zus. über 120 KByte) Cass. 30 DM; Disc. 40.- DM; (Typ angeben!) - bar/Scheck - Info 2.- DM (Bfmr.) an: * U. Hox, Ackerstr. 65, 4190 Kleve *

Verkaufe CPC-464-Keyboard für 350.- DM. ☎ 089 / 691 47 87

Schneider CPC 6128 mit Color Monitor neu m. Garantie VB: 1498.- DM + DMP 2000 neu für 628.- DM zu verk. ☎ 09 21 / 4 17 48 ab 18 Uhr

Tausche Programme für 664. Eibisch, Birkenstr. 16, 8068 Pfaffenhofen 1

Verk. Backgammon 15 DM, Beach Head 30 DM. Wirth, Badenweilerstr. 14 A, 7800 Freiburg

Verk. CPC 6128 Color m. AMX-Mouse, Creator Star, Terminal Star, Elite und weiteren Spielen für 1390.- DM. ☎ 030 / 881 94 43

Verkaufe CPC 464 mit Grünmonitor, ca. 5 Monate alt, 600.- DM. ☎ 073 24 / 82 56

1. Ich bleibe WORDSTAR 3.0, Original mit Handbuch (6128) für 125.- an.
2. Ich suche TAS-SPELL für 6128.
3. Ich bin Einzelkämpfer und suche Kontakt zu Gleichgesinnten. Reinhard Herzberg, Billerbecker Str. 42, 4408 Dülmen

●●● Tausche/Verkaufe ●●●

They sold a Million D 33.- ● Supergames I D 39.- ● Der Blaue Kristall D 39.- ● nur Originale ● Info: C. Denk, Werneckstr. 18, 8000 München 40 ●

Softwareerstellung und Beratung bei Software Riotte, Postfach 1673, in 6690 St. Wendel. Vorinformation unter ☎ 068 51 / 57 27. Alle Rechner (CPC und Joyce)! Festpreisgarantie!

Wir tauschen CPC-464-Software. Michael + Detlef Sander, Ulmenweg 15, 2250 Husum

Programme für Joyce zu verkaufen! Teilweise umsonst, Liste gegen 5.- DM in Briefmarken. Böttcher, Mittelstr. 4, 4925 Kalletal, ☎ 052 64 / 12 21

Augenoptiker sucht selbige zum Austausch von Anregungen, Erfahrungen oder Programmen zur allgemeinen Optik per Computer. Andreas Boebé, Bahnhofstr. 42, 5042 Erfstadt 1, ☎ 022 35 / 14 10

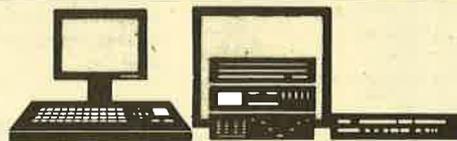
Suche Kontakt zu vortex-5 1/4"-Usern. Michael Heinrich, ☎ 079 71 / 89 57

Suche DDI-1. Jakob Kiehne, Falltorstr. 27, 5330 Königswinter 1

Biete 300 DM, suche DDI-1 ☎ 0 22 23 / 2 11 19

Stereo-Soundbox für Schneider-Computer, kompl. nur 89,90 DM. J. Moysis-zik, Kennrade 19, 5820 Gevelsberg 14

RICHTER'S



Thürmchenswall 3 · 5000 Köln 1
Telefon-Nr. 02 21 - 13 44 11

JOYCE-ADMINISTRATION

- Buchhaltung
 - Lohn und Gehalt
 - Faktura
 - Schecks
 - Kartel
 - Ausführliches Benutzerhandbuch
- komplett 798,- DM



JOYCE-GEHIRN TRUST

Expertensystem, dem Kurzzeitgedächtnis ähnlich.

- 8000 Begriffe
- 8000 Begriffserklärungen
- 144000 Begriffsverflechtungen

komplett 198,- DM

JOYCE-GEBÜHREABRECHNUNG

Steuerberatergebührenabrechnung gem. StbGebV 17.12.'81

- Rechnungsschreibung
- Adressenkartei (420 Klienten)
- Praxis-Statistik

komplett 128,- DM

Endlich!!! Aktuelle JOYCE-Software

JOYCE-Vereinsverwaltung

- Mitgliederkartei
- Mitgliederverwaltung
- Beitragsabrechnung

komplett 198,- DM

JOYCE-TURBO FAKTURA

- Rechnungsschreibung
- Programmierbarer Rechnungskopf
- Adressenkartei (700 - 1400 Adressen)
- Artikelkartei (bis 500 Artikel)

komplett 198,- DM

data berger

Drucker-Hitparade

star	NL-10	998.-
	SR-10	2100.-
	SD-10	1595.-
	NB-15	3395.-

Panasonic

KX-P 1080	698.-
1091	798.-
1092	1098.-
1592 (A3)	1398.-
1595 (A3)	1998.-
NEC P5	3499.-
P6	1699.-
P7	2229.-

Seikosha

SP-1000	799.-
BP-5420AI	5999.-

Druckerständer
für A4-Drucker **39.-**

Monitor-Ständer
für 12" Monitore
für CPCs und Joyce geeignet **39.-**

Das Diskettenablage-System der Zukunft

3" MEDIA-BOX für bis zu 100 Disketten	42.90
3.5" MEDIA-BOX für bis zu 150 Disketten	42.90
5.25" MEDIA-BOX für bis zu 70 Disketten	64.90
MEDIA CASSETTEN BOX für 16 Cassetten	20.90

Urteil der Fachpresse:

Die POSSO MEDIA BOX ist eine Diskettenbox, die ihrem Preis-, Leistungs- und Funktionsverhältnis endlich gerecht wird.

Genaue Beschreibung unserer Artikel finden Sie in unserem neuen 30seitigen Katalog mit den tollen günstigen Preisen und Beilagen vom Heim-Verlag und Markt & Technik. Versand nur gegen 2.- DM Rückporto.

**3" Disketten No Name
(Bulk) mit Shutter 79.-**

data berger

Im Lichtenfelde 76, 4790 Paderborn, RUF 05251/64852

Neues & Bekanntes für Ihren CPC

KALKUREM

Tabellenkalkulation mit Grafik

- 40*40 Felder, auch für Text verwendbar
- umfangreiche Berechnungsmöglichkeiten mit allen Möglichkeiten des CPC-Basic
- Speichern und Laden des Arbeitsfeldes
- Ausdrucken von Teilen des Arbeitsfeldes
- Löschen von beliebigen Bereichen
- Kopieren von Zeilen
- zusätzliche grafische Auswertung
- Balkengrafik, Kurvengrafik, Kuchengrafik
- Speicherung von Grafiken
- 3"-Diskette für CPC, nur 78.- DM

VOKABI

Der universelle Vokabeltrainer

Eine sinnvolle und zeitsparende Hilfe beim Erlernen von Vokabeln. VOKABI ist für fast jede Sprache einsetzbar und arbeitet mit einer besonders interessanten Lernmethode, wobei Fehler und Erfolge berücksichtigt werden.

- 3"-Disk nur 58.- DM, Cas. 48.- DM

ETATGRAF

Das Haushaltsbuch mit Grafik

Verwalten Sie Ihre Ausgaben mit dem Computer. Bis zu 18 verschiedene Kostenarten können festgelegt werden. Auswertung mit Balkengrafiken oder in Tabellenform.

- 3"-Disk nur 58.- DM, Cas. 48.- DM

TEXTKING

Die bedienungsfreundliche Textverarbeitung

- Cursor-gesteuerte Menüs
- Blocksatz, Textblockarbeiten
- Tabulator
- Floskeltasten
- wahlweise deutsche Tastaturbelegung
- Suchen und Ersetzen von Wörtern
- integrierte Diskettenverwaltung
- Schnittstelle zu ADRESCOMP-Adressendatei
- fertige Druckformulare
- umfangreiche Schriftgestaltung
- 3"-Diskette für CPC, nur 98.- DM

Bestellungen oder weitere Informationen bei

**VAN DER ZALM
SOFTWARE**

Programm-Entwicklung und Vertrieb
Elfriede van der Zalm
Schieferstätte, 2949 Wangerland 3,
Telefon 0 44 61 / 55 24

Software Hitparade

Endlich ist es soweit: Jetzt hat auch das CPC Magazin eine Software-Hitparade, die von nun an monatlich veröffentlicht wird. Wie sofort auffällt, ist die Hitparade in zwei große Blöcke unterteilt: Einerseits werden die Ergebnisse nach den Verkaufszahlen der aufgezählten Firmen ermittelt und andererseits durch Sie, die Leser.

Diese Unterteilung haben wir gewählt, da so jeder Anwender objektiv das seiner Meinung nach beste Programm auswählen kann. Häufig stimmen nämlich die hohen Verkaufszahlen mit der Qualität der Programme nicht überein, d.h., es werden manchmal auch sehr schlechte Programme in ungeheurer großer Anzahl verkauft. Mit der Software-Hitparade können Sie sich also sowohl an den Verkaufszahlen als auch an der Meinung der anderen CPC-Magazin-Leser orientieren und hoffentlich vermeiden, daß der nächste Programmkauf zu einem Fehlkauf wird.

Deshalb rufen wir hiermit alle Leser auf, uns den Namen ihres Lieblingsspiels auf einer Postkarte zuzuschicken. Zu gewinnen gibt's natürlich auch etwas: 5 Jahresabos und ein Exemplar vom Tip des Monats. Jeder User kann so mitentscheiden, wie die Leser-Top-Ten beim nächsten Mal aussehen wird.

Diese Software-Hitparade wird auch kritisch sein, indem wir die "Niete des Monats" ebenfalls berücksichtigen. Also bitte auch den größten Fehleinkauf bzw. das Programm mit unzureichender Qualität oder schlechtem Preis-/Leistungsverhältnis auf der Postkarte vermerken. Senden Sie diese dann mit Absender, Lieblingsspiel und "Niete" bis zum 30. November an folgende Adresse: CPC-Magazin, Stichwort: Leser-Top-Ten, Postfach 1640, 7518 Bretten.

Markus Pisters

Spiele-Ecke

Ob Action oder Adventure, es gibt immer wieder Programme, die einem zum Verzweifeln bringen. Sei es eine schier unüberwindliche Hürde im 3. Level, 7. Screen, oder eine knifflige Situation in einem Abenteuer, niemand ist vor solchen Problemen sicher.

Auf der anderen Seite gibt es aber auch immer wieder Freaks, die sich Tage und Nächte um die Ohren schlagen, um die tiefsten Geheimnisse eines Programms zu ergründen (manchmal hilft auch einfach der Zufall). Unsere Spiele-Ecke, ab sofort eine ständige Einrichtung, soll allen Spielern Hilfestellungen geben, also die Wissenden und die Hilfesuchenden zusammenführen. Wer Lösungshinweise für Adventure oder Unsterblichkeits-Pokes für Spiele hat, wer Tips + Tricks kennt, um Anwenderprogramme besser in den Griff zu bekommen, kann diese einschicken. Jeder Hinweis, gleich welcher Art, der anderen CPC-Benutzern hilfreich sein kann, wird gebraucht.

Spielepokes

Immer wieder erreichen uns Anfragen nach Spielepokes. Hierzu haben wir in den Heften 5/86 und 8-9/86 schon einige veröffentlicht. Wer noch unveröffentlichte Spielepokes kennt, kann sie uns einschicken. Spezielle Anfragen liegen im Moment für die Spiele Commando, Project Future, Airwolf, Finders Keepers, Ghosts'n Goblins und Fairlight vor. Wer kennt zu Pyjamarama einen anderen Poke als 16087,0?

Wer weiß mehr?

Chimera

Wer hat hierzu Tips?
Dirk Hansen, Barderup

Strangeloop

Welche Gegenstände müssen mit den Robotern getauscht werden?

Dirk Hansen, Barderup

Ghosts'n Goblins

Ich komme hier trotz Poke 20648,255 nicht über Level 2 hinaus. Wer kennt Tricks?

John Stöckmann, Bad Berneck

The Neverending Story

Wer kann mir hier den kompletten Lösungsweg zuschicken?

Jürgen Heinz
Föhrenweg 5
7332 Eisingen

Finders Keepers

Wie komme ich hier aus dem "Slimey Lower Maze" heraus?

Jörn Moser, Wilnsdorf 2

Der blaue Kristall

Wer weiß, wie ich den magischen Schlüssel von der Steintür aus finde?

Michael Wiesböck, Neubuerm

Wintergames

Ich komme hier nicht ganz klar. In der Beschreibung steht, man könne unter 18 Nationen wählen. Wie aber geht das, wenn nach dem Laden direkt das Menü erscheint? Außerdem wüßte ich gerne, wie die 8 Sportler mitspielen können und wie die Eröffnungszereemonie abläuft. Wie kann ich, wenn ich zwei Disziplinen geladen habe, die folgenden weiterladen, ohne den Computer auszuschalten, und wie komme ich zur Ehrung des Gesamtsiegers?

Peter Wienczierz, Wirges

Chiller

Wie komme ich hier durch das 5. Bild?

Axel Albrecht, Feldgeding

Spellbound

Wie komme ich hier in die unteren Stockwerke und wo finde ich Gimbal, den Zauberer?

Marcus Münch, Hamburg

Spieletips

Get Dexter alias Crafton & Xunk

Björn Winter, der an Get Dexter hängt, kann geholfen werden. Hier eine Liste mit allen Gegenständen und ihrem Verwendungszweck:

Kerzen sind weiß und stehen auf grünen Tellern. Sie verwandeln einige Robotertypen wie

z.B. die kleinen Hüpfer und die Kugel in Wasserlachen (zisch!).

Pfefferstreuer sind ebenfalls Roboterkiller. Sie nehmen sich die Vögelchen vor.

Ufos machen einen Heidenlärm, wenn man sie ablegt (durch "D"). Alles, was sich bewegt (außer Spielfigur natürlich), wird dann starr, und man kann in Ruhe arbeiten, bis das Ufo aufhört zu pfeifen.

Die grünen Flaschen erfreuen den Punk, die weißen killen Roboter, diesmal die Kühlerroste und Tonnen.

Magnete sind eine tolle Sache: Sobald nämlich ein Roboter einen berührt, hängt er für alle Zeiten dran fest.

Spritzen liegen meist an sehr unzugänglichen Stellen, was seinen Grund hat, da sie die wichtigsten Gegenstände überhaupt sind. Latscht nämlich ein Professor in eine solche, fängt er wie wild an zu zucken und verrät eine Codezahl.

Betten sind sehr gut geeignet, um Krankenschwestern einzusperren. Außerdem kann man sie als Trampolin benutzen (die Betten, nicht die Schwestern!). Dazu begibt man sich auf eines und bleibt auf dem Feuerknopf. Aber nicht zu doll, da es sonst zusammenkracht.

Mit den Büchern kann man gleichfarbige Türen öffnen. Um einen anderen Gegenstand in einen geschlossenen Raum zu bringen (z.B. die Spritze), legt man diesen vor der Tür ab, holt das Buch, öffnet die Tür, geht hindurch, legt es gleich dahinter ab ("D") und hat dann immer noch genug Zeit, wieder hinauszugehen, die Spritze zu holen und wieder hineinzugehen, bevor sich die Tür schließt.

Blumen werden von der Schwester bevorzugt. Wenn man sie ihr in den Weg stellt, sieht sie sich diese gerne an.

Schränke verdecken gelegentlich einen Ausgang und können weggezogen werden ("P"). Auf einen Schrank kommt man am besten mit einem Bett oder einer Treppe aus mehreren anderen Gegenständen.

Ventilatoren hängen an der Zimmerdecke und drehen sich. Sie haben an sich keine Bedeutung, erzeugen aber schöne Effekte, wenn man mal dranspringt (Splash!).

Xunk (wie der Kleine bei der englischen Version heißt, weiß

ich nicht) ist sehr nützlich. Er zeigt versteckte Fallen, indem er in diese reinläuft und dann zu schreien beginnt. Auch weiß er immer den kürzesten Weg zum nächsten Professor (ihm nach!!!) und hält dem Spieler gelegentlich blaue Roboter vom Hals.

Kreise auf dem Boden sind Aufzüge. Draufstellen und ab geht's!

Quadrate auf dem Boden sind Schalter, die Türen öffnen oder sonst was bewirken. Leider sind meist drei da, von denen zwei Stromfallen sind. Also erst einmal schauen, wo Xunk nicht "aia" schreit und dann daraufgestellt.

Spielautomaten sind die Dinger mit dem "H" auf dem Dach. Da kann man Energie nachtanken, wenn man sich drunterstellt.

Die Gitter von oben sind ein großes Problem in einem bestimmten Raum. Doch sind sie an bestimmten Stellen, die Xunk zeigt, zu langsam, und man kann nach vorne hüpfen. Nur Xunk folgen! (Warum der sich da auskennt, weiß nur der Programmierer.)

So, das sollte genügen, um die größten Probleme zu lösen. Noch ein Tip am Rande: Am besten ist es, man zeichnet eine Karte und schreibt auch auf, was es in den einzelnen Räumen gibt. Das erleichtert die Sache ungemein.

Jens Scholz
Eilmendinger Weg 18 a
7538 Kelttern 1

Michael Wiesböck
Findlingstraße 5 b
8201 Neubuern

Hexenküche

Sinn des Spiels ist es, die 6 im Reim genannten Zutaten zu suchen, um den "goldenen Besen" zu bekommen. Diese Zutaten befinden sich in den verschiedenen unterirdischen Höhlen. Man muß sie einsammeln, zum Hexenhaus bringen und dann den "Kürbis" besiegen, der den Besen bewacht. Die Höhlen sind in der Reihenfolge genannt, wenn man von Bild 1 aus gesehen nach rechts fliegt.

1. Höhle: goldener Besen
2. Höhle: Knochen, Fledermaus, Krug
3. Höhle: nicht betreten
4. Höhle: Truhe, Lava-Gerinnsel
5. Höhle: Kröte, Schierlingswurzel, Molchauge

Den Krug benötigt man, um an das Lava-Gerinnsel zu kommen, und ohne die Truhe kann man den Knochen nicht mitnehmen. Es ist also notwendig, mehrmals hin und her zu fliegen. Da das Spiel mehr als 100 Räume hat und das Weiterkommen gerade in den unterirdischen Höhlen sehr schwierig ist, können sich die ganz Verzweifelten auch an uns persönlich wenden. Hier jeden einzelnen Raum zu beschreiben, wäre unmöglich.

Harald Morr + Frank Hänel
Hunsrückstraße 19
5628 Heiligenhaus

Rambo II

Frank Geschke aus Bargtheide wollte in Heft 8-9/86 wissen, wo sich die Gefangenen befinden, die man nach der Hubschrauberlandung befreien soll. Hier die Lösung: Die Gefangenen sind unten links im unzerstörbaren Wall. Man braucht für die Befreiung das Messer.

Martin Wohlhöfner
Alpenweg 3a
7918 Illertissen

Schritt-für-Schritt-Adventurelösungen

Wir werden in Zukunft nur noch ab und zu komplette Adventurelösungen abdrucken, damit der Spaß am Spiel erhalten bleibt. Trotzdem: Kurztips, Fragen und Antworten oder Spielepokes wird es nach wie vor geben. Wer eine vollständige Lösung sucht, kann sich an die genannten CPC-Magazin-Leser wenden, die uns eine Komplettlösung eingeschickt haben. Jede Zuschrift sollte aber ausreichend Rückporto und einen Betrag für Fotokopien enthalten. Hier die Lösungen und die jeweiligen Einsender:

Mindshadow

Jan Andresen
Haffweg 36
3004 Isernhagen 2

Gremlins

Frank Maletz
Auerhahnhof 4
3000 Hannover 91

Heroes of Karn, The Neverending Story, Forest at World's End

Kai Haferkamp
Wilhelmstraße 131
4500 Osnabrück

Space-Mission

Aus dem Data-Becker-Buch "Adventures und wie man sie auf dem CPC programmiert".

J. Schefers
Drususallee 5
4040 Neuss 1

Jack and the Beanstalk

Hier haben wir in Heft 5/86 nach einer Lösung gefragt. Diese hat uns Alexander Knölke eingeschickt. Wer die Lösung noch sucht, kann sich an ihn wenden.

Alexander Knölke
Raabstraße 16
3000 Hannover 1

Eden Blues

Uns ist es gelungen, dazu die Lösung zu finden. Vielleicht können wir damit einigen Lesern weiterhelfen. Hier die Story in Kurzform:

Zellentrakt verlassen; im Gefängnis den Roboter ausschalten; bis zur Mitte des Hofes laufen, dann rechts bis zur Tür (Stabilität=255); Tür öffnen; nach links den Gang entlang bis zur 1. Tür; Tür öffnen; durch die verschiedenen Räume, bis man erneut auf einen Korridor stößt; rechts den Gang entlang bis zur Tür; Tür öffnen; Raum durchqueren; nächste Tür öffnen.

Hier noch einige Tips:

1. Der letzte Raum, den man durchqueren muß, verkleinert sich durch ein Absenken der Decke sehr schnell. Die Tür, die man öffnen muß, ist aber äußerst stabil (255). Uns ist dies nur gelungen, indem wir den Raum betreten, einmal vor die Tür treten und dann den Raum sofort wieder verlassen haben. Das Ganze muß man ca. fünfzehnmal wiederholen.
2. Den in manchen Räumen auf dem Boden schwimmenden Hindernissen entkommt man am besten, wenn man sich der Wand entlang an ihnen vorbeibewegt und dann zur Tür rennt.
3. Da es mit den vorhandenen Einheiten für Kraft, Mut und Leben nur schwer möglich ist, das Spiel zu beenden, hier noch einen Poke, der die Zähler für Mut, Kraft, Leben und Vorrat erhöht. Zuerst den Basic-Lader laden und danach 25 Poke 29574,255: Poke 29579,100 eingeben.
4. Die Anzahl der Leben darf 100 nicht überschreiten, da das Programm dann abbricht. Die Einheiten für Mut und Kraft dürfen nicht größer als 255 sein.

Frank Hänel + Harald Morr
Rheinlandstraße 15
5628 Heiligenhaus

Mindshadow

Hier eine vollständige Lösung:

1. Teil: Auf der Insel

Get shell, N, Go hut, Get straw, E, E, Get steel, E, Get vine, W, W, S, E, Drop shell straw steel, Tie vine to rock, D, W, Dig, Get map, Get stone, E, U, Get all, W, N, Examine map, N, N, E, N, E, E, S, S, Get rum, W, N, N, W, W, S, W, S, S, S, Bang steel with rock, Give rum to captain.

Auf dem Piratenschiff

N, N, W, W, S, S, Examine boat, Get canvas, S, S, E, E, N, N, N, W, S, Kick man, S, Get cleaver, N, N, E, S, S, S, W, Cut chain with cleaver, E, N, N, E, E.

2. Teil:

An der englischen Küste

Drop Rock Map Steel Cleaver, E, E, E, S, Search man, Get hat, N, W, W, S, Give some money to the old fisherman, N, W, Fish debris, drop pole, Examine newspaper, Remember Arcman, E, N, Give some money to tramp, S, E, N, N, E, Chandralt, give money to the man, W, S, E, Give hat, E, Examine drink, Follow man, Remember tycoon, S, W, Get hat, W, S, W, N, N, Remember Luxembourg, Examine hat, Enter plane.

In Luxemburg

N, W, Booth 11, Search man, Examine note, Remember jared, Examine ID, Remember Bob, E, E, N, N, W, W, N, U, S, E, N, Wear canvas, W, Get parchment, Examine parchment, E, S, W, N, D, S, E, E, E, E, S, S, E, Dig, Drop shell and parchment, Get leaflet, W, N, W, W, W, Examine leaflet, AN11649, Open box, Get gun, Drop box and leaflet, Get gun, E, N, W, W, N, U, S, E, N, N, N, N, E, Kill man, Get message, Examine message, Remember William Arcman.

Georg Ledermann
Mohlbergstraße 11
5030 Hürth

Hunchback

Wer etwas üben möchte, dem hilft POKE &511A,0 vor dem CALL.

Android One

Der bekannte POKE funktioniert bei mir nicht, dafür geht POKE &6D6F, 6. Ob es sich hierbei um einen Übertragungsfehler handelt oder um eine andere Version, ist mir nicht bekannt.

Gino H. Bark
Mahlesdorfweg 4
3000 Hannover

Johnny Reb II

Strategiefans werden diesmal schon fast verwöhnt, wenn man bedenkt, daß normalerweise nur wenige Programme dieser Art auf den Markt kommen. Neben "Arnhem" ist "Johnny Reb" das zweite Strategiespiel in diesem Heft. Den Hintergrund zu diesem Programm stellt der amerikanische Bürgerkrieg 1861 dar. Nord- und Südstaaten stehen



sich hier in einem mörderischen Kampf gegenüber. Die Rollen der beiden Parteien können entweder zwei Spieler oder ein Spieler und der Computer übernehmen.

Wie bei solchen Simulationen üblich, treten die eigentlichen Kampfhandlungen zu Gunsten strategischer Überlegungen in den Hintergrund. Das beginnt bereits nach Programmstart beim Aufbau des Schlachtfelds. Die Spieler können nacheinander Mauern, Häuser und Zäune aufstellen. Dazu kann man mit dem Cursor durch die ganze Landschaft fahren und geeignete Plätze auswählen. Die nächste Option erlaubt das Aufstellen der eigenen Truppen. Hier wird gewählt, ob Artillerie oder Infanterie eingreifen sollen, ob die Truppe aus Neulingen oder Veteranen besteht (verschie-

Arnhem

Vor kurzem noch im Fernsehen, heute im RAM des CPC. Wie Johnny Reb II gehört auch Arnhem zur Kategorie der Strategiespiele. Von Ihnen hängt es ab, ob die Operation Market Garden mißlingt oder nicht.



dene Kampfstärken), welche Bewaffnung eingesetzt wird usw. Die hier gebildeten Truppen können dann mit der nächsten Option positioniert werden. Wer gegen den Computer spielt, muß besonders aufpassen, da man nicht weiß, was der Gegner macht (seinem Freund kann man schon mal über die Schulter peilen). Punkt 4 dient Reserveeinheiten.

Drei unterschiedliche Spiellevel stehen zur Verfügung, von denen die Spieldauer abhängt. 30, 35 oder 40 Runden können gewählt werden. Die Optionen 5 und 6 dienen weniger dem Spiel, dafür mehr dem Service am Rande; z.B. kann man die Tastaturbelegung oder die Farbgestaltung ändern. Auch der Spielstand läßt sich abspeichern. Das finde ich gut, da eine komplette Spielrunde ziemlich lange dauern kann. Wenn all diese Punkte nacheinander abgearbeitet sind, beginnt das eigentliche Spiel, in dessen Verlauf man auch noch viele Möglichkeiten hat, es zu beeinflussen.

Insgesamt betrachtet gefällt mir "Johnny Reb" besser als "Arnhem", da es einfach interessanter gestaltet ist. Man muß allerdings viel Zeit mitbringen. Vielleicht ein Spiel für die anstehenden Wintermonate?

System: CPC 464/664/6128
 Hersteller: Lothlorien
 Bezugsquelle: Joysoft
 Rolf Knorre

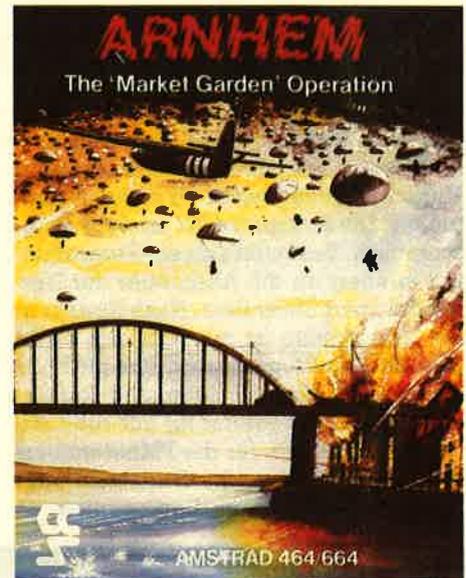


Johnny Reb II

Immer mehr werden die neuesten Militärspielchen als Strategiekämpfe gestaltet. Der Spieler soll nicht nur sinnlos umherballern, sondern vor dem Einsatz von Panzern und Kanonen erst mal seinen Krips anstrengen.

Arnhem

Strategiespiele nehmen im Gesamtbereich der Spiele-Software nur einen kleinen Platz ein, zum Leidwesen der Fans solcher Programme. Erweitert wird das Angebot jetzt durch "Arnhem", der Simulation eines Szenarios aus dem 2. Weltkrieg. Schauplatz ist Holland, genauer das Gelände rund um die Brücke von Arnhem. Simuliert wird die historische "Market Garden"-Operation von 1944, in deren Verlauf General Montgomery mit drei Luftlandedivisionen der alliierten Streitkräfte die Nazi-Herrschaft brechen wollte.



Die Grafik im Programm ist nicht überragend, reicht aber aus, das Schlachtfeld zu dokumentieren. Da im Vordergrund ja taktische und strategische Überlegungen der Spieler stehen, ist das Monitorbild eher zweitrangig. Wer sich für Strategieprogramme interessiert, wird an "Arnhem" sicher Gefallen finden.

System: CPC 464/664
 Hersteller: Cases Computer Simulations
 Bezugsquelle: Joysoft
 Stephan König



Rescue on Fractalus!

Vor einigen Jahren hat George Lucas weltweit mit seinem Kino-Mehrteiler "Krieg der Sterne" (Star Wars) großen Erfolg gehabt. Inzwischen laufen im Kino kaum noch Streifen von Lucas, dafür scheint er den Homecomputer entdeckt zu haben. Was "Rescue on Fractalus" nun genau mit Lucas zu tun hat, ob dieser zu Werbezwecken nur seinen Namen hergegeben hat, oder ob er aktiv an der Herstellung beteiligt war, kann ich nicht genau sagen. Die Programmidee könnte jedenfalls von ihm stammen, da sie ein bißchen an Sequenzen aus "Krieg der Sterne" erinnert.

Fractalus ist ein Planet irgendwo im Universum, auf dem gerade mal wieder Krieg herrscht. Einige Piloten der eigenen Mannschaft wurden über diesem Planeten abgeschossen, konnten sich aber retten. Sie benötigen nun dringend Hilfe. Also nichts wie rein in den Raumgleiter. Die Startprozedur erfolgt vollautomatisch. Besonders dieser Programmteil erinnert an die Abenteuer der Herren Solo und Skywalker. Nach Eintritt in die Atmosphäre ist man auf sich allein gestellt. Das Programm ist eine gute Mischung aus Flugsimulator und Actionspiel. Auf dem Monitor ist nur rund ein Drittel des Bildes für die Planetenoberfläche vorgesehen, der Rest wird vom Instrumentenpult des Raumgleiters in

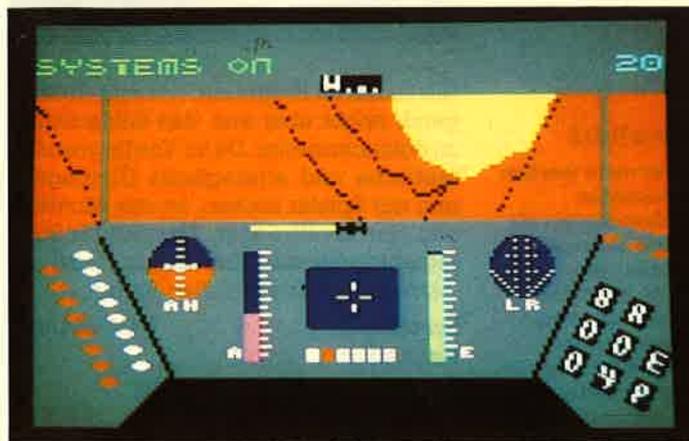
Anspruch genommen. Hier sind zahlreiche Instrumente angebracht, die der Spieler kontrollieren muß. Einige davon sind Höhenmesser, künstlicher Horizont, Zielerfassungsanlage, Kompaß und Schubkraft. Es ist gar nicht so einfach, alle Instrumente zu beobachten und auch noch aufzupassen, wohin man fliegt.

Fractalus ist ein sehr ungemütlicher Planet mit hohen Bergen und tiefen Schluchten. Die Steuerung wird einem nicht leicht gemacht. Sobald man in Reichweite eines Piloten ist, kann der Landevorgang eingeleitet werden: Systeme ausschalten, Luftschleuse öffnen (nachdem der Pilot angeklopft hat), Schleuse schließen und abheben, und schon kann es weitergehen. Wenn man nun einige Leute aufgenommen hat, ist es ratsam, diese zum Mutterschiff zurückzubringen. Dort kann der Spieler auch neue Energie auftanken.

Punkte gibt es für den Flug an sich, das Aufnehmen und Abliefern der Piloten, aber auch für Kampfhandlungen. Hin und wieder tauchen auf dem Monitor feindliche Flugzeuge oder Stationen auf. Denen kann man ausweichen oder sie abschießen, was den Score erhöht. Da neben den normalen Steuerungsmöglichkeiten des Flugzeugs noch zahlreiche Tastatureingaben möglich sind und auch die Angaben auf dem Instrumentenpult erst einmal einstudiert werden müssen, ist "Rescue on Fractalus" kein Spiel für Eilige. Nur mit viel Übung hat man irgendwann Erfolg.

Insgesamt betrachtet kann dieses Programm empfohlen werden. Ich würde mir zwar eine bessere bzw. detailreichere Grafik wünschen, bin aber auch so zufrieden. Die Freunde von Flugsimulatoren, denen ein bißchen Action gut tut, sollten sich dieses Programm einmal ansehen.

System: CPC 464/664/6128
 Hersteller: Lucasfilm Games
 Bezugsquelle: Activision
 Stephan König



Rescue on Fractalus!

Eine Mischung aus Flugsimulator und Actionspiel. Der Spieler hat die Aufgabe, mit seinem Raumgleiter Überlebende eines Absturzes auf Fractalus zu bergen und zum Mutterschiff zurückzubringen.

Glass

Quicksilva, eines der führenden englischen Software-Häuser beschert uns mit "Glass" einen fetzigen Actionknaller der Spitzenklasse. Angesiedelt im Science-Fiction-Milieu, muß der Spieler drei Städte des Planeten Glass von Feinden befreien. Er darf sich dabei als Pilot eines modernen Kampfflugzeuges fühlen. Das Programm ist aber deshalb kein Flugsimulator, sondern ein reines Action- bzw. Schießspiel.

Nachdem man die endlos lange Ladezeit der Cassettenversion überstanden hat, erscheint auf dem Monitor das Anfangsbild, das teilweise auch im späteren Spiel benötigt wird. Es handelt sich dabei um das Cockpit des Fluggeräts, genauer um eine Draufsicht auf das Instrumentenbord und den Kontrollmoni-



tor. Die Instrumente zeigen allerlei Schnickschnack an (hier wird man besonders an einen Flugsimulator erinnert), was jedoch ohne Bedeutung für das Spiel ist. Wichtig sind nur die Anzeigen für die Spielzeit, den Schutzschirm, die Flugrichtung und natürlich den Punktestand. Gesteuert werden kann das Programm über Tastatur oder Joystick. Optimal fand ich eine Mischung daraus, nämlich die Steuerung über Joystick und das Feuern über die Space-Taste.

Ein Druck auf "Feuer" startet das Spiel. Nach dem Verlassen der Basis tauchen auf dem Kontrollmonitor (nimmt etwa den halben Bildschirm in Anspruch) die ersten Feinde auf. Die Aufgabe des Spielers ist nun relativ einfach. Er muß ohne eigene Verluste möglichst viele dieser Gestalten abschießen, wofür es Punkte gibt. Die Zeit, die für diese



Glass

Der Spieler darf bei Glass solange herumballern, bis der eigene Schutzschirm verbraucht ist. Obwohl in der Hauptsache nur geschossen wird, hebt sich Glass doch durch Grafik und spielerisches Umfeld von der Masse solcher Spiele ab.

Aufgabe zur Verfügung steht, kann man der Instrumententafel entnehmen; umgerechnet ist es etwa eine Minute. Ist diese Zeit vorbei, wechselt das Bild zum nächsten Bereich. Hier wiederholt sich dieser Ablauf. Beendet wird das Spiel, wenn der eigene Schutzschirm aufgebraucht ist.

Hervorragend finde ich die Gestaltung der feindlichen Objekte. Mal tauchen bewaffnete Roboter, mal überdimensionale Raupenfahrzeuge auf; auch fliegende Maschineninsekten sind anzutreffen. Zu Anfang bewegen sich diese ungemütlichen Zeitgenossen noch recht langsam. Jeder weitere Fortschritt beschleunigt die Feinde aber erheblich. Neben den Schießrunden gibt es auch noch eine Probe der Geschicklichkeit. Der Spieler muß sich und sein Fahrzeug dann durch eine Landschaft steuern, die aus rechteckigen Objekten besteht. Die Ablaufgeschwindigkeit ist auch hier von der Spielstufe abhängig. Ein Tip: Wenn man ständig nach rechts steuert, ist das Spiel leichter zu schaffen. Eine Kollision mit einem Quader verringert die Schutzschirmenergie.

Ist das Spiel verloren, erscheint auf dem Bildschirm eine Einstufung, die sich nach den Punktzahlen richtet. Insgesamt gibt es 14 Stufen; ich habe es bisher nur bis Stufe 6 gebracht. Eine Besonderheit bei "Glass" offenbart sich erst hier. Der Spieler hat die Möglichkeit,

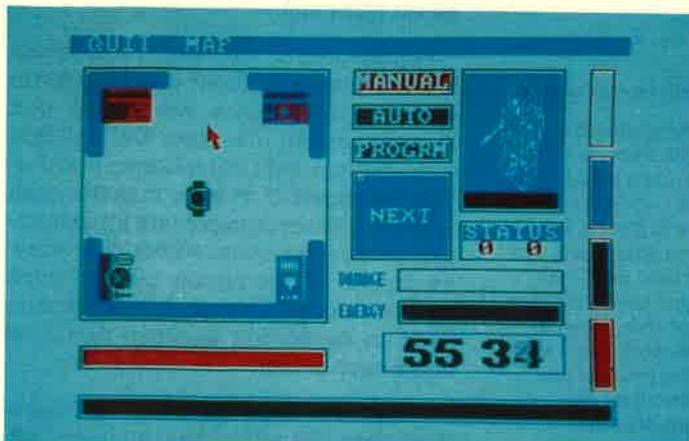
entweder ein neues Spiel zu beginnen oder beim schon Erreichten weiterzumachen. Der Punktezähler wird zwar wieder auf Null gesetzt, man hat aber durch diese Option endlich mal eine faire Chance, das ganze Spiel zu Gesicht zu bekommen.

Fazit: Trotz der einfachen Handlung halte ich "Glass" für ein tolles Actionspiel. Die Grafik ist wirklich gut gemacht, was auch für den Sound gilt. Was will man mehr?

System: CPC 464/664/6128
 Hersteller: Quicksilva
 Bezugsquelle: Peter West Records
 Rolf Knorre

Mission Omega

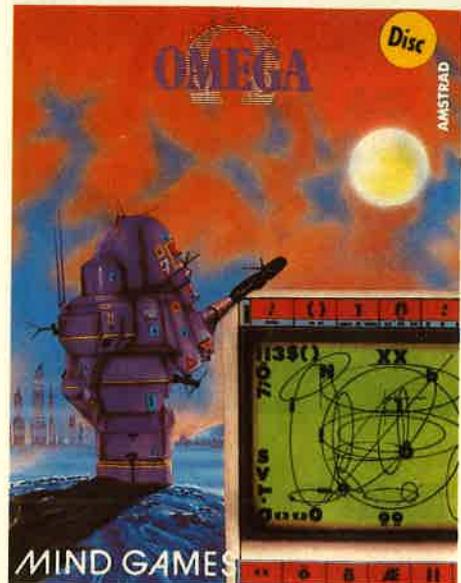
Ich will die Story des Spieles diesmal nicht näher beschreiben, da es sich wieder um einen kleinen Science-Fiction-Roman handelt. Wie so oft ist die Geschichte auch nur Beiwerk und von keinerlei Bedeutung für das Spiel. Bei "Mission Omega" handelt es sich um ein symbolgesteuertes Programm, das eine Mischung aus Action und Strategie bietet. Die Symbolsteuerung vermeidet lästige Tastatureingaben. Man kann also alle Aktionen über den Joystick einleiten.



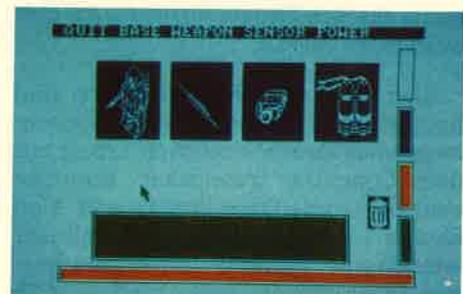
Mission Omega

Ein Actionspiel der Sonderklasse. Der Spieler muß hier einem Roboter in einem Labyrinth nachjagen. Das Zeitlimit beträgt 45 Minuten. Keine leichte Aufgabe, wenn man bedenkt, daß der Spieler nur einen kleinen Teil des Irrgartens sieht.

Die erste Spielphase besteht aus dem Konstruieren eines Roboters. Vier verschiedene Grundmodelle sind vorhanden, dazu zahlreiche Waffen, Ortungssysteme und Energiequellen. Der Spieler kann sich die einzelnen Möglichkeiten optisch auf dem Bildschirm anzeigen lassen, wozu auch die technischen Da-



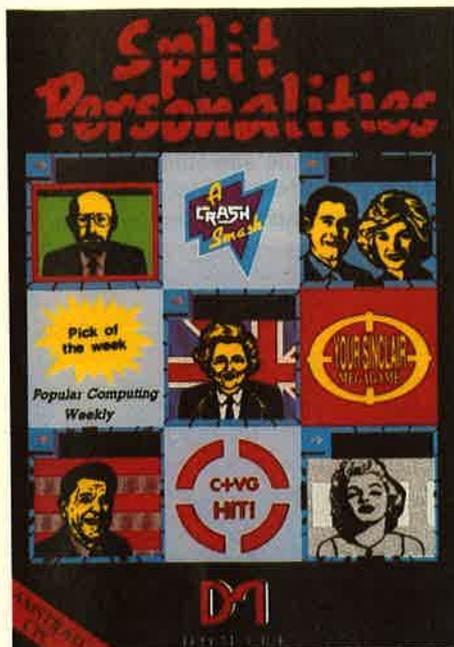
ten gehören. Natürlich kann man nicht beliebig viele Dinge mitnehmen, gewisse Beschränkungen der Transportkapazität sind zu beachten. Hat man sich für eine Zusammenstellung entschieden, geht es weiter im Spiel. Jetzt taucht der zuvor kreierte Roboter am Anfang eines Labyrinths auf; die Jagd kann beginnen. Innerhalb von 45 Minuten muß der Spieler einen Eindringling ausschalten, der die Erde zu zerstören droht. Vom Laby-



rinth sieht man immer nur einen kleinen Ausschnitt; man kann sich aber eine Karte auf den Bildschirm holen, die den Gesamtüberblick zeigt. Das erleichtert die Aufgabe erheblich. Trotzdem wird man zu Anfang Schwierigkeiten haben, die Zeit einzuhalten.

Ich halte "Mission Omega" für ein interessantes Spiel, das besonders technisch orientierte Actionfreaks begeistern kann. Grafik, Sound und Spielidee sind überdurchschnittlich.

System: CPC 464/664/6128
 Hersteller: Mind Games
 Bezugsquelle: Joystick
 Stephan König



Split Personalities

Es kommt wirklich nicht oft vor, daß mich ein Spiel total begeistert. Mal mangelt es an der Grafik, mal an der Spielidee, mal an sonstirgendwas. Das geht sicher nicht nur mir so. Die guten Sachen sind in einer Zeit, in der Monat für Monat haufenweise Programme erscheinen, selten geworden. "Split Personalities" ist jedoch so ein Programm, das voll überzeugen kann. Dabei ist die Idee gar nicht so umwerfend. Es geht um Puzzles, die zusammengesetzt werden müssen. Ein holländisches Programmiererteam, das sich Ernieware nennt, hat diese Idee umgesetzt.

Also, die Leute von Ernieware sind hergegangen und haben 10 Persönlichkeiten aus dem öffentlichen Leben mit dem Computer gezeichnet. Darunter sind Leute wie Clive Sinclair und Alan Shugar (Amstrad), aber auch Politiker wie Maggie Thatcher und Ronnie Reagan. Diese Bildchen haben sie dann in 25

Einzelteile zerschnitten. Auf dem CPC-Monitor sieht das folgendermaßen aus: Die linke Bildschirmhälfte stellt die Spielfläche dar. Auf der rechten Seite sieht man oben das Originalbild, wie es später aussehen muß, und unten eine Anzeige für Level, Score und Zeit. Mit einem Cursor, der die Größe eines der Teile hat, kann man nun beliebig viele Einzelheiten des Ganzen plazieren. Setzt man den Cursor auf ein Teil, leuchtet im Gesamtbild rechts die Stelle auf, wo der Schnipsel hin muß. Erschwerend kommt hinzu, daß man die Teile nur an den Spielfeldrand oder an bereits eingesetzte Teile schieben kann. Es ist z.B. nicht möglich, ein Teil in der Mitte des Feldes anzuhalten; es rutscht bis an den Rand durch.

Hat man ein Bild komplett zusammengesetzt, geht es mit dem nächsten weiter. Natürlich ist das aber noch nicht alles. An den Spielfeldrändern öffnen sich ständig Türen. Wird das nicht beachtet,

Flagge Bonuspunkte – die Flagge mit Hammer und Sichel sollte nicht eingesetzt werden –, und bei Maggie Thatcher erscheint auf einmal ihr Mann Dennis. So tauchen nach und nach immer mehr Gags auf, die nebenbei das Spiel ganz schön erschweren können. Und dann gibt es auch noch ein Zeitlimit, das nicht überschritten werden darf. Passt das doch, fängt man halt wieder von vorne an. Ächz!

"Split Personalities" macht süchtig. Meiner Meinung nach darf dieses Programm in keiner Sammlung fehlen, da es wirklich viel Spaß bereitet und auch hervorragend gemacht ist. Es soll übrigens auch bald eine deutsche Version mit deutschen Persönlichkeiten geben (oder solchen, die sich dafür halten). Darauf bin ich jetzt schon gespannt.

System: CPC 464/664/6128
 Hersteller: Ernieware/Domark
 Bezugsquelle: Joysoft
 Rolf Knorre

Split Personalities

Ronald Reagan komplett zusammengesetzt. Aber Vorsicht: Die sowjetische Flagge sollte man bei ihm nicht wählen, denn wie man ja weiß, ist er auf Moskau nicht immer gut zu sprechen.



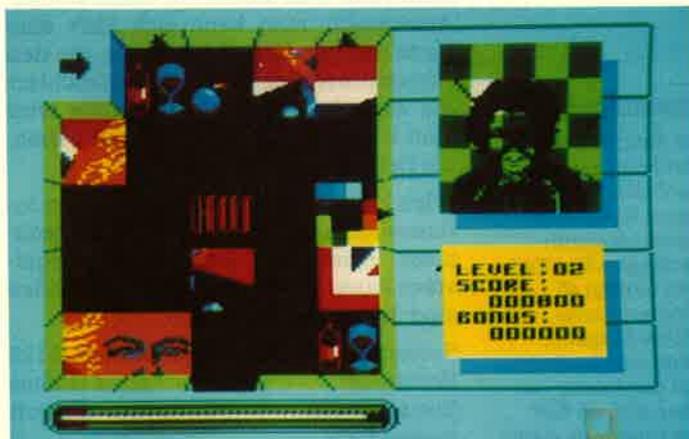
kann so ein Teil wieder verlorengehen. Außer den 25 tatsächlich benötigten Teilen tauchen andere Sachen auf, die verschiedene Wirkungen haben. Eine brennende Bombe muß z.B. schnellstmöglich rausgeworfen werden. Aber es gibt auch Bonuspunkte. Dabei tauchen zu jedem Bild spezifische Teile auf. Bei Ronald Reagan bringt die amerikanische

Bounder

Also Leute, dieses Spiel kann einen wirklich zur Verzweiflung treiben. Schlaflose Nächte und tränende Augen sind garantiert. Dabei macht das Ding auf den ersten Blick einen recht harmlosen Eindruck; auch die Anleitung (übrigens sehr kurz und sogar in deutsch) verrät nicht viel.

Worum geht's? Der Spieler muß einen kleinen Tennisball über den Bildschirm steuern. Hindernisse wie Berge und Mauern sind zu umgehen. Der Ball läßt sich aber nur auf sechseckigen Plattformen bewegen. Dort kann man ihn auch abprallen lassen, um größere Entfernungen zu überspringen. Außer den unbeweglichen Hindernissen gibt es auch noch feindliche Objekte, mit denen man nicht in Berührung kommen darf. Das war's auch schon.

Außer der Jagd nach einem Super-Highscore gibt es hier kein anderes Ziel.



Split Personalities

Wer glaubt, bei Split Personalities handele es sich um ein Legespiel für Zehnjährige, hat sich getäuscht. Der Spieler muß hier ganz schön auf der Hut sein, wenn er alle Steine aus dem Weg räumen und obendrein die Bonuspunkte kassieren will.

Aber gerade diese Programme haben ja in der Vergangenheit größte Erfolge gefeiert (siehe Pacman), und dazu könnte demnächst auch "Boulder" gehören. Die Bewegung des Balls würde ich eher als gemächlich bezeichnen. Trotzdem ist es ungeheuer schwierig, ihn optimal zu steuern. Das liegt natürlich auch an der Gestaltung des Spielfelds. Schon kurz nach Spielstart kommt man in Situationen, die unweigerlich zum Ballverlust führen.

Glücklicherweise stehen 7 Bälle zur Verfügung. Außerdem läßt sich das Spiel mit einer Pausentaste anhalten, damit das weitere Vorgehen in Ruhe überlegt werden kann. Es ist übrigens möglich,

te Sprungweite, ein Fragezeichen bringt Zusatzpunkte. Wer weit kommen will, muß sich im Laufe der Zeit einen Lageplan anlegen, um nicht immer wieder hängenzubleiben.

Wie schon zu Anfang angedeutet, kann "Boulder" süchtig machen. Gerade das einfache Programmkonzept sorgt dafür, daß man immer wieder angestachelt wird, es doch nochmals zu versuchen. Und genau das werde ich jetzt tun!

System: CPC 464/664/6128
 Hersteller: Gremlin Graphics
 Bezugsquelle: Peter West Records
 Rolf Knorre



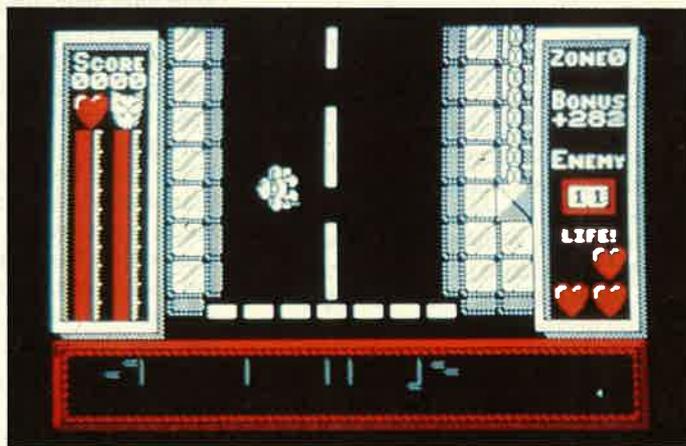
Boulder

Eine Mischung aus Geschicklichkeitsspiel und Tennis-Krimi. Man muß einen Tennisball über ein Netz steuern, wobei etliche Widersacher aufkreuzen, die einem das Leben schwer machen.

sich während des Spiels ausschließlich auf die rechteckigen Plattformen zu konzentrieren, da alles andere schädlich ist und umgangen werden muß. Einige dieser Plattformen sind besonders gekennzeichnet. So erlaubt z.B. ein Pfeil doppel-

Stainless Steel

Schießen, Autofahren, U-Boote jagen – alles in einem einzigen Spiel verpackt. Bei Stainless Steel geht echt die Post ab. Obwohl es schon massenhaft Schießspiele gibt, ist diese Art der Umsetzung recht gut gelungen.



Stainless Steel

Hier machen wir Bekanntschaft mit dem jugendlichen Superhelden Ricky Steel, der mit seinem Wunderauto "Nightwind" und unserer Hilfe die tollsten Abenteuer bestehen soll. Zusätzlich lernen wir noch den Finsterling Dr. Vardos kennen, der mit seinen Robotertuppen die Erde unterdrücken will. Soweit soll es aber nicht kommen.

Das Programm "Stainless Steel" ist in mehrere Abschnitte eingeteilt, die es zu

bewältigen gilt. Wer nach dem Programmstart einen Moment wartet, kommt in den Genuß des Demo-Mode. Hier kann man einen kleinen Überblick über das Spielgeschehen bekommen. Im Grunde handelt es sich dabei um ein Schießspiel mit Geschicklichkeitseinlagen, da das Fahrzeug ja auch noch zu steuern ist.

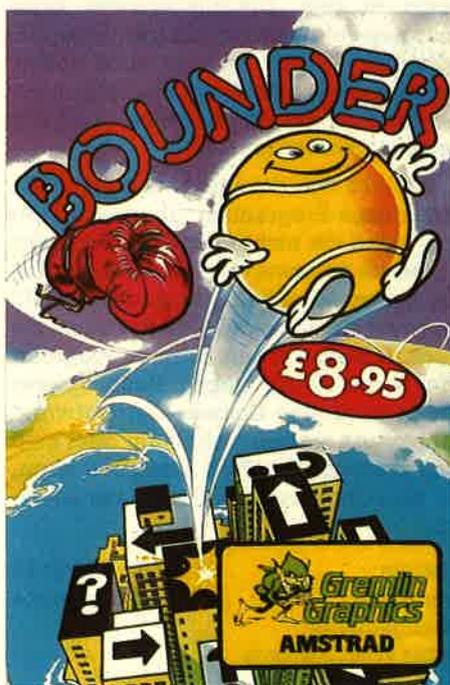
Die erste Spielrunde muß aber ohne Auto bewältigt werden. Ricky Steel ist da noch Fußgänger auf dem Weg zum Parkplatz, muß aber schon jetzt insgesamt 11 Feinde eliminieren. (Die Anzahl der jeweiligen Feinde wird in einem Zählwerk auf dem Bildschirm dargestellt.) Ricky bewegt sich über die Straße, auf der Autos parken, hin und wieder Tankfüllungen vorbeischieben und eben die Feinde angreifen. Hat er den Wagen bestiegen, fängt der Streß erst richtig an. Pro Leben steht in dieser Anfangsrunde einmal ein Schutzschirm zur Verfügung, der unverwundbar macht.

Weiter geht es dann mit Zone 1, in der das Wunderauto durch die Gegend fliegt. In Zone 2 muß unser Held Unterseeboote bombardieren, und in Zone 3 darf wieder gefahren werden.

Grafik und Animation würde ich in diesem Spiel als gut bezeichnen, ebenso die Spielidee. Neu ist die Geschichte zwar nicht, und Schießspiele gibt es auch schon massenhaft; trotzdem macht "Stainless Steel" Spaß. Und mehr kann man von einem Computerspiel doch nicht erwarten, oder?

System: CPC 464/664/6128
 Hersteller: Micro-Gen
 Bezugsquelle: Joysoft

Stephan König



Software immer preiswerter

Wenn man bedenkt, daß bis zum Erscheinen der Mastertronic-Produkte Cassetten dreißig bis vierzig Mark gekostet haben, wird klar, wie gut der Markt die "Billigsoftware" aufgenommen hat. Somit ist auch klar, daß Nachahmer nicht ausbleiben.

Zu diesem Thema möchte ich heute ein neues, englisches Software-Label vorstellen, das die Bezeichnung "Americana" trägt. Die Cassetten dieser Firma kosten rund 12.-DM, wobei es sich aber nicht nur um neue Programme, sondern auch um Titel handelt, die es für andere Computer bereits gibt bzw. gegeben hat.

Drei Programme aus dieser Reihe erhielten wir zum Test. Das erste Programm trägt den etwas merkwürdig klingenden Titel "Doppleganger". Es handelt sich dabei um ein Arcade-Adventure, also ein Abenteuer, das mit dem Joystick kontrolliert wird. Die Handlung klingt recht simpel. Sie sollen Gold-, Silber- und Bronzeteile finden und zusammentragen. Neu ist hier, daß der Spieler nicht nur eine, sondern zwei Figuren steuern kann, die nach Programmstart sofort bereitstehen. Allerdings kann immer nur eine Figur gesteuert werden, während die andere stillsteht. Um die Aufgabe zu lösen, sind aber beide notwendig. Der Grund dafür liegt in der Programmgestaltung. Die insgesamt 25 Räume des Hauses haben zahlreiche verschiedenfarbige Türen. Beide Figuren können nun z.B. durch die weißen Türen gehen, die grünen und blauen Türen kann aber immer nur eine Figur passieren. Zu andersfarbigen Türen muß erst ein passender Schlüssel gefunden werden. Besitzer eines Grünmonitors werden über diese Farbgebung nicht sonderlich begeistert sein, da die Unterschiede schwer festzustellen sind. Trotzdem kann man das Programm durchspielen, hat allerdings einen er-

Doppleganger

Software wird immer preiswerter. Auch dieses Arcade-Adventure bekommt man für runde 12 Mark. Wie der Spieltest zeigt, sind auch diese Billigprodukte zum größten Teil ihr Geld wert. Bei Doppleganger sollte man allerdings einen Farbmonitor besitzen.



höhten Schwierigkeitsgrad zu bewältigen. In Farbe ist die Aufgabe etwas leichter zu lösen. Da nur ein Leben zur Verfügung steht, muß der Spieler höllisch aufpassen. Bis alle Teile und Schlüssel zusammen sind (jede Figur kann nur einen Schlüssel und ein Teil tragen), vergeht schon einige Zeit. Grafik und Sound dieses Programms sind gut gelungen, so daß "Doppleganger" sein Geld wert ist.

Das nächste Programm wendet sich an die Freunde echter Adventures. "Spytrek" bietet außer dem eigentlichen Textabenteuer noch eine ansehnliche Grafik, die die Spielhandlung begleitet, aber auch abschaltbar ist. Die Handlung liest sich wie ein Krimi. Sie werden zum Geheimagenten und Topspion, der gerade einen neuen Auftrag erhält. Die schlechten Nachrichten reißen nicht ab. Erst vor zwei Wochen wurde ein anderer Topagent bei dem Versuch, geheime Pläne nach England zu schaffen, umgebracht. Bevor es dazu kam, konnte er die Pläne irgendwo in Europa verstecken. Nun sollen Sie diese Pläne herbeischaffen. Dazu wird Ihnen eine neue Identität verpaßt; ab sofort heißen Sie Mike Rodot. Das Spiel kann beginnen.

Sie liegen in einem verschlossenen Eichensarg und sind von jetzt an auf sich selbst gestellt. Weitere Hilfestellungen enthält die Anleitung nicht. Welches Vokabular zur Verfügung steht, muß erst

mühsam herausgefunden werden. Kommuniziert wird mit dem Programm über die üblichen Ein- oder Zwei-Wort-Befehle wie z.B. "Examine Coffin". Es geht aber auch komplexer: Take the wig, wear it then enter the hole. Das macht schon deutlich, daß Englischkenntnisse erforderlich sind. Die Eingaben werden je nach Länge einige Sekunden überprüft, was jedoch zu verschmerzen ist. "Spytrek" hat einige Gags auf Lager, die das Programm in Verbindung mit der kniffligen Aufgabe zu einem unterhaltsamen Spiel machen.

Der letzte Titel, "Eye Spy", hat schon wieder etwas mit Spionen zu tun. Diesmal geht es aber weniger um's Nachdenken als um schnelle Reaktion. Ein kleiner Kerl mit riesigen Glubschaugen, die ständig in Bewegung sind, muß wieder mal etliche Teile finden. Das Programm ist eine Mischung aus Plattform- und Labyrinthspiel. Alles ist ständig in Bewegung; überall lauern Gefahren auf den kleinen Spion. Es macht viel Spaß, durch die gut gestaltete Grafik zu huschen und dabei noch möglichst erfolgreich zu sein. Glücklicherweise kann man den Sound, der ständig mitläuft, abschalten. Für 12.-DM ist auch dieses Programm durchaus zu empfehlen, da es genauso gut ist wie seine erheblich teureren Konkurrenten.

Ich war von der guten Qualität aller Americana-Programme überrascht. So viel hatte ich nicht erwartet. Wenn der Trend zu preiswerten Programmen dieser Art weiter anhält (und damit ist wohl zu rechnen, da sowohl Mastertronic als auch Americana bereits neue Programme vorbereiten), haben Raubkopierer kaum noch eine Chance. Der Aufwand lohnt dann einfach nicht mehr. Was kein Kopierschutz der Welt erreichen kann, ist also über den Preis durchaus machbar.

System: CPC 464/664/6128
 Hersteller: Americana
 Bezugsquelle: Peter West Records
 Stephan König



Spytrek

Ein unterhaltsames Text-/Grafik-Adventure für englischsprechende Spielfreaks. In dieser turbulenten Spionagegeschichte muß man höllisch aufpassen, daß man Mike Rodot lebend ans Ziel bringt.

VORSCHAU



Das neue CPC-Magazin gibt es ab dem 26.11. am Kiosk

Joyce-Spiel

Der Joyce wird natürlich vorwiegend für professionelle Anwendungen benutzt, aber auch Profis spielen. Ob es sich für "Lord of the Rings", einem der ersten Spielprogramme für den Joyce lohnt, die Textverarbeitung zu unterbrechen, lesen Sie im Dezember-Heft.



Die Sprache C

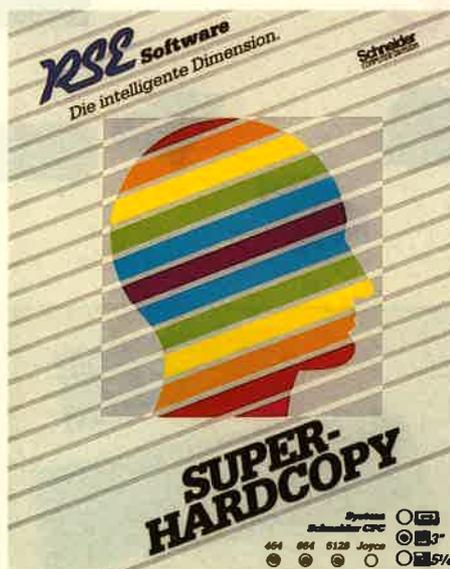
Programmiersprachen waren im CPC-Schneider-Magazin immer ein wichtiges Thema. Im Dezember-Heft befassen wir uns mit der Sprache C, die jetzt auch für Schneider-Computer zur Verfügung steht. Obwohl C eine sehr komplexe Sprache ist, wird sie immer beliebter. Vielleicht bekommen Sie nach der Lektüre unserer Einführung Lust, es auch einmal in C zu versuchen?

Hardware

Wir haben uns einiges an neuer Hardware für die Schneider-Computer angesehen. Unter anderem berichten wir über einen sog. Modulator, der die Computer von der Stromversorgung durch die Monitore unabhängig macht. Ganz besonders interessant ist natürlich auch das Knackmodul "Disk-Wizzard", das vor keinem Kopierschutz halt macht. Und wenn Sie mehrere Computer und Drucker verbinden wollen, wird Sie unser Bericht über einen Druckerumschalter interessieren.

Super-Hardcopy

Hardcopies vom Monitor der CPC-Computer sind nicht ohne weiteres möglich. Diesem Manko will ein neues Programm mit dem vielversprechenden Titel "Super-Hardcopy" abhelfen. Wir haben es für Sie auf Herz und Nieren getestet und sagen Ihnen, ob es hält, was sein Name verspricht.



Inserentenverzeichnis

Carat Soft	S. 25/93
Comal	S. 113
Computer Max	S. 124
Computer Studio Erding	S. 108
CSE Schauties	S. 14/105
Data Becker	S. 9
Becker, Ulrike	S. 123
data berger	S. 11/125
Deltacom	S. 65
Diepholzer Computerversand	S. 8
Die Zwei	S. 61
Ferdi's Computersoftware	S. 51
GAI	S. 126
Gerdes	S. 69
G. u. W.	S. 8
Hashagen	S. 123
Holschuh	S. 123
Joysoft	S. 97
Kunz	S. 95
Naujoks	S. 51
Peter West Records	S. 140
Philosoft	S. 15
PiZie-Data	S. 103
Plan Datenverarbeitung	S. 37
PR8-Soft	S. 6
Rätz-Eberle	S. 10
Richter	S. 123
Scholl	S. 123
Schuster	S. 70/71
SFK-Elektro	S. 122
Sikos	S. 124
Sybex	S. 2
T. G. Soft	S. 31
Ton & Technik	S. 15
Tüftler Union	S. 69
Van der Zalm	S. 37/125
vortex	S. 138/139
vortex-Versand	S. 17
Waldeck	S. 23
WHS-Hinderer	S. 118
Zimmerle	S. 65

Impressum

Herausgeber	Dipl.-Wirt.-Ing. (FH) Thomas Eberle Werner Rätz
Chefredakteur	Thomas Eberle
Techn. Redaktion	Werner Rätz
Ständige freie Mitarbeiter	Andreas Zallmann Manfred W. Thoma Rolf Knorre Markus Pisters Dipl.-Ing. H. P. Schwaneck Hans Joachim Janke Gerhard Knapieski Prof. Walter Tosberg Friedrich Lorenz Thomas Tai Christoph Schillo Berthold Freier
Versandservice	Elvira Rätz
Titelbild	Rainer Grinda
Anzeigen	Arno Weiß Es gilt die Anzeigenpreisliste Nr. 2 vom 24. 9. 1985.
Montage	Frederique Melchers
Satz	Druckerei Sprenger 7143 Vaihingen/Enz
Druck	Südd. Zeitungsdienst GmbH Druckerei- u. Verlags-GmbH 7080 Aalen
Vertrieb	Verlagsunion 6200 Wiesbaden
Anschrift des Verlages	Verlag Rätz-Eberle Postfach 1640 Pforzheimer Str. 43 7518 Bretten Telefon 0 72 52 / 4 29 48

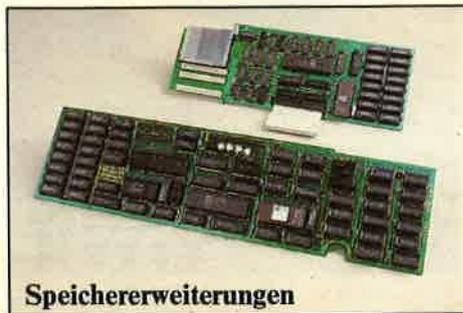
Manuskript- und Programmeinsendungen: Manuskripte und Programm Listings werden gerne von der Redaktion angenommen. Sie müssen frei von Rechten Dritter sein. Sollten sie auch an anderer Stelle zur Veröffentlichung oder gewerblichen Nutzung angeboten worden sein, muß dies angegeben werden. Mit der Einsendung von Manuskripten und Listings gibt der Verfasser die Zustimmung zum Abdruck in den vom Verlag Rätz-Eberle herausgegebenen Publikationen und zur Vervielfältigung der Programme auf Datenträgern. Für unverlangt eingesandte Manuskripte und Listings wird keine Haftung übernommen. Eine Gewähr für die Richtigkeit der Veröffentlichungen kann trotz sorgfältiger Prüfung durch die Redaktion nicht übernommen werden. Die gewerbliche Nutzung, insbesondere der Schaltpläne und Programme, ist nur mit schriftlicher Genehmigung des Herausgebers zulässig.

Das CPC-Magazin erscheint monatlich jeweils am letzten Mittwoch des Vormonats. Das Einzelheft kostet 6,- DM.

Die CPC-M



Floppy Disk Stationen



Speichererweiterungen



Hard Disk Stationen

Erhältlich bei Karstadt, Horten, Kaufhof, Quelle, Deutscher Bücherbund, Ringfoto, Brinkmann und im Fachhandel.

her

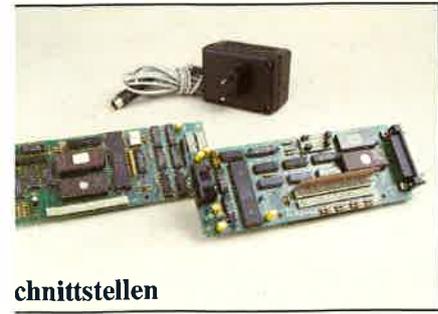


CPC 464 - CPC 664 - CPC 6128
 JOYCE - PC 1512
 Atari ST

Anfragen und Bestellungen:
 Tel. 07131/520 61
 vortex Versandservice
 Tel. 07131/520 65

User Sprechstunde:
 Montags und Donnerstags 18 - 21 Uhr
 Tel. 07131/520 61

vortex Computersysteme · Vertriebs GmbH
 Falterstraße 51-53 · 7101 Flein
 Tel. 07131/52061-63 · Telex 728915tron d



chnittstellen



Software

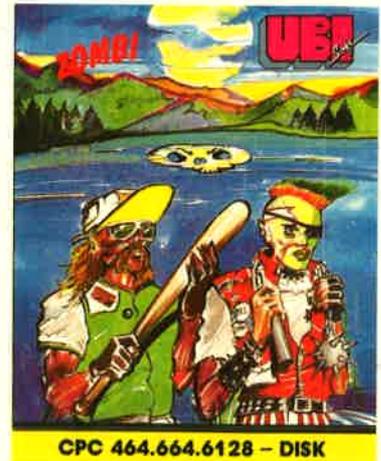
peter west records



Für Schneider, Spectrum und C 64



Für Spectrum und Schneider



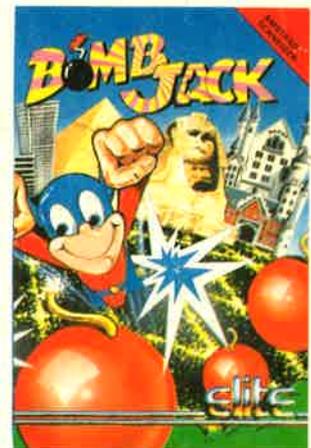
Nur für Schneider Disc



Für Schneider, C 64, Spectrum und C 16



Für Schneider 464



Für Schneider, C 64, Spectrum und C 16

Händleranfragen an:

**Peter West
Records GmbH**

Am Heerdter Hof 15
4000 Düsseldorf 11

Telefon: 02 11 / 50 21 31
Telex: 8 582 493 pwr
Telefax: 02 11 / 5 04 86 19