

wöchentlich

Compute mit

41/84 COMMODORE

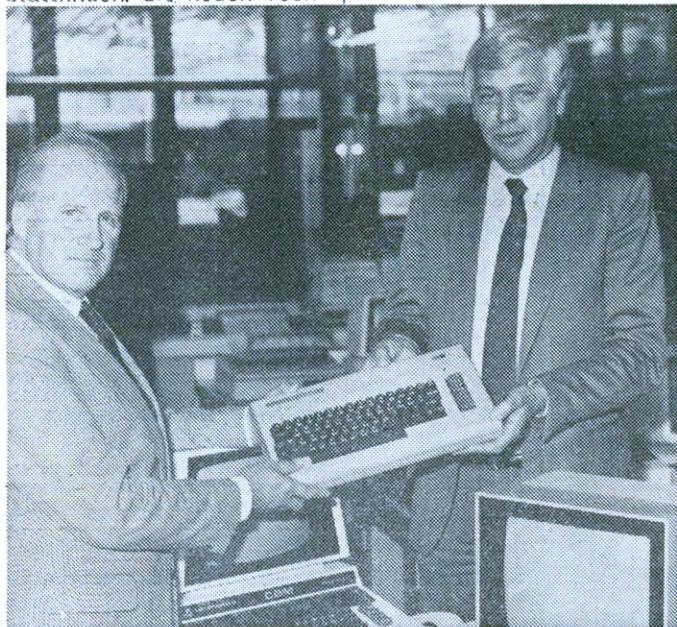
Das unabhängige Magazin für Anwender und Interessenten von Commodore Computern

2,80 DM
24 öS
2,80 sF

Aktion Computer und Bildung

Das CAL (Computer Aided Learning) wird nun von Software- und Hardwareherstellern gefördert, nachdem die Bundes- bzw. Landesregierungen nur zögernd die notwendigen Mittel zur Anschaffung bereitstellt. So kann in Zukunft an ca. 90 weiteren Schulen ein computergestützter Unterricht stattfinden, die neuen Tech-

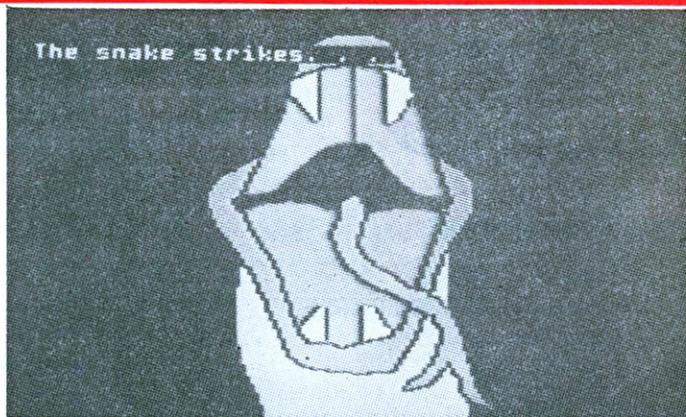
nologien halten Einzug in das Klassenzimmer. Was in anderen Ländern längst Alltag ist, wird auch bei uns in Zukunft zu sehen sein: Klassenzimmer mit vernetzten Computern. Dies wird auch nötig sein, denn die Entwicklung geht rasend voran.



Textverarbeitung für den C-64

Professionelle Textverarbeitung ist meist nur mit erheblichem Kostenaufwand verbunden. Die Software einiger Hersteller ist einfach zu teuer, andere wiederum nicht leistungsfähig genug.

Die "Compute mit" veröffentlicht für ihre Leser ein wirklich hervorragendes Textverarbeitungsprogramm zum Nulltarif. Durchgehende Menüauswahl, guter Editor und viele Raffineszen zum Ändern von Texten sind nur einige Leistungsmerkmale unseres Textverarbeitungsprogrammes zum Abtippen. Also dann, gleich anfangen auf Seite 29!



Mask of the Sun

Heute in unserer Adventure-Ecke: Mask of the Sun - Amerikas Superhit nun auch in Deutschland. Exklusiv auf Seite 3.

Was gibt es Neues auf dem Computermarkt:

u.a.

- Olympia Electronic Compact 2 mit Ili-Interface
- Neue Software für alle Zielgruppen
- 2.500.000 Quickshot - Joysticks SVI -
- Neue Dysan Kopfreinigungs-Diskette
- Joystick ohne Kabel
- CBS Computerspielprogramme
- Telexstar: Fernschreiben mit dem Personalcomputer



Basic-Kurs Teil 4

In der diesmaligen Lektion werden wir die Aufgabe des letzten Heftes lösen und uns weiter mit dem Erlernen der Befehle, die für das Arbeiten in Basic unerlässlich sind, befassen. In dieser Folge stellen wir Ihnen die For...Next Schleife vor.

Soft-Service	3
Basic Kurs	
Teil 4	4
Telex	
Ein Herz für Schüler...	5
Neue Software für alle Zielgruppen	
Olympia electronic Compact 2 mit Ili-Interface	8
Quickshot-Joysticks SVI-101	6
Neue Dysan Kopfreinigungsdiskette	
Bietet mehr Reinigungszyklen pro Anwendung, bei geringeren Kosten.	
Joystick ohne Kabel	7
CBS-Computerspielprogramme – Eine sinnvolle Ergänzung der Produktpalette	
Telexstar: Fernschreiben mit dem Personalcomputer	
Neu: ROW/Computer - Zubehör	
Bekannte Fotomarkte expandiert in "Mikro"	8
CM-Lexikon	
Diesmal ist der Buchstabe "C" an der Reihe	9
Software	
Mörderjagd (VC-20)	12
Battelstar (VC-20)	17
Top Secret (VC-20)	20
Joystick Grafik (VC-20)	38
Vier gewinnt (VC-20)	58
Textverarbeitung (C-64)	29
Magdomnat (C-64)	42

Bomber (C-64)	48
Asterix (C-64)	49
Think (C-64)	53

Tips & Tricks

Zwei Datasetten am C-64

Renumber für den C-64
Dieses Basic-Lade Programm erzeugt nach dem Start durch RUN ein Maschinen-Spracheprogramm.

CBM-64 Kontrolltaste	24
----------------------	-----------

Leserbriefe

Bücher
Heilborn/Talbott: Das C-64 Anwenderhandbuch
Das Anwenderhandbuch zum VC-20 Schacht: Basic – Schritt für Schritt

TV-Ecke
In unserer TV-Ecke geben wir regelmäßig Hinweise auf Computersendungen im Fernsehen

Softreviews
Sprachsynthesizer (für den C-64)
Bass-Compiler (für den C-64)

Werkstatt
Logik im Basicprogramm (Teil 5)

Assembler Kurs (Teil 5)

Berichte
Mikrocomputerboom erreicht neuen Höhepunkt:
Sprunghaftes Wachstum bei Commodore

Kleinanzeigen

Rätselseite

Impressum

Compute mit
erscheint wöchentlich im:
Roeske Verlag, Eschwege

Herausgeber:
Ralph Roeske

Redaktion:
Ralph Roeske (Chefredakteur, verantwortlich)
Gertrud Marx-Fischer (stellvertr. Chefredakteurin)
Horst Franke, Stefan Kaus, Thomas Morgen,
Stefan Ritter, Gisela Schwellach

Freie Mitarbeiter:
O. Amblank, H. Pauditz, Ch. Laentzsch,
B. Zimmermann, A. Kern, M. Hering, M. Bormann,
Th. Vogel, O. Probst, R. Burki, H. Götz, J. M. Heise,
H. Ahrensfeld

Gestaltung u. Fotografie:
Renate Wells, Gerd Köberich

Herstellung:
Roeske Verlag, Eschwege

Satz und Reproduktion:
Roeske Verlag, Eschwege

Druck:
Parzeller, Fulda

Vertrieb:
Inland (Groß-, Einzel- und Bahnhofsbuchhandel),
sowie Österreich und Schweiz:
Verlagsunion
Friedrich-Bergius-Straße 7
6200 Wiesbaden
Tel.: 06121/2660

Anfragen nicht an Vertrieb oder Druckerel, sondern nur an den Verlag!

Anschrift:
Roeske Verlag
Compute mit
Fuldaer Straße 6
3440 Eschwege
Tel. Sa. Nr. 05651/8558

Anzeigenleitung:
Roeske-Verlag, Eschwege
Christian Widuch,
Tel.: 05651/8559

Erscheinungsweise:
Compute mit erscheint wöchentlich, Mittwoch

Anzeigenpreise:
Bitte Mediaunterlagen anfordern

Urheberrecht:
Alle in Compute mit veröffentlichten Beiträge sind urheberrechtlich geschützt. Alle Rechte, auch Übersetzungen, vorbehalten.
Reproduktionen jeder Art (Fotokopie, Microfilm, Erfassung in Datenverarbeitungsanlagen, usw.) bedürfen der schriftlichen Genehmigung des Verlags. Alle veröffentlichte Software wurde von Mitarbeitern des Verlages oder von freien Mitarbeitern erstellt.
Aus ihrer Veröffentlichung kann nicht geschlossen werden, daß die beschriebenen Lösungen oder Bezeichnungen frei von Schutzrechten sind.

© Roeske Verlag, Eschwege
Redaktion "Compute mit"

Bezugspreise:
Einzelheft: 2,80 DM

Autoren, Manuskripte:
Der Verlag nimmt Manuskripte und Software zur Veröffentlichung gerne entgegen.

Sollte keine andere Vereinbarung getroffen sein, so gehen wir davon aus, daß Sie mit einem Honorar von DM 100,- pro abgedruckter Seite im Heft, einverstanden sind.

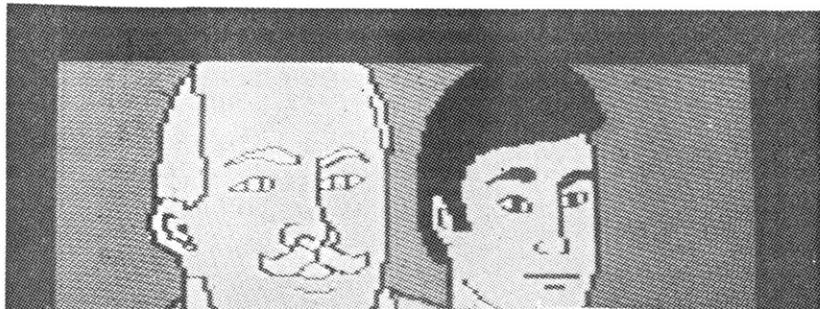
Bei Zusendung von Manuskripten und Software, erteilt der Autor dem Verlag die Genehmigung zum Abdruck und Versand der veröffentlichten Programme auf Datenträger.

Rücksendung erfolgt nur gegen Erstattung der Unkosten. Zusendungen von Software zur Veröffentlichung soll bitte folgendes enthalten:

Kopierfähige Kassette oder Diskette mit dem Programm (Computer-Bezeichnung), von Drucker erstelltes Listing oder Serie von Bildschirmfotos (keine Schreibmaschinenlistings), evtl. Bildschirmfotos von einem Probelauf, ausführliche Programmbeschreibung, (Erklärung der Variablenliste, Beschreibung des Bildschirmaufbaues, Farbe, Grafik usw.)

Für eingesandte Programmunterlagen kann keinerlei Haftung übernommen werden.

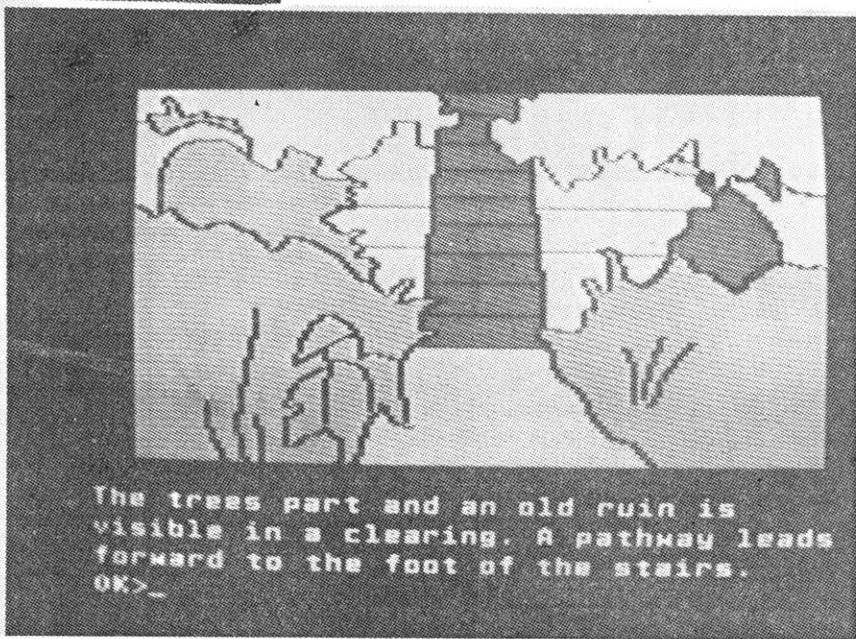
Mask of the Sun



"I brought you this map of that might contain the fabled the supplies are in the jee [waiting]

Finden Sie die Sonnenmaske der Azteken. Eine schwierige Aufgabe, denn noch niemand ist von dieser Expedition zurückgekehrt. Mask of the Sun ist ein englischsprachiges Text-Abenteuer mit faszinierender Grafik.

In wenigen Wochen wird eine deutsche Version der Firma Ariolasoft auf dem Markt erhältlich sein.



The trees part and an old ruin is visible in a clearing. A pathway leads forward to the foot of the stairs.
OK>

(sr)

soft service

Unser Softwareversand bietet Ihnen zu wirklich einmalig günstigen Preisen jede Woche die neuesten Programme, auf Kassette an.

Wenn Sie die hier aufgeführten Bedingungen beachten, können auch Sie von unserem Angebot Gebrauch machen.

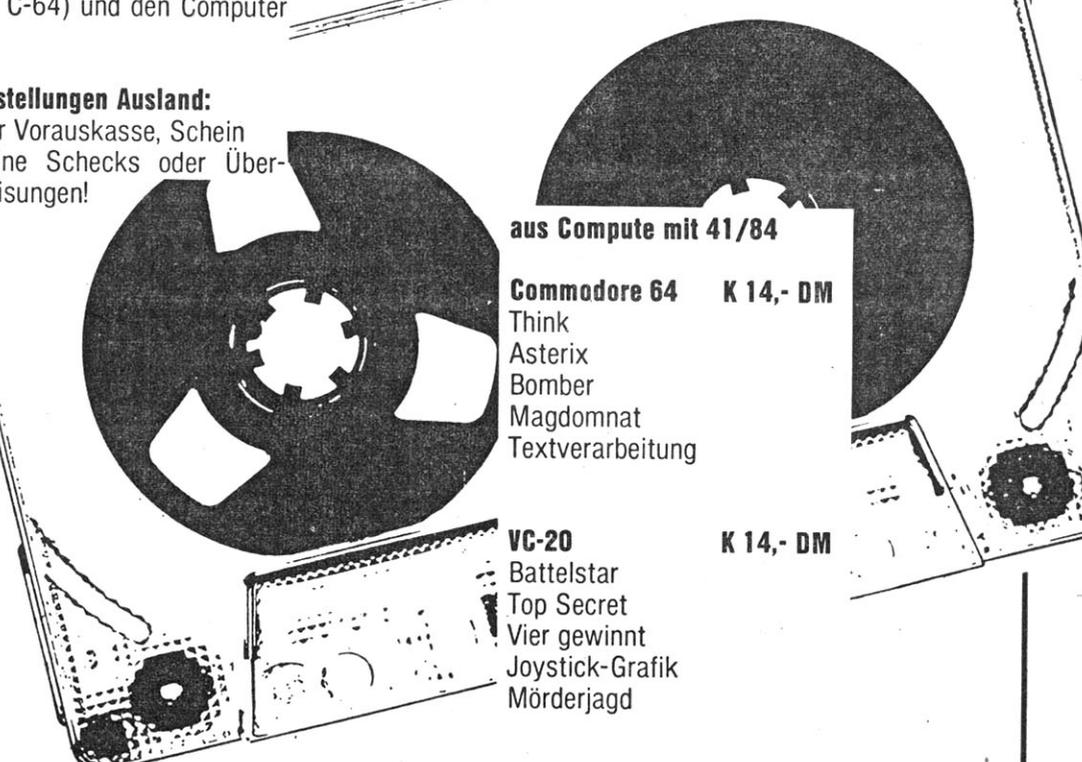
Bestellungen Inland:

Gegen Einsendung eines Schecks oder Vorauszahlung auf unser Konto bei der Raiffeisenbank Eschwege eG, Bankleitzahl 522 603 85 Kto.-Nr. 245 8888 senden wir Ihnen die gewünschten Programme schnellstmöglich zu. Geben Sie bitte unbedingt die genaue Bezeichnung der Kassette (z.B. Compute mit 37/

84 C-64) und den Computer an.

Bestellungen Ausland:

Nur Vorkasse, Schein
Keine Schecks oder Überweisungen!



aus Compute mit 41/84

Commodore 64 K 14,- DM
Think
Asterix
Bomber
Magdomnat
Textverarbeitung

VC-20 K 14,- DM
Battelstar
Top Secret
Vier gewinnt
Joystick-Grafik
Mörderjagd

**** COMMODORE 64 BASIC V2 ****
64K RAM SYSTEM 38911 BASIC BYTES FREE

Basic Kurs

Teil 4

Herzlich willkommen zum 4. Teil unseres BASIC-KURSES.

Letzte Woche hatten wir unseren ersten Programmablauf erstellt und auch gleich eine knifflige Aufgabe zu lösen. Wenn man sich das Programm der letzten Woche einmal genauer angesehen hat, so war die Lösung der Übungsaufgabe theoretisch gar nicht so schwer. In der ersten Zeile des Programms hatten wir eine Variable "TAGE" mit der Rechenbasis 365 belegt.

Diese Basis war natürlich etwas ungenau, da sie Schaltjahre nicht berücksichtigte, aber das war auch nicht so wichtig, da nur die grundsätzliche Funktion eines Algorithmus und die Abarbeitung eines Basic-Programms verständlich gemacht werden sollte.

Als ersten Schritt zur Lösung der Aufgabe müßten wir eine zweite Rechenbasis mit der Anzahl der Monate eines Jahres definieren. Wir schreiben also:

15 Monate = 12

Machen Sie sich keine Gedanken darüber, daß die Zeile 15 als letztes eingegeben wurde, der Computer ordnet sie an die richtige Stelle, was Sie leicht mit dem LIST-

Befehl kontrollieren können. Danach müssen wir einen zweiten Rechenvorgang in das Programm einbauen. Außerdem muß eine zweite Variable, die nachher das Ergebnis in Monaten beinhaltet erstellt werden.

Logisch wäre wenn wir diese Variable "ERGEBNIS 2" nennen würden. Der Computer erkennt jedoch nur die beiden ersten Buchstaben als Variablenname an. Wenn wir also zwei Variable mit dem Namen "ERGEBNIS" und "ERBRECHEN" definieren würden, so würde der Computer später im Programmablauf nur der Variable "ER" zweimal einen Wert zuweisen und das Ergebnis könnte nicht stimmen.

Demzufolge nennen wir die Variable also einfach nur "E", denn der Computer kann "E" und "ER" unterscheiden. Der Rechenvorgang lautet also:

```
35 E=ALTER*Monate  
oder  
35 E=AL*MO
```

das zweite Beispiel spart Zeit bei der Eingabe, ist aber später nicht mehr so schön übersichtlich.

Als letztes müssen wir noch die Printanweisung vervollständigen, wir ändern die Zeile 40 von:

```
40 PRINT"SIE SIND";ERGEB-  
NIS;"TAGE ALT"  
in:
```

```
40 PRINT"SIE SIND";ERGEB-  
NIS;"TAGE UND ";E;"MO-  
NATE ALT"
```

oder wir schreiben in Kurzform:

```
40?"SIND SIND";ER;"TAGE  
UND ";E;" MONATE ALT"
```

Wie Sie bemerken werden, ist es recht vorteilhaft Programme in der zweiten Schreibweise einzugeben. Wir werden bei längeren Programmen in späteren Lektionen darauf zurückkommen.

Kommen wir nun zu einem weiteren Befehl, den Sie in Ihre Programme implementieren können und zwar:

DIE FOR...NEXT SCHLEIFE

Diese Schleife wiederholt einen Teil Ihres Programms, so oft wie es im FOR-Befehl angegeben ist. Die Anweisung wird in der Form:

```
10 FOR 'VARIABLE' =  
X TO Y  
20  
30  
40  
50  
usw.  
100 NEXT 'VARIABLE'
```

in Ihren Computer eingegeben. Der Wert von 'VARIABLE'

wird bei uns zum besseren Verständnis fast immer 'T' sein, Sie können natürlich auch jede andere Variable dafür verwenden. Sie sollten jedoch darauf achten, daß diese Variable an keiner anderen Stelle des Programms abgefragt wird. Der Wert 'X' bezeichnet wo der Zähler starten soll und der Wert 'Y' wo der Zähler enden soll. Während des Zählens wird die Variable immer um ein STEP hochgesetzt. Im Normalfall ist der STEP immer 1. Man kann die Größe des Steps jedoch auch festlegen indem man an die FOR...NEXT Schleife die Anweisung STEP 'Z' anhängt. In 'Z' muß die Schrittweite stehen. Dieser Wert kann auch negativ sein. Wir geben also ein:

```
10 FOR T = 1 TO 209  
20 PRINT T  
30 NEXT T
```

und starten wie gewohnt mit RUN.

Der Computer wird auf dem Bildschirm die Zahlen von 1 bis 209 in schneller Folge von oben nach unten durchzählen. Dies geschieht allerdings so schnell das wir diesen aufregenden Vorgang gar nicht in seiner vollen Pracht bewundern können und man fragt sich, wie man dies wohl verlangsamen kann. Der soeben

gelernte Befehl kann auch diese Aufgabe übernehmen, indem wir ihn einfach eine Weile ohne Plan und Ziel zählen lassen. Wir schreiben:

```
10 FOR T = 1 TO 209
20 PRINT T
25 FORT2 = 1 TO 300
26 NEXT T2
30 NEXT T
```

Nach dem Starten mit RUN werden die Zahlen in kleinen zeitlichen Abständen auf dem Bildschirm erscheinen und zwar immer in dem Abstand in dem der Computer von 1 bis

300 zählt ohne in der Zwischenzeit etwas anderes zu tun.

Wie Sie sehen, ist der FOR...NEXT Befehl ziemlich nützlich, wenn man einen Vorgang oft wiederholen will. Wichtig ist noch, daß eine Schleife immer mit einem NEXT aufhören muß, da der Computer sonst eine Fehlermeldung ausgibt. Außerdem sollte man darauf achten, daß FOR...NEXT Schleifen nicht kreuz und quer ineinander verschachtelt sein sollen. Der Computer kann sich sonst sehr leicht in einer solchen

Schleife fangen und meldet sich nie wieder von selbst zurück.

In unserer Aufgabe für das nächste Mal verwenden wir das Programm vom letzten Heft und machen es noch ein wenig komfortabler, indem wir eine FOR...NEXT Schleife einbauen. Um die Sache jedoch nicht ganz so einfach zu machen, lassen wir den Befehl NEXT T heraus und Sie können sich überlegen an welche Stelle des Programms er am besten passen würde.

```
10 INPUT "VON WIEVIELEN
```

```
PERSONEN SOLL DAS
ALTER ERRECHNET WER-
DEN?";X
```

```
20 FOR T = 1 TO X
30 INPUT "ALTER:";AL
40 TA = 365
50 MO = 12
60 ER = AL*TA
70 E = AL*MO
80 PRINT "SIE SIND ";ER;"
TAGE UND ";E;" MONATE
ALT"
90 AL = 0
100 ER = 0
110 E = 0
120 PRINT "AUF WIE-
```

Ein Herz für Schüler...

Der Aktion 'Computer und Bildung' wurden jetzt in Frankfurt die ersten 360 Mikrocomputer des speziell für Schulen entwickelten Modells 4064 sowie des legendären C-64 plus Monitor 1701 übergeben. Hinzu kommt die Palette der Westermann-Lernprogramme (Rechenlöwe/Rechtschreib-

löwe).

Die Spende, die einschließlich der Peripheriegeräte, Software und Lehrmaterialien einen Wert von rund 1. Million Mark darstellt, nahm der Präsident der 'DABEI'. Deutsche Aktionsgemeinschaft Bildung - Erfindung - Innovation e.V., Bonn, Prof. Dr. Gerd Kortz-

fleisch, aus den Händen von Commodore-Vizepräsident und Europachef Harald Speyer (rechts) entgegen. Sie ist für 90 Grundschulen im gesamten Bundesgebiet bestimmt. Die Aktion 'Computer und Bildung' wurde unlängst durch die Bundesminister für Bildung und Wissenschaft sowie für Forschung und Technik ins Le-

ben gerufen. Ziel dieser Aktion ist es, Schulen und Bildungseinrichtungen durch konkrete Hilfen bei der Lehrerbildung, der Ausstattung mit Geräten und Computerprogrammen sowie bei der Bewältigung der mit neuen Informationstechnologien verbundenen pädagogischen Aufgaben zu unterstützen.



NEU FÜR DEN COMMODORE C-64



**Die preiswerte
Einsteiger-Software.**

SM SOFTWARE AG / MÜNCHEN

Neue Software für alle Zielgruppen:

Die zur Hannover Messe 84 angekündigten neuen Problemlösungen für die Home- und Semiprofi-Szene sowie den kommerziellen Markt sind nunmehr in erheblichem Umfang verfügbar. Damit wird das Programm-Spektrum von Commodore den Erwartungen einer Anwenderschaft gerecht, die eine immer vielfältigere und anspruchsvollere Struktur aufweist.

Der Mikro-Marktführer legte bei diesen Neuentwicklungen besonderen Wert auf einfaches Handling und hohen Bedienungskomfort. Das Unternehmen folgt damit dem breiten Käufertrend, Computer ohne Fachkenntnisse als persönliche wie berufliche Werkzeuge einzusetzen. Zugleich wird immer klarer, daß praxisnahe Software auch im Mikro-Markt zu einem Wettbewerbsfaktor von vorrangi-

ger Bedeutung aufrückt – also mindestens ebenso kaufentscheidend ist, wie das Preis-/Leistungsverhältnis der Hardware.

Im Mittelpunkt steht dabei die Tatsache: Die neuen Commodore-Rechner sind teilweise mit integrierten Software-Paketen erhältlich. Sie liegen aber dennoch nicht über den marktüblichen Hardware-Preisen.

Zu dem Bemühen, mit eigenen und dezidierten Lösungen auf dem Markt präsent zu sein, sieht sich der Mikro-Hersteller trotz des Angebots zahlreicher freier Programmierer und Software-Häuser deshalb veranlaßt: Geprüfte und getestete Software entspricht schlichtweg den elementaren Erwartungen der Commodore-User.

Im einzelnen sind folgende

neuen Programme nunmehr verfügbar:

Für Home- und Semiprofi-Rechner

- Synthesizer, ein Programm zum Komponieren und Üben
- Intelligente Computerspiele
- Schreibmaschinen-Kursus
- Basic-Lernkassette
- Grafische Kunst
- Logo – die neue Programmier- und Lernsprache, die auch für Kinder durch ihre Symbolelemente geeignet ist
- Datenmanager, ein Programm für die Datei-, Adress- und Mitgliederverwaltung etc.
- Integrierte Software (Plus 4)
- Btx-Modul, das den Commodore 64 Btx-fähig macht

Für kommerzielle bzw. professionelle Computer

- Lohn 700 – dieses Programm ist auf die 8.000er-Serie übertragbar
- Mitgliederverwaltung für CBM 80XX
- Emulator – ein Hilfsprogramm für CBM 700, das die Nutzung der 8.000er-Programme ermöglicht
- Superoffice – hierbei handelt es sich um ein interaktives Programm-Paket sowohl für die Dateiverwaltung als auch die Textverarbeitung der Systeme CBM 8296 und 700
- Platine für die Modellreihe 700 zur Nutzung der 8.000er-Software

Insgesamt sind im Mikro-Markt annähernd 5.000 Programme für Commodore-Computer (größtenteils von freien Software-Häusern erstellt) erhältlich.

Olympia electronic Compact 2 mit Iti-Intergace

Die Firma iti-Datentechnik schickt Interessenten kostenlos die Parametertabelle, um mit einer Olympia electronic compact 2 mit iti-Interface das Data-Becker-Textprogramm Textomat auf dem C-64 leistungsfähig zu implementieren.

Gegen eine geringe Gebühr kann eine Diskette mit Druckertabelle, Formular und Beispielprogramm enthalten. Mit dieser Anpassung läßt sich im laufenden Text die Schreibdichte (10, 12 und 15 Zeichen/Zoll), die Zeilendichte (1, 1,5 und 2-zeilig) ändern. Diese Ansteuerungen sind bei den meisten anderen Geräten dieser Preisklasse nicht möglich. Daneben sind die Hochstellen und Tiefstellen (Halbzeile positiv und negativ) sowie die weiteren

Zeichen des 100-Zeichentypenrads erreichbar. Daß alle Umlaute gedruckt werden ist heute selbstverständlich.

Die Maschine wird von iti-Datentechnik mit eingebautem seriellen Commodoreinterface ab Lager zu einem Preis von DM 1.498,- inkl. MwSt. geliefert.

Ein eventueller späterer Umbau auf das Centronicsinterface oder auf V.24 ist leicht möglich. Die Geräte werden auf Wunsch auch mit diesen Schnittstellen geliefert.

Desweiteren ist über die ON-OFF-Linetaste die Umschaltung auf normalen Schreibmaschinenbetrieb mit Korrekturspeicher möglich. Als Farbbänder stehen Einkarbonbänder, Multistricke und Nylonbänder zur Verfügung.

2.500.000 Quickshot-Joysticks SVI-101

Überaus werbewirksam wurde rechtzeitig zur HiFiVideo 84 in Düsseldorf vom deutschen Importeur der Quickshot-Joysticks bekanntgegeben, daß seit Produktionsbeginn 2.500.000 Stück des Typs SVI-101 auf dem Weltmarkt abgesetzt worden sind.

Der deutsche Generalimporteur, die Firma B. Jöllenbeck, nahm dieses Jubiläum zum Anlaß, auf dem Stand in Halle 9 das 2.500.000ste Exemplar sowie das inzwischen erhältliche Modell SVI-102, dessen besonderes Merkmal ein zuschaltbares automatisches Dauerfeuer ist, vorzustellen.

Das Interesse des Publikums war diesem Renner unter den Zubehörartikeln gewiß.



Neue Dyan Kopfreinigungs-Diskette

bietet mehr Reinigungszyklen bei geringeren Kosten pro Anwendung

Mit dem einfach anzuwendenden, wirtschaftlichen DYSAN Kopfreinigungs-Kit können Computeranwender nun Datenverlust, Systemausfall sowie Disketten- und Kopfverschleiß als Folge von Verschmutzung vermeiden.

Das Kit gibt es in einer 5 1/4- und einer 8-Zoll-Version für 1- und 2-seitige Laufwerke. Es enthält 2 Reinigungs-Disketten und eine Dosierflasche mit genügend Reinigungsfluid für 40 Reinigungsvorgänge. — Das bedeutet 50% mehr Reinigungszyklen gegenüber dem herkömmlichen Verfahren. Dazu sind die Kosten pro Reinigung günstiger.

Als Besonderheit haben die Reinigungs-Disketten einen dünnen Film-Aufkleber über der Schreib-/Lese-Öffnung. Mit ihm wird bei einseitigen Laufwerken eine gleichmäßige Druckverteilung und damit ein besserer Kontakt zwischen Diskette und Kopf beim Reinigen erreicht. Die sorgfältig abgemessene Abgabe

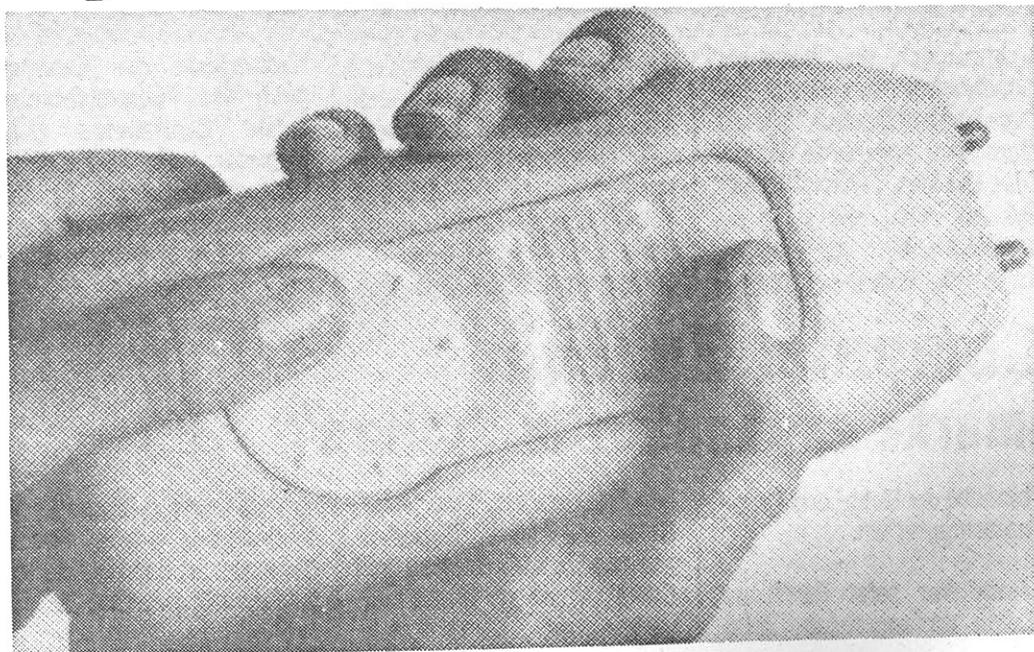


der Reinigungslösung vermeidet Über- und Underdosierung. Das Kit ist in einer kostenlosen, wiederverwendbaren Disketten-Archivbox verpackt. Entwickelt und hergestellt von Garlic Technology, einem zu DYSAN gehörenden Hersteller von Köpfen für Flexible Disk Drives, hat das DYSAN

Kopfreinigungs-Kit ähnlich strenge Qualitätssicherungsverfahren durchlaufen müssen, wie die anderen DYSAN Produkte; es wird eine 90-Tage Garantie gegeben. Das Kit wurde dem DYSAN Händlernetz im August vorgestellt. Die Verfügbarkeit beim Händler ist seit Oktober gegeben.

Die DYSAN Corporation ist einer der führenden Hersteller rotierender magnetischer Computer-Medien sowie zugehöriger Produkte und Dienstleistungen. Weitere Informationen durch Dyan GmbH, Eschborn

Joystick ohne Kabel



Der Remote Action Transmitter, der erste kabellose Joystick, wurde von dem englischen Hersteller Cheetah präsentiert. Der RAT arbeitet auf der Basis von Infrarotstrahlen, und kann bis zu einer Entfernung von ca. 10m zum Computer eingesetzt werden. Die Infrarotstrahlen werden zu einem Controller gesendet, der im Expansionsport des Computers steckt. Für den ZX-Spectrum von Sinclair ist der RAT bereits im Handel, für den Commodore wird er Ende des Jahres verfügbar sein.

(sr)

CM-LEXIKON

Es folgt Teil 3 unseres kleinen Wörterbuches. Haben Sie schon eine bißchen was dazugelernt? Auf jeden Fall lohnt es sich auch weiterhin, dabei zu bleiben! Immerhin sind wir schon bis zum Buchstaben C vorgeedrungen. Übrigens, es lohnt auch jetzt noch einzusteigen. Bis zum Ende unseres Computeralphabets werden garantiert noch einige Begriffe auftauchen, die auch Fortgeschrittenen bisher nicht verständlich waren. In diesem Sinne, mit Vollgas weiter voran!

Centronics-Schnittstelle:

Parallel-Schnittstelle zur Übertragung von Computer-Daten in den Drucker. Die vom Computer zum Drucker übermittelten Daten werden bitparallel auf 8 parallelen Leitungen übertragen. Da im ASCII-Code ein Zeichen aus 8 bits besteht, wird auch immer gleichzeitig ein ganzes Zeichen zum Drucker gegeben. Dadurch ist diese Art der Datenübertragung wesentlich schneller als bei der seriellen Schnittstelle, wo immer 8 bits hintereinander gesendet werden.

Cobol:

höhere Programmiersprache, wird meist für kaufmännische Aufgaben eingesetzt.

Compiler:

Als Compiler bezeichnet man ein Programm, das Programmschritte der höheren Programmiersprachen in Maschinensprache übersetzt, so daß der Mikroprozessor die Befehle auch versteht. Der Compiler übersetzt dabei in prozessor eigene Befehle, da jeder Mikroprozessor nur seine eigene Maschinensprache versteht.

CAL:

computerunterstütztes Lernen. Gewinnt immer mehr an Bedeutung, gerade in Schulen wird das CAL gezielt gefördert.

CP/M

eines der weit verbreitetsten Betriebssysteme bei Personal Computern. CP/M wurde 1976 von Digital Research entwickelt und kann heute auf eine fast unübersehbare Softwarepalette zugreifen, CP/M bedeutet "Control Programm für Mikrocomputer" und wurde für 8 bit Rechner erstellt.

CPU:

Zentraleinheit eines Computer-Systems. Die CPU ist das Steuerwerk, sie holt Befehle, decodiert und führt sie aus. Ein Leistungsmerkmal ist die Anzahl der Bits, die eine CPU auf einmal verarbeiten kann. Diese Maßeinheit nennt man Wortbreite. Es gibt Mikroprozessoren mit 8 Bit, 16 Bit und auch 32 Bit Wortbreite.

COMAL:

höhere Programmiersprache, die Grundelemente von Basic und Pascal aufweist. Mit COMAL kann strukturiert programmiert werden, d.h. ein Problem wird in Unterprobleme aufgeteilt und nacheinander abgearbeitet.

CURSOR:

Ein blinkender Zeiger auf dem Bildschirm. Er gibt an, an welcher Stelle des Bildschirms Eingaben gemacht werden können. Der Cursor ist freibeweglich, d.h. er kann positioniert werden. Im Regelfall sind auf Computertastaturen extra Cursortasten.

Character Generator:

elektronischer Schaltkreis, der normalerweise im ROM untergebracht ist. Er hat die Aufgabe, binäre Computerdaten so

umzuwandeln, daß die entsprechenden Zeichen und Buchstaben entstehen. Ein Zeichen ist dabei aus vielen Einzelpunkten entstanden, für diese Darstellung ist der Character Generator verantwortlich.

(sr)

Compilersprachen:

im Prinzip alle höheren Programmiersprachen. Diese Programme sind an der menschlichen Sprache orientiert, der Computer versteht sie nicht. Jedes Programm muß vom Interpreter oder Compiler in den jeweiligen Maschinencode übersetzt werden. Dabei ist ein vom Compiler übersetztes Programm bis zu 50-mal schneller als ein vom Interpreter übersetztes.

Compilersprachen sind u.a. Basic, Fortran, Pascal.

Control-Bus:

Auf den Leitungen des Control-Bus werden Steuerzeichen übermittelt. Sie haben die Aufgabe, die Bauelemente so zu schalten, daß störungsfreier Datenverkehr erfolgen kann.

CPI:

Characters per Inch. Maßeinheit bei Druckern für die Zeichendichte und damit im allgemeinen Größe der Schrift.

CPS:

Zeichen pro Sekunde. Wichtig u.a. für die Leistungsfähigkeit eines Druckers, gibt an, wieviel Zeichen pro Sekunde gedruckt werden.

Beispiel:

Matrixdrucker 100-160 cps
Typenradrucker 50-100 cps

Microcomputerboom erreicht neuen Höhepunkt:

Sprunghaftes Wachstum bei Commodore

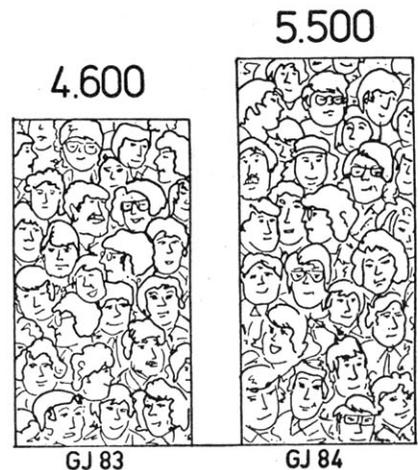
Frankfurt, den 6. September. – "Unser gegenwärtiger Erfolg mag manchen Mitbewerbern wie ein pures Wunder erscheinen. Doch Tatsache ist, daß er aus einer konsequenten, ja radikal anwendungsnahen Unternehmens-, Produkt- und Absatzpolitik resultiert."

Mit diesen Worten kommentierte Harald Speyer, Vice President von Commodore International, vor Journalisten den anhaltenden Höhenflug des in rund 60 Ländern aktiven Mikrocomputer-Konzerns. Die Resultate des vergangenen Geschäftsjahres (01.07.83 – 30.06.84) übertrafen die ohnehin kühnen Erwartungen des Unternehmens erheblich: Weltweit erzielte Commodore ein Um-

satzplus von 86 Prozent und damit 1,267 (0,681) Mrd. US-Dollar. Zugleich schnellten die Gewinne (vor Steuern) um 99 Prozent auf 224,5 (112,9) bzw. (nach Steuern) um 57 Prozent auf 143,8 (91,7) Mio. US-Dollar hoch. Die Umsatzrendite lag (brutto) bei 18 (17) bzw. (netto) bei 11 (13) Prozent. Pro Aktie stiegen die Gewinne um 63 Prozent auf 4,66 (2,86) US-Dollar. Der Pro-Kopf-Umsatz rangierte

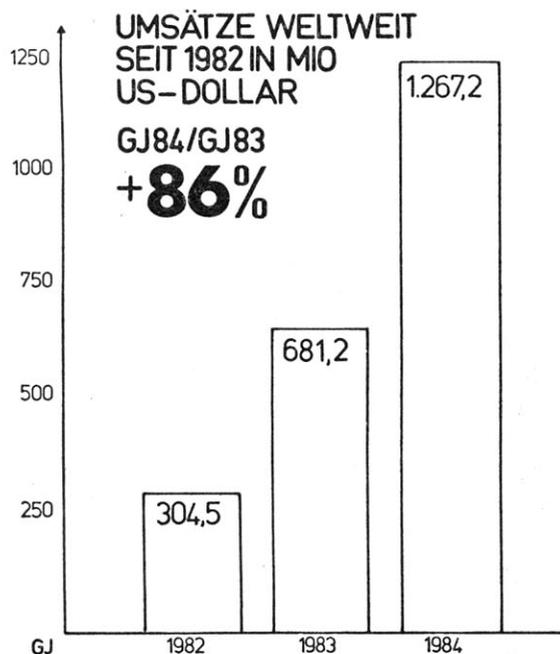


ANZAHL DER MITARBEITER
WELTWEIT (PER 30.6.)

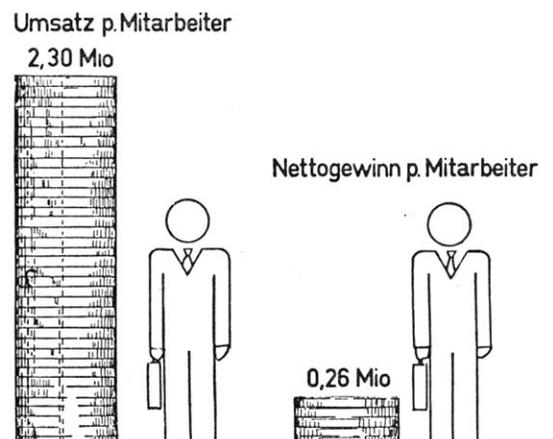


schlaggebender Bedeutung sein. So vor allem: Investition sämtlicher Gewinne ausschließlich in die eigene Gesellschaft, möglichst hohe Selbstversorgung mit Bauteilen, partnerschaftliche Kooperation mit den Distributoren wie auch Consumer-Marketing in allen Absatzfeldern.

Neben der stark popularisierten Werbung sei dabei auch die Direktansprache der Interessenten wichtig, wie sie auf der am 06. September eröffneten "Vierten Internationalen Commodore-Fachausstellung" (CFA) erfolgte. Hierbei handelt es sich um die erste und bislang einzige Messe eines DV-Anbieters in



UMSATZ UND GEWINN PRO MITARBEITER WELTWEIT GJ 84 IN US-DOLLAR



bei 2,30 (1,48) und der -Nettogewinn bei 0,26 (0,20) Mio. US-Dollar. Die Zahl der Mitarbeiter kletterte (per 30.6.84) um 20 Prozent auf 5.500 (4.600).

Speyer geht davon aus, daß der Höhepunkt des Mikro-

Booms "erst in einigen Jahren erreicht sein wird und Commodore bis dahin weiterhin überdurchschnittliche Expansionsraten erzielt." Dabei würden – führte der Vice President weiter aus – einige wesentliche Faktoren für den künftigen Erfolg von aus-

Deutschland. Sie bietet diesmal über 20.000 erwarteten Besuchern eine umfassende Präsentation der Hard- und Software sowie Serviceleistungen aus der gesamten Commodore-Welt. Rund 100

Aussteller waren auf etwa 4.000 Quadratmetern in der Halle 1 des Frankfurter Messegeländes vertreten.

Im Bereich der Produktpolitik will sich Commodore nach

Speyers Worten auf zukunfts-trächtige Neuentwicklungen konzentrieren. Dabei gelte es, möglichst hohe Stückzahlen pro Modell zu erreichen, um via Kostendegression noch günstigere Preise realisieren zu können. Ein kontinuierlicher Ausbau der eigenen Fertigungskapazitäten sei im Gang, was eine Entschärfung der gegenwärtigen Lieferengpässe erwarten lasse.

Speyer: "Unter Wettbewerbsaspekten kommt es darauf an, immer komfortablere und bedienungsfreundlichere Geräte anzubieten – so beispielsweise mit integrierter Software." Um erhöhte Kaufanreize zu schaffen, werde Commodore in allen Ländern die Palette an Lernmedien, Begleit- und Arbeitsmaterial in nächster Zeit drastisch er-

weitern. Schließlich prognostizierte der Commodore-Chef einen gewaltigen Boom für Mikrosoftware, aber auch für andere neue Zweige im Umfeld der Kleinrechner (Zusatzgeräte, Zubehör, Aus- und Fortbildung etc.).

Einen imposanten Wachstumssprung realisierte im abgelaufenen Geschäftsjahr (ebenfalls 01.07.83 - 30.06.84) die deutsche Commodore Büromaschinen GmbH. Der Gesamtumsatz stieg um 136 Prozent auf 742,9 (314,4) Mio. Mark. Am stärksten expandierte dabei die Zentrale Frankfurt (incl. fünf Vertriebsbüros): Die Inlands-Erlöse schossen gar um 267 Prozent auf 366,7 (100,0) Mio. Mark hoch. Die Exporte des Werkes Braunschweig

beim Werk Braunschweig bei 1,79 (1,14) Mio. Mark. Diese ungewöhnlich günstige Relation erklärt Stumpf damit, "daß bei Commodore die eigenen Computer, wie die Informationstechnologie überhaupt, optimal zum Einsatz gelangen und damit zu einer äußerst günstigen Arbeitseffizienz führen." Die Zahl der Mitarbeiter stieg innerhalb der letzten zwölf Monate um 12 Prozent auf 339 (304) – davon arbeiten 140 in der Zentrale Frankfurt (incl. Vertriebsbüros) und 199 im Werk Braunschweig (Stand: 30.06.84).

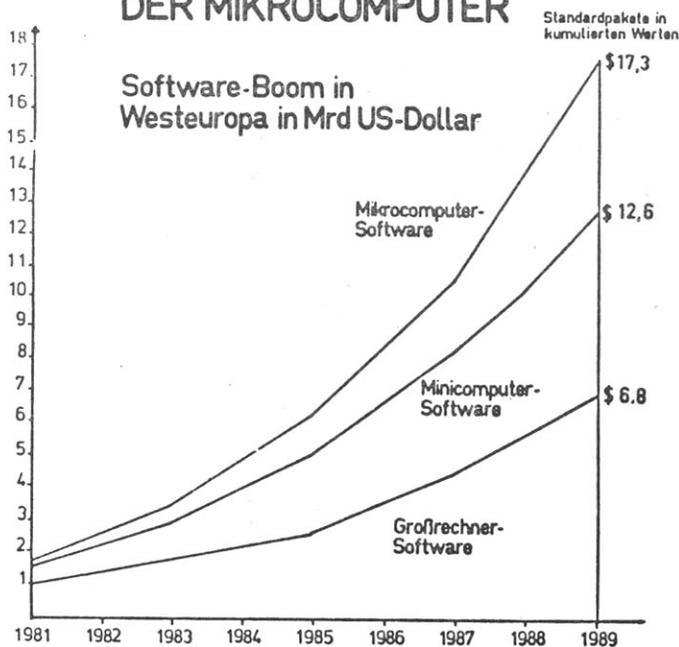
Wie Stumpf weiter darlegte, sind nunmehr die zur Hannover Messe angekündigten neuen Home- bzw. Semiprofi-Computer (Commodore 116 und 16 sowie Plus 4) wie auch der kommerzielle Rechner CBM 8296 D lieferbar. Zugleich wurde das Produktionsprogramm des Werkes Braunschweig aufgestockt: Dort werden nun auch die neue Modellreihe 8296 D sowie Floppy-Laufwerke (1541) hergestellt. Die Aufnahme der Fertigung weiterer Hardware wird gegenwärtig vorbereitet (bisherige Palette: Commodore 64, CBM 8296 und länderspezifische Modelle). Zugleich sei das verfügbare Software-Spektrum im Home- wie im Kommerz-Bereich kräftig erweitert worden, betonte Stumpf.

- markt sowie der Semiprofi-Bereich wachsen gegenwärtig am schnellsten.
- Der Ausleseprozess unter den Distributoren eskaliert stärker als unter den Mikro-Anbietern.
- Zunehmende Einstiegs- und Existenzprobleme für kleine Neuanbieter.
- Trend zu anwendungs- oder berufstypischen Computern (integrierte Spezial-Software).
- Programm-Produzenten werden mehr und mehr zu Zulieferern der Hardware-Hersteller oder zu OEM-Partnern.
- Die Arbeitsteilung zwischen Personal- und Großcomputern setzt sich durch und erfordert neuartige DV-Konzepte.
- Programmbanken für Spezial-Software (ähnlich wie Videotheken) sind im Kommen begriffen.

Wie Stumpf abschließend bekanntgab, verfügte Commodore (auf der Basis von IDC- und Diebold-Erhebungen per 1.1.84) zu Beginn des laufenden Kalenderjahres über einen Gesamtanteil von etwa 50 Prozent im Home-, Hobby- und Semiprofi-Markt und von mindestens 25 Prozent im kommerziell-professionellen Bereich (in installierten Werten).

Der Mikro-Markt in Deutschland – so die neueste Prognose von Diebold – "wird in den kommenden Jahren bei kontinuierlich verbesserter Leistung der Geräte und weiter sinkenden Preisen ungestüm wachsen": Während zum Jahresbeginn 1984 hierzulande etwa 1,02 Mio. Rechner installiert waren, sollen es – so die Beratungsgesellschaft – Anfang 1986 bereits 2,55 und im Januar 1988 schon 4,08 Mio. Stück sein.

EXPANSIVES UMFELD DER MIKROCOMPUTER



Weitere Mikro-Wachstumsbranchen:

- Peripherie
- Zubehör
- Fachzeitschriften/Bücher
- Mikroschulen und Lernprogramme
- Beschaffungs-/Einführungsberatung

Quelle: IDC

konnten immerhin um 66 Prozent auf 356,8 (214,4) Mio. Mark gesteigert werden. Wie der für den deutschen Vertrieb zuständige Geschäftsführer Alwin Stumpf berichtete, entwickelten sich die

– nicht veröffentlichten – Gewinne ebenfalls "in der erwarteten positiven Weise". Der Pro-Kopf-Umsatz lag insgesamt bei 2,19 (1,03) Mio. Mark; bei der Zentrale Frankfurt sogar bei 2,61 (0,86) und

Als vorherrschende und generelle Tendenzen im Mikro-Markt der Bundesrepublik konstatierte der Geschäftsführer:

- Immer mehr Nachfrage konzentriert sich auf immer weniger Anbieter.
- Der Aus- und Fortbildungs-

MOERDER JAGD

für den VC-20 mit mindestens 8K

Graf Hinrich von Fallobst ist ermordet worden. Der Mörder befindet sich noch im Haus. Der Spieler, der sich als Meisterdetektiv behaupten soll, beginnt mit seinen Ermittlungen und nennt seine Verdachtsmomente.

Zuerst wird nach dem möglichen Tatort gefragt, anschließend nach dem Verdächtigen, dann nach dem Tatwerkzeug. Der Computer zeigt an, wieviel Angaben richtig waren. Bei jedem Spielzug rückt der Mörder näher an den Detektiv heran. Nach 10 erfolglosen Ermittlungsschritten ist das Spiel für den Meisterdetektiven aus und das Geheimnis wird gelüftet.

In jeder Phase des Spiels kann die Alibitaste 'H' eingegeben werden. Es erscheint daraufhin ein Alibi eines der verdächtigen Personen. Allerdings hat der

Mörder dann Zeit, noch näher zu kommen...

Variablenliste:

C\$ (1-5), A\$ (1-5), B\$ (1-5): Beinhalten die Namen der 5 Verdächtigen, Tatorte und Tatwerkzeuge.

I, K, B, V, W, X, Y, Z: Werden benötigt, um die Verdächtigen, die Tatorte und die Tatwerkzeuge zu mischen.

TW\$ (1-5), M\$ (1-5), TA\$ (1-5): Beinhalten nach dem Mischen die Namen der 5 Verdächtigen usw.

TI\$: Zeitmessung

AB\$: Übernimmt die Zeit für das letzte Bild

Z: Zählt die Ermittlungsschritte

P: Zeigt an, wieviele Eingaben richtig waren

Programmaufbau:

240: Zuordnung der Variablen, Mischen der Verdächtigen, der Tatorte und der Tatwerkzeuge

300-400: Einstiegsgrafik

500-600: Die Verdächtigen, die Tatorte und die Tatwerkzeuge werden genannt

700: Zeit wird auf "000000" gestellt

1000-1400: Abfrage der Verdachtsmomente und Überprüfung der Eingaben

1500-1700: Ausgabe des Ermittlungsstandes

2000-2880: Täter überführt

3000-3380: Mörder hat den Detektiv erreicht

4000-4240: Bekanntgabe eines Alibis

5000-5240: Spielerklärung

10000-16700: Grafikteil

```

1 POKE36879,25:GOSUB5000
5 PRINT"□":CLR:POKE36879,110:PRINT"MOERDER IN VERBRECHEN WIRD "
6 PRINT"GESCHEHEN...KEIN ZEUGE.....KEIN LAUT....."
10 C$(1)="KUECHE":C$(2)="WOHNZ IMMER":C$(3)="TERASSE"
15 C$(4)="FLUR":C$(5)="SCHLAFZ IMMER"
20 A$(1)="FR.STRAUCH":A$(2)="BARON BLITZ"
25 A$(3)="HR.DACKEL":A$(4)="LADY DA":A$(5)="FRL.NUSS"
30 B$(1)="HAMMER":B$(2)="WASSERROHR":B$(3)="ROHRZANGE"
35 B$(4)="PISTOLE":B$(5)="STRICK":Z=0:GOTO205
40 T=1
50 GOSUB150:GOTO91
60 B(I)=K:I=I+1:IFI>5THEN200
70 GOSUB150
91 IFI=2ANDK=B(1)THEN70
92 IFI=3ANDK=B(2)ORK=B(1)THEN70
93 IFI=4ANDK=B(1)ORK=B(2)ORK=B(1)THEN70
94 IFI=5ANDK=B(1)ORK=B(2)ORK=B(3)ORK=B(4)THEN70
100 GOTO60
150 K=INT(RND(X)*5)+1:RETURN
200 V=B(1):W=B(2):X=B(3):Y=B(4):Z=B(5):RETURN
205 GOSUB40:PRINT"MOERDER MOERDER..■"
210 M$(1)=A$(V):M$(2)=A$(W):M$(3)=A$(X):M$(4)=A$(Y):M$(5)=A$(Z)
215 GOSUB40:PRINT"MOERDER...HAT...■"
220 TW$(1)=B$(V):TW$(2)=B$(W):TW$(3)=B$(X):TW$(4)=B$(Y):TW$(5)=B$(Z)
225 GOSUB40:PRINT"MOERDER...ZUGESCHLAGEN!!!■"
230 TA$(1)=C$(V):TA$(2)=C$(W):TA$(3)=C$(X):TA$(4)=C$(Y):TA$(5)=C$(Z)
240 FORL=1TO1000:NEXTL
    
```

```

290 REM**BILD 1**
300 POKE36879,8 :PRINT"□"
310 PRINT"Stad"SPC(3);"  " ;SPC(6);"  "
315 PRINTSPC(3);"  " ;SPC(6);"  "
320 PRINTSPC(5);"  " ;SPC(6);"  "
325 PRINTSPC(5);"  " ;SPC(6);"  "
330 PRINTSPC(3);"  " ;SPC(6);"  "
335 PRINTSPC(3);"  " ;SPC(6);"  "
340 PRINTSPC(3);"  " ;SPC(6);"  "
345 PRINTSPC(3);"  " ;SPC(6);"  "
350 PRINTSPC(12);"  "
355 PRINTSPC(3);"  " ;SPC(6);"  "
360 PRINTSPC(3);"  " ;SPC(7);"  "
365 PRINTSPC(10);"  "
370 PRINTSPC(10);"  "
375 PRINTSPC(8);"  "
380 PRINTSPC(7);"  "
385 PRINT"  " ;SPC(2);"  WER ERMORDETE  "
390 PRINTSPC(2);"  GRAF HINRICH VON  "
395 PRINTSPC(2);"  FALLLOBST ????  "
400 FORL=1TO3000:NEXTL
500 PRINT"□":POKE36879,25
510 PRINT"  WER WAR ES?"
520 FORI=1TO5:PRINT"  " ;PRINTA$(I);PRINT"  " :NEXTI
530 FORL=1TO500:NEXTL
540 PRINT"  WOMIT GESCHAH ES?"
550 FORI=1TO5:PRINT"  " ;PRINTB$(I);PRINT"  " :NEXTI
560 FORL=1TO500:NEXTL
570 PRINT"  WELCHER TATORT?"
580 FORI=1TO5:PRINT"  " ;PRINTC$(I);PRINT"  " :NEXTI
590 PRINT"  "
600 POKE198,0:WAIT198,1
700 TI$="000000"
1000 IFZ>10THENGOTO3000
1110 GOSUB15000:P=0:Z=Z+1
1120 PRINT"  " :INPUTW$
1130 IFW$=C$(1)THENGOSUB11000:GOTO1200
1140 IFW$=C$(2)THENGOSUB10100:GOTO1200
1150 IFW$=C$(3)THENGOSUB12000:GOTO1200
1160 IFW$=C$(4)THENGOSUB13000:GOTO1200
1170 IFW$=C$(5)THENGOSUB14000:GOTO1200
1175 IFW$="H"THEN4000
1180 PRINT"  DDRT GEH ICH NICHT HIN"
1190 FORL=1TO3000:NEXTL:GOTO1000
1200 IFW$=TA$(1)THENP=P+1
1220 PRINT"  " ;SPC(5);"  VERDACHT GEGEN":INPUT"  " ;V$
1230 IFV$=A$(1)THENGOSUB16000:GOTO1300
1240 IFV$=A$(2)THENGOSUB16500:GOTO1300
1245 IFV$=A$(3)THENGOSUB16500:GOTO1300
1250 IFV$=A$(4)THENGOSUB16000:GOTO1300
1260 IFV$=A$(5)THENGOSUB16000:GOTO1300
1265 IFV$="H"THEN4000
1270 PRINTSPC(5);"  NICHT BEKANNT"
1280 FORL=1TO500:NEXTL
1290 PRINT"□";SPC(5);"  " :GOTO1220
1300 IFV$=M$(1)THENP=P+1
1320 PRINT"  " ;SPC(5);"  TATWAFFE " :IFG=0THENGOTO1324
1322 INPUT"  " ;X$:GOTO1330
1324 INPUT"  " ;X$
1330 IFX$=B$(1)ORX$=B$(2)ORX$=B$(3)ORX$=B$(4)ORX$=B$(5)THEN1400
1335 IFX$="H"THEN4000
1340 PRINTSPC(5);"  NICHT BEKANNT"
1350 FORL=1TO500:NEXTL
1360 PRINT"□";SPC(5);"  "
1380 GOTO1320
1400 IFX$=TW$(1)THENP=P+1
1500 IFP>2THEN2000
1520 IFP>2THEN2000
1530 GOSUB10000:IFP=0THEN1580
1540 PRINT"  " ;SPC(3);"  NICHT SCHLECHT."
1550 PRINTSPC(3);"ES WAREN"
1560 PRINTSPC(3);P;" ANGABEN"
1570 PRINTSPC(3);"RICHTIG":GOTO1700
1580 PRINT"  " ;SPC(3);"KEIN VERDACHT"

```

programme

```
1590 PRINTSPC(3); "HAT SICH"
1600 PRINTSPC(3); "BESTAETIGT!"
1700 PRINT"XXXXXXXXXX♥";SPC(11-2); "←"
1750 FORL=1TO2000:NEXTL:GOTO1000
2000 IFZ>6THEN2500
2010 PRINT"□":POKE36879,106:PRINT"■"
2020 AB$=TI$
2030 PRINT"DU BIST EIN SEHR GUTERKRIMINALIST.DEIN SPUER"
2040 PRINT"OSINN ZEICHNET DICH AUS"
2050 PRINT"IN ";Z;" VERSUCHEN"
2060 PRINT"HAST DU DEN TAETER UEBERFUEHRT.DEINE ZEIT"
2070 PRINT"OFUER DIESE HERVORRAGENDE LEISTUNG:"
2080 PRINT"■";AB$
2090 PRINT"GRATULIERE,GRATULIERE"
2100 GOTO2800
2110 PRINTSPC(3); "GUT"
2500 PRINT"□":POKE36879,25:PRINT"■"
2510 AB$=TI$
2520 PRINT"GRATULIERE.DER MOERDERIST HINTER SCHLOSS"
2530 PRINT"UND RIEGEL.ER WIRD SEINER GERECHTEN STRAFE"
2540 PRINT"NICHT ENTGEHEN!"
2550 PRINT"DEINE ZEIT VOM ZEIT-":PRINT"PUNKT DES MORDES BIS"
2560 PRINT"ZUR AUFKLAERUNG:"
2570 PRINT"■";AB$
2580 PRINT"VERDAECHTIGUNGEN: ";Z
2600 PRINT"SO SOLL EINE NEUE MORD-"
2810 PRINT"SACHE AUFGEKLAERT WERDEN (J/N)?"
2820 GETY$
2830 IFY$="J"THENRUN
2840 IFY$="N"THENGOTO2860
2850 GOTO2820
2860 PRINT"□"
2870 PRINT"XXXXXXXXXXSO LONG COLUMBO !!"
2880 END
3000 PRINT"□":POKE36879,25
3100 PRINT"■DIE UHR IST ABGELAUFEN"
3110 PRINT"DER MOERDER STEHT HIN-TER DIR UND BERUEHRT"
3120 PRINT"DEINE SCHULTER."
3130 AB$=TI$
3140 PRINT"DU SIEHST....."
3150 FORL=1TO1000:NEXTL
3160 PRINTM$(1)
3170 PRINT"DER MOERDER ZEIGT DIR DEN TATORT....."
3180 FORL=1TO2000:NEXTL
3190 PRINT"■"TA$(1)
3200 PRINT"JA...UND ALS LETZTES IN DEINEM LEBEN SIEHST DUDIE TATWAFFE:"
3210 FORL=1TO2000:NEXTL
3220 PRINT"■";TW$(1); "■"
3230 PRINT"DEINE ZEIT: ";AB$
3240 PRINT"NEUES SPIEL (J/N)?"
3250 GETY$
3260 IFY$="N"THENGOTO3300
3270 IFY$="J"THENRUN
3280 GOTO3250
3300 PRINT"□"
3310 PRINT"XXXXXXXXXXRADIOS LITTLE DETEKTIV!"
3320 END
4000 PRINT"□":POKE36879,25:Z=Z+2
4100 GOSUB150
4120 J=K
4130 IFJ=1THENGOTO4100
4140 PRINT"■ L I B I FUER:"
4150 PRINT"■"M$(J)
4160 PRINT"■DIESE PERSON HIELT"
4170 PRINT"ICH ZUM ZEITPUNKT DESMORDES AUF:"
4180 PRINT"■"TA$(J)
4190 PRINT"ES BEFAND SICH BEI"
4200 PRINT"DIESER PERSON:"
4210 PRINT"■"TW$(J)
4215 PRINT"■DAS ALIBI IST PERFEKT"
4230 FORL=1TO5000:NEXTL
4240 GOTO1000
5000 PRINT"□"
5010 PRINT"■SPIELERKLAERUNG J/N?"
```

```

5020 FORL=1T020
5030 PRINT"SPIELERKLAERUNG J/N?"
5040 FORK=1T050
5050 NEXTK,L
5060 GETY$
5070 IFY$="N"THENRETURN
5080 IFY$="J"THENS100
5090 GOT05000
5100 PRINT" "
5110 PRINT"MOERDERJAGD     EIN SPIEL VON     "
5120 PRINT"    MARTIN BORMANN (C)1984"
5130 PRINT"5 VERDAECHTIGE     5 RAEUME"
5140 PRINT"5 MOEGLICHE TATWAFFEN"
5150 PRINT"-----"
5160 PRINT"GESUCHT WIRD DER MOER-DER DES GRAFEN!!!"
5170 PRINT"VERSUCHE DEN TAETER     ZU SCHNAPPEN,BEVOR ER"
5180 PRINT"DICH ERWISCHT!"
5190 PRINT"DURCH DRUECKEN DER 'H'TASTE BEKOMMST DU EIN"
5200 PRINT"ALIBI VON EINEM VER-     DAECHTIGEN.GEH ABER"
5210 PRINT"SPARSAM DAMIT UM"
5220 PRINT"BITTE TASTE DRUECKEN"
5230 POKE198,0:WAIT198,1
5240 RETURN
10000 PRINT" "
10020 PRINT"-----"
10030 FORI=1T010:PRINT" |";SPC(18);"| ":NEXTI
10040 PRINT"-----"
10050 RETURN
10100 POKE36879,93
10110 GOSUB10000
10120 PRINT" ";SPC(10);" "
10130 PRINTSPC(9);" "
10140 PRINTSPC(8);" "
10150 PRINT" ";SPC(9);" "
10160 PRINTSPC(3);" "
10170 PRINTSPC(3);" "
10180 PRINTSPC(3);" "
10190 PRINTSPC(3);" "
10200 PRINT" ";C$(2)
10300 RETURN
11000 POKE36879,238
11100 GOSUB10000
11110 PRINT" ";SPC(11);" "
11120 PRINTSPC(3);" "
11130 PRINTSPC(3);" "
11140 PRINTSPC(3);" "
11150 PRINTSPC(3);" "
11160 PRINTSPC(3);" "
11170 PRINTSPC(3);" "
11180 PRINTSPC(3);" "
11190 PRINTSPC(3);" "
11200 PRINT" ";C$(1)
11300 RETURN
12000 POKE36879,126
12100 GOSUB10000
12110 PRINT" ";SPC(10);" "
12120 PRINTSPC(7);" "
12130 PRINTSPC(6);" "
12140 PRINTSPC(6);" "
12150 PRINTSPC(2);" "
12160 PRINTSPC(2);" "
12170 PRINTSPC(2);" "
12180 PRINTSPC(2);" "
12190 PRINTSPC(2);" "
12200 PRINTSPC(2);" "
12250 PRINT" ";C$(3)
12300 RETURN
13000 POKE36879,25
13100 GOSUB10000
13110 PRINT" ";SPC(7);" "
13120 PRINTSPC(8);" "
13130 PRINTSPC(8);" "
13140 PRINTSPC(8);" "
13150 PRINTSPC(8);" "

```


Top Secret

für den VC-20 + 3K

Sie werden beauftragt, der GSG 9 eine geheime Laserkanone aus einem alten Labyrinth zu beschaffen. In diesem Labyrinth gibt es Geister und tiefe Brunnen, die Sie aber nur kurz sehen, weil Sie keine Taschenlampe dabei haben, und das letzte Streichholz verbrannt ist.

Berühren Sie einen Geist oder fallen Sie in einen Brun-

nen, so gibt der Computer Ihnen noch eine Chance, und stellt Ihnen eine Rateaufgabe. Wenn Sie diese Aufgabe auch nicht meistern, so haben Sie Ihr Leben verwirrt und das Spiel ist zu Ende.

Viel Glück!

Das Spiel wird mit dem Joystick gesteuert.

```

0 REM*****
1 REM*THOMAS VOGEL*
7 REM*****
8 SC=0:POKE36878,15:GOSUB6000:POKE37154,255:GOSUB7000
9 C=0:HR=0:GE=36877:POKE36879,8:GOTO28
27 REM***SPIELFELD***
28 SC=SC+1:PRINT"
29 RESTORE:GOSUB6000:A=7703:B=38423:D=29
30 A$(0)="
31 A$(1)="
32 A$(2)="
33 A$(3)="
34 A$(4)="
35 A$(5)="
36 A$(6)="
37 A$(7)="
38 A$(8)="
39 A$(9)="
40 A$(10)="
41 FOR I=1 TO 21:XX=INT(RND(1)*11):PRINTA$(XX):NEXT
42 PRINT"
43 PRINT"
44 REM***WAFFE***
45 AA=INT(RND(1)*462)+1:W=A+AA:V=B+AA
46 IFW<77050RW>8136THEN44
47 F=PEEK(W)
48 IFF=280RF=290RF=310RF=33THEN44
49 POKEW,30:POKEV,1
50 REM***GEISTER***
51 FOR I=1 TO INT(FA/2):BB=INT(RND(1)*462)+1:S=A+BB:T=B+BB
52 IFS<77050RS>8136THEN50
53 J=PEEK(S)
54 IFJ=280RJ=290RJ=300RJ=31THEN50
55 POKES,33:POKET,0
56 REM***BRUNNEN***
60 CC=INT(RND(1)*462)+1:Y=A+CC:M=B+CC
61 IFY<77050RY>8136THEN60
62 Q=PEEK(Y)
63 IFQ=280RQ=290RQ=300RQ=31THEN60
64 POKEY,31:POKEM,0:NEXT
153 IFHR=1THENRETURN
    
```

```

1100 REM***HAUPT-***
1101 REM***PROGRAMM***
1105 GOSUB3000:GOTO1250
1110 POKE37154,127:P1=PEEK(37152):P2=PEEK(37151):E=0
1120 IF(P2AND16)=0THENPOKEGE,240:POKEGE,0:GOSUB1240:A=A-1:B=B-1:GOSUB1210:GOTO1160
1130 IF(P2AND4)=0THENPOKEGE,240:POKEGE,0:GOSUB1240:A=A-22:B=B-22:GOSUB1210:GOTO1170
1140 IF(P2AND8)=0THENPOKEGE,240:POKEGE,0:GOSUB1240:A=A+22:B=B+22:GOSUB1210:GOTO1180
1150 IF(P1AND128)=0THENPOKEGE,240:POKEGE,0:GOSUB1240:A=A+1:B=B+1:GOSUB1210:GOTO1190
1160 IFE=29THENA=A+1:B=B+1:GOTO1250
1170 IFE=29THENA=A+22:B=B+22:GOTO1250
1180 IFE=29THENA=A-22:B=B-22:GOTO1250
1190 IFE=29THENA=A-1:B=B-1:GOTO1250
1191 IFE=30THEN1890
1192 IFE=31THEN3355
1193 IFE=33THEN4000
1200 GOTO1250
1210 IFA<768ITHENA=7680:B=38400
1220 IFA>8184THENA=8185:B=38905
1230 E=PEEK(A):RETURN
1240 POKEA,42:POKEB,0:RETURN
1250 POKEA,28:POKEB,1:GOTO1110
1260 POKEA+C,D:POKEB+C,8:GOSUB3000:RETURN
1350 REM**BILDWECHSEL*
1390 POKE36879,40
1900 FORI=255TO0STEP-.5:FORA=1TO5:NEXTA:POKE36876,I:NEXT:FORI=1TO2000:NEXT:POKE36879,8
2000 FORI=38TO150STEP.3:POKE36865,I:NEXT
2200 HR=1:GOSUB28
2300 FORI=150TO38STEP-.2:POKE36865,I:NEXT
2400 FA=FA+1:GOTO1105
3000 POKE36879,40:FORI=1TOLE:NEXT:POKE36879,8:RETURN
3350 REM***VERLIERER**
3351 REM***BRUNNEN***
3355 RESTORE
3358 POKE36879,40:GOSUB6000
3400 FORI=255TO0STEP-.5:POKE36874,I:NEXT:FORI=1TO3000:NEXT:POKE36879,8
3500 PRINT"DU SIE SIND IN EINEN TIEFEN BRUNNEN GEFALLEN."
3501 PRINT"JENN SIE JEMAND RAUSHOLEN SOLL,DRUECKEN SIE >J"
3502 PRINT"DER COMPUTER ENTSCHEIDET DANN OB SIE IM BRUNNEN VERHUNGERN."
3503 PRINT"#####:##%#####('###)"
3510 GETA#:IFA#=""THEN3510
3511 IFA#="A"ORA#="2"ORA#="."ORA#="/"THEN3510
3515 IFA#="J"THENLO=INT(RND(1)*3)
3516 IFA#="J"THENPRINT"DU SIE SIND BIS JETZT DER ERSTE DER VERHUNGERN WILL."
:GOTO3522
3520 IFLO=2THENPRINT"DU SCHADE SIE WAREN DEM COMPUTER UNSYMPATISCH"
3521 IFLO<>2THENGOTO3525
3522 PRINT"DU SIE ERREICHEN JEDOCH DAS "SC" FELD.":PRINT"DU AREADY.":END
3525 PRINT"DU HABEN SIE EIN GLUECK, DAS DER COMPUTER SIE MAG. SIE DUERFEN N OCHMAL."
3526 FORI=1TO6000:NEXT:RESTORE:GOSUB6000:GOTO2000
3550 REM***VERLIERER**
3551 REM***GEISTER***
4000 RESTORE
4100 POKE36879,40:GOSUB6000:POKE37154,255
4400 FORI=0TO255STEP3:POKE36874,I:NEXT:FORI=255TO0STEP-.5:POKE36874,I:NEXT
4450 FORI=1TO3000:NEXT:POKE36879,8
4500 PRINT"DER SINN DES SPIELS IST ES DIE GEHEIMWAFFE ZU FINDEN, UND NICHT
4501 PRINT"GEISTER UMZURENNEN.":GOSUB5000
4505 PRINT"ZUR STRAFE WERDEN SIE AN EINEM DUENNEN SEIL UEBER EINEN BRUNNEN
"
4506 PRINT"GEHAENGT. UM SICH ZU RETTEN,MUESSEN SIE MEINE ZAHL ZWISCHEN 1+
3WAEHLE
4507 POKE198,0:WAIT198,1:GETA#:IFA#=""THEN4507
4510 ZC=INT(RND(1)*3)+1
4515 IFA#="1"THENZU=1
4516 IFA#="2"THENZU=2

```

```

4517 IFA#="3"THENZU=3
4520 IFZC=ZUTHENPRINT"WBRAVO SIE HABEN IHR WGLUECK FEST IM GRIFF.":GOTO4550
4530 GOTO4560
4550 PRINT"WZUR BEHLONUNG DUERFEN WSIE NOCHMAL.":GOSUB5200:RESTORE:GOSUB6000:GO
T02000
4560 PRINT"WWARUM HABEN SIE DIE WZAHL"ZU"GEWAHLT UND W NICHT "ZC".
4565 PRINT"WSCHADE DER GEIST WAR WIHR VERHAENGNISS.":GOTO3522
5000 X=1:Y=.5
5001 POKE7988+X+Y,33:POKE7988+X+Y+1,28
5002 POKE38708+X+Y,1
5003 X=X+1
5005 FORI=1T0200:NEXT
5006 POKE7988+X-Y,27
5007 POKE38708+X-Y,0
5016 IFX=21THENGOTO5100
5017 GOTO5001
5100 X=21
5101 POKE7988+X-Y,33
5102 POKE38708+X-Y,1
5103 X=X-1
5105 FORI=1T0200:NEXT
5106 POKE7988+X+Y,27
5107 POKE38708+X+Y,0
5116 IFX=1THENRETURN
5117 GOTO5101
5200 REM**BEWEGUNG
5201 REM***GEIST
5210 X=21
5220 POKE7988+X-Y,28
5230 POKE38708+X-Y,1
5240 X=X-1
5250 FORI=1T0200:NEXT
5260 POKE7988+X+Y,27
5270 POKE38708+X+Y,0
5280 IFX=1THENRETURN
5290 GOTO5220
5500 REM***DATA'S***
6000 FORI=7384T07511:READA:POKEI,A:NEXT
6050 DATA0,0,0,0,0,0,0
6100 DATA0,126,90,255,255,102,60,24,255,195,165,153,153,165,195,255
6110 DATA,,96,248,255,248,96,,24,36,66,129,129,66,36,24,,,,,,,,
6120 DATA28,62,42,62,62,62,62,42
6130 DATA0,0,0,0,0,0,0,0,31,32,85,157,149,0,15,16,255,0,77,73,109,9,201,63
6140 DATA224,16,136,8,136,8,176,192,0,0,0,0,0,1,2,2,32,64,128,0,0,0,0
6150 DATA2,2,26,38,35,63,59,47,59,47,59,47,59,47,61,47
6160 DATA0,0,0,24,24,0,0,0
6399 POKE36869,255:RETURN
7000 POKE36879,8:POKE36869,240
7001 A#="<<<<< TOP SECRET >>>>>":PRINTA#
7002 PRINT"WSIE WERDEN BEAUFTRAGT WDER GSG 9 EINE GEHEIMEWLASERKANNONE IN
7003 PRINT"WEINEM ALTEN LABYRINTH WSICHERZUSTELLEN."
7004 GOSUB8000
7009 PRINTA#
7010 PRINT"WACHTUNG:WDA SIE KEINE WTASCHENLAMPE MITNEHMENWDUERFEN,KOENNEN SIE
"
7011 PRINT"WDIE GEISTER UND DIE WBRUNNEN NUR AM EIN- WANG KURZ SEHEN."
7012 GOSUB8000
7020 PRINTA#
7030 PRINT"WSTEUERUNG:"
7031 PRINT"WGESTEUEBT WIRD MIT DEMW JOYSTICK
7034 INPUT"WLEVEL (0-9)";LX:LE=5000-(LX*500)
7050 POKE36869,255:RETURN
8000 PRINT"W >>F7<<<"
8001 GETB#:IFB#=""THEN8001
8002 IFB#="|"THENRETURN
8003 IFB#<>"|"THEN8001
READ).

```

Tips & Tricks

für den Commodore 64

Zwei Datasetten am C-64

Mit Hilfe der nachfolgenden Schaltung können zwei Datasetten am C-64 betrieben werden. So können z.B. Dateien wesentlich einfacher verwaltet werden.

Stellen Sie sich z.B. ein Programm vor, welches Namen und Adressen in Files auf Band ablegt. Die Files wollen wir A,B,C,D und E nennen.

Der Computer kann nun ein Eingangsmenue erstellen, von dem der Benutzer ein Kriterium auswählen kann.

Nehmen wir als Beispiel den File "C". Der Computer sucht nun das Band nach File "C" ab, findet er diesen, wird er in den Arbeitsspeicher geladen. Nun kann ein Problem auftreten: Wenn neue Daten eingegeben werden, oder Daten geändert werden, wird die Datei dementsprechend länger. So kann es passieren, daß zwischen File "B" und File "D" der Platz für den neuen File "C" nicht mehr ausreicht. Die Folge wäre das Überschreiten des Files "D".

Alternativ dazu, kann man nun das Mastertape und die Dateien auf zwei Datenträger aufsplitten.

Da dies jedoch sehr zeitaufwendig ist und Ungenauigkeiten der alten Bandposition hervorrufen kann, empfiehlt sich das Arbeiten mit zwei Datasetten.

So kann verhindert werden, daß wichtige Daten verlorengehen oder Dateien doppelt belegt werden.

Abb.1 zeigt das vom Transistor gesteuerte Relais. Zur Ansteuerung muß das Ausgangsregister high oder low gepoked werden.

Die Steuerleitung wird vom

PIN C des C-64 User-Ports abgegriffen, die Stromversorgung erfolgt über PIN 10 (+) und PIN A (4).

Die Kassettensteuerung wird vom Kassettenport abgenommen und in den Leitungen Motor, Read und Switch umgeschaltet (Abb.2)

Bestückungsliste:

Widerstände:

1-1K

1-10K

Halbleiter:

2-1N4001

1-BC108 oder Vergleichstyp

Kapazitäten:

1 Elektrolytkondensator

68 µF/120V

Zusatzbauteile

1-3-poliges Wechselrelais

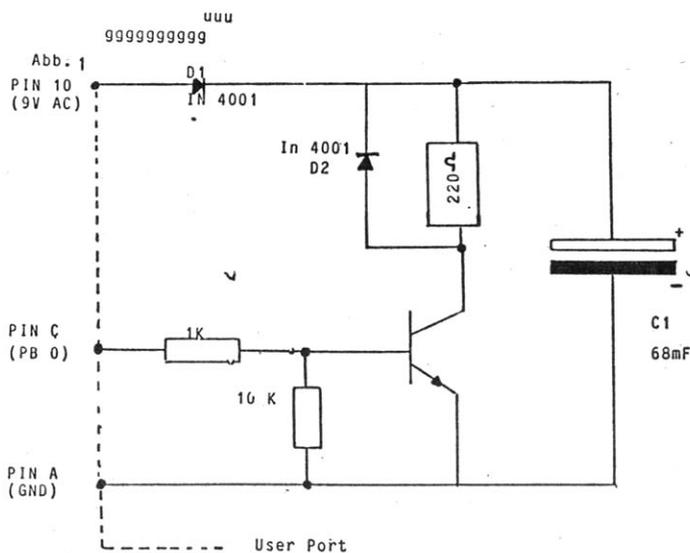
1-User Port Stecker

2-Datasetten

2-Datasettenstecker

1-Datasettenbuchse

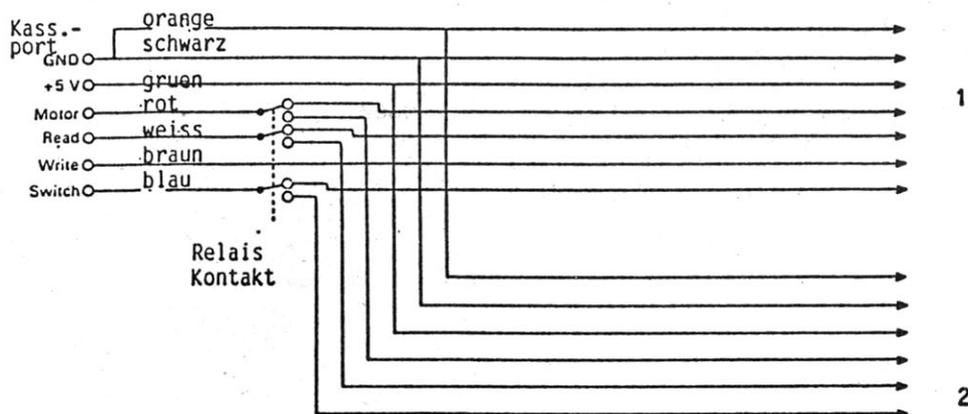
(sr)



Treibersoftware

```

5 Input"1=Tape 1:0=Tape 2";x:if x< 0 or x> 1Then 5
10 CB = 56576
20 C1 = CB + 1           60 POKE C3,63
30 C2 = CB + 3           70 POKE C4,7
40 C3 = 0                80 POKE C2,255
50 C4 = 1                90 POKE C1,X
    
```



Renumber

für den C-64

Tips & Tricks

Dieses Basic-Lade-Programm erzeugt nach dem Start durch RUN ein Maschinenspracheprogramm auf der Diskette (oder Kassette nach Änderung der Sekundär-Adresse) unter dem Namen "Renumber".

Wird dieses Programm absolut (mit Load "Renumber", 8,1) in den Rechner geladen, so besteht nun für den Anwender die Möglichkeit, ein sich im Speicher befindliches Basic-Programm neu zu nummerieren.

Dies geschieht mittels SYS 50076, st, sw - wobei st die Anfangsadresse und sw die Schrittweite bedeuten.

Das Programm berücksichtigt dabei die Sprungbefehle GOTO, GOSUB ON x GOTO a,b,..., If x THEN a, sowie den RESTORE a Befehl.

Es sollte lediglich darauf geachtet werden, daß keine Adressen, die größer als 63999 sind erzeugt werden, da der Rechner diese nicht mehr anerkennt.

Auch sollte das Programm "Renumber" vor dem Schreiben eines neuen Programms oder dem Laden eines Programms, welches neu nummeriert werden soll, geladen werden.

```
100 OPEN1,8,1,"@:RENUMBER":PRINT#1,CHR$(0)CHR$(192);:A=0
110 FORI=49152TO50101:READQ:PRINT#1,CHR$(Q);:A=A+Q:NEXT:CLOSE1
120 IFAC>137072THENPRINT"#CHTUNG FEHLER IN DATAZEILEN"
130 DATA 32,253,174,32,138,173,32,247,183,165,20,133,176,133,251,165
140 DATA 21,133,177,133,252,32,253,174,32,138,173,32,247,183,165,20
150 DATA 133,150,165,21,133,159,169,0,133,178,133,167,169,196,133,179
160 DATA 169,8,133,168,160,1,177,167,208,5,200,177,167,240,79,160
170 DATA 1,177,167,133,169,200,177,167,133,170,200,177,167,170,200,177
180 DATA 167,160,1,145,178,136,138,145,178,24,165,178,105,2,133,178
190 DATA 165,179,105,0,133,179,160,3,165,176,145,167,200,165,177,145
200 DATA 167,56,165,169,233,1,133,167,165,170,233,0,133,168,24,165
210 DATA 176,101,150,133,176,165,177,101,159,133,177,76,52,192,169,1
220 DATA 133,176,169,0,133,177,133,166,169,5,133,165,160,0,177,165
230 DATA 201,137,240,62,201,140,240,58,201,141,240,54,201,167,240,50
240 DATA 201,0,208,37,160,0,177,176,170,208,7,200,177,176,208,1
250 DATA 96,136,200,177,176,133,177,134,176,24,165,165,105,5,133,165
260 DATA 165,166,105,0,133,166,76,156,192,230,165,208,193,230,166,76
270 DATA 158,192,230,165,208,2,230,166,177,165,201,32,240,244,166,165
280 DATA 134,249,166,166,134,250,201,48,144,166,201,58,176,162,132,253
290 DATA 132,254,162,1,24,233,47,133,2,165,254,133,171,165,253,10
300 DATA 38,171,10,38,171,101,253,133,253,165,171,101,254,133,254,6
310 DATA 253,38,254,165,253,101,2,133,253,144,2,230,254,230,165,208
320 DATA 2,230,166,177,165,201,48,144,7,201,58,176,3,232,144,196
330 DATA 134,181,132,178,169,196,133,179,160,0,177,178,197,253,208,7
340 DATA 200,177,178,197,254,240,17,24,165,178,105,2,133,178,165,179
350 DATA 105,0,133,179,201,200,208,222,160,0,132,2,132,171,56,165
360 DATA 179,233,196,133,179,24,102,179,102,178,24,165,2,101,150,133
370 DATA 2,165,171,101,159,133,171,56,165,178,233,1,133,178,165,179
380 DATA 233,0,133,179,176,228,24,165,2,101,251,133,2,165,171,101
390 DATA 252,133,171,56,165,2,229,150,133,2,165,171,229,159,133,171
400 DATA 201,39,144,13,208,6,165,2,201,16,144,5,162,5,76,240
410 DATA 193,165,171,201,3,144,13,208,6,165,2,201,232,144,5,162
420 DATA 4,76,240,193,165,171,208,6,165,2,201,100,144,5,162,3
430 DATA 76,240,193,165,2,201,10,144,5,162,2,76,240,193,162,1
440 DATA 134,255,138,56,229,181,133,181,208,3,76,226,194,48,110,24
450 DATA 165,169,101,181,133,169,133,247,165,170,105,0,133,170,133,248
460 DATA 160,0,177,247,164,181,145,247,56,165,247,233,1,133,247,176
470 DATA 2,198,248,165,247,197,249,208,231,165,248,197,250,208,225,165
480 DATA 176,133,247,165,177,133,248,160,0,177,247,208,8,170,200,177
490 DATA 247,240,28,136,138,24,101,181,145,247,200,177,247,105,0,145
500 DATA 247,136,177,247,170,200,177,247,133,248,134,247,76,55,194,24
510 DATA 165,45,101,181,133,45,144,2,230,46,76,226,194,198,181,165
520 DATA 181,73,255,133,181,56,165,165,229,181,133,247,165,166,233,0
530 DATA 133,248,164,181,177,247,160,0,145,247,230,247,208,2,230,248
540 DATA 165,247,197,169,208,236,165,248,197,170,208,230,56,165,169,229
550 DATA 181,133,169,176,2,198,170,56,165,45,229,181,133,45,176,2
560 DATA 198,46,165,176,133,247,165,177,133,248,160,0,177,247,208,8
570 DATA 170,200,177,247,240,28,136,138,56,229,181,145,247,200,177,247
580 DATA 233,0,145,247,136,177,247,170,200,177,247,133,248,134,247,76
590 DATA 186,194,165,255,201,4,240,44,201,2,240,104,201,3,240,68
600 DATA 201,1,240,118,162,0,56,165,2,233,16,133,247,165,171,233
610 DATA 39,133,248,144,12,232,165,247,133,2,165,248,133,171,76,246
620 DATA 194,32,141,195,162,0,56,165,2,233,232,133,247,165,171,233
630 DATA 3,133,248,144,12,232,165,247,133,2,165,248,133,171,76,22
640 DATA 195,32,141,195,162,0,56,165,2,233,100,133,247,165,171,233
650 DATA 0,133,248,144,12,232,165,247,133,2,165,248,133,171,76,54
660 DATA 195,32,141,195,162,0,56,165,2,233,10,133,247,144,8,232
670 DATA 165,247,133,2,76,86,195,32,141,195,162,0,198,2,48,4
680 DATA 232,76,108,195,32,141,195,165,249,133,165,165,250,133,166,160
690 DATA 0,177,165,201,44,240,3,76,158,192,76,226,192,138,160,0
700 DATA 24,105,48,145,249,230,249,208,2,230,250,96,160,13,185,165
710 DATA 0,153,32,203,136,208,247,32,0,192,160,13,185,32,203,153
720 DATA 165,0,136,208,247,96
```

CBM 64 Kontrolltaste

Manche Commodore 64 Besitzer werden vielleicht noch nicht wissen, daß die CTRL-Taste gemeinsam mit den alphabetischen Tasten benutzt werden kann, um den CHRS Code von 1 bis 26 zu erzeugen.

Beispiel:
CTRL und A = CHRS (1)
CTRL und B = CHRS (2)
usw.

Diese Abfragen können entweder im Direktmodus verwendet werden, oder in einem Programm.

Leserbriefe

Ich habe einen SX-64 portable Computer. Da der eingebaute Monitor teilweise sehr schlecht zu lesen ist, möchte ich mein Gerät nun an einen Farbfernseher anschließen. Am SX-64 ist jedoch kein TV-Ausgang vorhanden. Können Sie mir weiterhelfen?

P. Burg, Offenbach

Redaktion:

Besorgen Sie sich einen TV-Modulator von einem alten VC-20. Das Gerät kann das Composite-Video-Signal des C-64 in das PAL-Normsignal umwandeln. Lediglich die DIN-Buchse des VC-20 Modulators muß gegen den 7-poligen Video-stecker des SX-64 getauscht werden. Die Änderung in der Verdrahtung des Steckers entnehmen Sie dem Handbuch.

Seit etwa 6 Wochen bin ich stolzer Besitzer eines C-64. Da ich mich hauptsächlich für Musik interessiere, habe ich bereits zwei Synthesizer-Programme im Einsatz. Um einen besseren Klang zu erzielen, habe ich nun den C-64 an meine Stereoanlage angeschlossen. Mit Verwunderung mußte ich nun feststellen, daß ein sehr starkes Rauschen zu hören ist. Liegt das an meinem Computer, oder etwa an der Software?

C. Kirst, Berlin

Redaktion:

Das Rauschen wird durch den SID-Chip verursacht. Der SID hat ein gewisses Eigenrauschen, dies läßt sich auch nicht unterdrücken. Bei guten Synthesizerprogrammen kann mit Hilfe der Oszillatoreneinstellung etwas Abhilfe geschaffen werden.

Kürzlich habe ich einen VC-20 zum Geburtstag geschenkt bekommen. Leider komme ich damit überhaupt nicht klar. Mir ist ein Rätsel, wie ich Programme, die ich eifrig aus Zeitschriften abtippe, auf Kasette speichern kann.

B. Stern, Hünfeld

Redaktion:

Zunächst einmal gibt es 2 Möglichkeiten des Abspeicherns. Programme werden mit **Save "Programmname"** abgesaved, die Datasette wird automatisch gesteuert. Daten werden mit **Open 1,1,1,"-**

Programmname" abgespeichert. Im übrigen gibt das Handbuch darüber Aufschluß.

Für meinen VC-20 habe ich mir kürzlich ein 64K-RAM zugelegt. In der Hoffnung, daß ich C-64 Programme auf meinem VC-20 laufen lassen kann. Erstaunt mußte ich feststellen, daß nicht eines der Programme mit der 64K-Erweiterung lief. Warum funktionierte das nicht und wie bekomme ich diese Programme zum Laufen.

K. Reuter, Kassel

Redaktion:

Der alleinige Kauf einer 64K-Erweiterung bringt noch lange nicht die Möglichkeiten, die ein C-64 besitzt. Es bringt genaugenommen noch nicht einmal die gleiche Speicherleistung wie ein 64'er. Die meisten dieser Erweiterungen sind in 8K Blocks aufgesplittet, von denen immer nur ein Block angesprochen wird. Niemals 64KB auf einmal. Anders als beim C-64, bei dem die vollen 64K auf einmal zur Verfügung stehen, liegen bei einer 64K-Erweiterung des VC-20 immer 56K hinter dem eigentlichen Arbeitsspeicher versteckt. Außerdem stimmen die Sprungadressen für das Kernel-ROM sowie die PEEK-Abfragen nicht überein. Und selbst wenn Sie all dieses ändern könnten, so würde Ihr Programm spätestens beim ersten Aufruf eines Sprites oder Ansprechen der SID nicht mehr funktionieren.

Als erstmaliger Leser Ihrer Computerzeitschrift darf ich Sie beglückwünschen, eine aktuelle, frische und mit ausreichenden Programmisten ausgestattete Zeitschrift gemacht zu haben, die zu einem "Spitze"-Preis erhältlich ist.

Nachdem Sie für Verbesserungsvorschläge zugänglich sind, führe ich nachstehend einige auf:

1. Inhaltliche Aufstockung: Programmisten für Lern- und Lehrprogramme für Schüler, z.B. Vokabel-Lernprogramm, Deutsch- und Mathematik-Lernprogramm, etc.

Denken Sie daran, daß sich der Kreis der Interessenten erweitert um die vielen "Normalverbraucher".

2. Die Lesbarkeit von wiedergegebenen Programmisten durch bessere Drucktechnik verbessern. Graphische Steuer-

zeichen mit reverser Darstellung waren in Nr. 37/84, Seiten 10, 18, 19-23, 36-39, usw., nur zu erraten, besonders, wenn es um verkleinerte Wiedergabe ging.

3. Anfänger-Tips für Selbstprogramme etc.

4. Leser-Ecke mit Beratungs-Antworten der Redaktion

Ich würde mich freuen, wenn Sie die Anregungen realisieren würden.

H. Mentzel, Erlangen

Redaktion:

Stellvertretend für eine Flut von Leserbriefen haben wir das Schreiben von Herrn Mentzel abgedruckt.

Wir werden uns in Zukunft bemühen, die Softwareprogramme mit mehr Anwendungen zu füllen. Da die "Compute mit" eine Zeitschrift für den Leser ist, können auch Sie "Ihre" Zeitschrift mitgestalten. Also dann, schicken Sie uns Ihre selbstgeschriebenen Programme, bei Abdruck gibt's ein gutes Taschengeld. Der Abdruck der Steuerzeichen, da haben Sie völlig Recht, war in der Erstaussgabe wirklich nicht gut lesbar. Doch wir bemühen uns auch weiterhin, klare und deutliche Listings abdrucken, in den Folgeausgaben von "Compute mit" war es bereits wesentlich besser, oder?

Eine Leser-Ecke haben wir bereits eingerichtet, wir werden selbstverständlich versuchen, jede Zuschrift zu beantworten.

Anfänger-Tips gibt's bei uns jede Menge, unsere Serie "Logik in Selbstgestrickten" schließt ebenfalls daran an.

Da wir die "Compute mit" für Sie gestalten wollen, fordern wir Sie auch weiterhin auf: Nehmen Sie Ihren Kugelschreiber und schicken Sie uns Ihre Anregungen, Tips, Programme und selbstverständlich auch Kritiken.

Wir freuen uns über jede Zuschrift!

(sr)



Heilborn/Talbott Das Commodore 64 Anwenderhandbuch

388 Seiten Wissenswertes über den Commodore 64 – für Einsteiger leicht verständlich und für Benutzer voller praktischer Tips und Tricks.

Dieses durchgehend illustrierte Buch vermittelt alles nötige Wissen für den Umgang mit dem C-64 und seinen Zusatzgeräten.

Dem Anfänger bietet es

- eine Anleitung für den Aufbau und den Betrieb des C-64 und seiner Peripherie
- eine vollständige Einführung in die C-64 Basic-Programmierung
- eine Darstellung der besonderen Grafik- und Tonfähigkeiten des C-64.

Dem fortgeschrittenen Benutzer dient es als unschätzbare Arbeitsmittel und Nachschlagewerk. Es enthält eine detail-



lierte Übersicht aller Basic-Statements und Funktionen, ein ausführliches Verzeichnis der Speicher-Adressierung zusammen mit einer Erklärung, was jede adressierbare Speicherzelle leistet und wie sie arbeitet und einen besonderen Abschnitt zur fortgeschrittenen Color-Grafik und Sprite-Grafik.

Im Anhang finden sich Speicherbelegung, Einsprungadressen ins Betriebssystem für die Maschinenprogrammierer, Anschlußbelegungen der Steckbuchsen, Umrechnungstabellen hexadezimal zu dezimal und zurück, eine ausführliche ASCII-Tabelle und ein alphabetisches Verzeichnis der BASIC-Befehle.

Die Daten: MC Graw-Hill Bookcompany, Hamburg 1984, 388 S. kart., DM 39,80, ISBN 3-89028-016-1

Schacht BASIC – Schritt für Schritt

Die Programmierung eines Computers wird heute noch zum Teil als Geheimwissenschaft betrachtet, die nur wenige lernen können. Die Zeit dürfte aber nicht fern sein, in der Kenntnisse über die Programmierertechnik zum Allgemeinwissen gehören werden.

BASIC ist ohne Zweifel nicht nur die am weitesten verbreitete, sondern auch die am leichtesten erlernbare Sprache für die Programmierung von Computern. Diese Sprache wird in diesem Buch auch für Schüler und Laien verständlich erklärt.

Auch wer bereits das Handbuch eines Computers besitzt, wird in diesem Buch noch viele ergänzende Hinweise für die eigene Programmierarbeit finden, denn es wird darin nicht nur die Wirkung der BASIC-Befehle erläutert, sondern erklärt, wie aus den Befehlen ein nützliches Programm entsteht.

In die Didaktik ist das Buch anders aufgebaut als die meisten sonst bekannteren

BASIC-Lehrbücher, bei denen vorausgesetzt wird, daß der Leser einen Computer vor sich hat, um durch Probieren zu lernen. In diesem Buch werden auch die Kleinigkeiten, die man beachten muß, systematisch und vollständig in der richtigen Reihenfolge erklärt.

Leider ist BASIC noch nicht in allen Einzelheiten genormt; darum werden in den Beispielen nur die Befehle verwendet, die im wesentlichen einheitlich bei allen BASIC-Dialekten zulässig sind. Geringfügige Abweichungen werden ausdrücklich erklärt.

Die Programmbeispiele selbst sind so ausgewählt, daß sie in eigene nützliche Programme eingebaut werden können, wodurch der Schritt von der bloßen Kenntnis der Sprachelemente zum eigenen Programmieren erleichtert werden soll. (Aus dem Vorwort.) Die Daten: Humboldt (ht), 1. Aufl. 1984, 160 S., ISBN 3-58166456-9

Heilborn/Teilbott Das Anwenderhandbuch zum VC-20

Es vermittelt alles nötige Wissen für den Umgang mit dem VC-20 und seinen Zusatzgeräten. Die VC-20-Basic-Programmierertechnik und der ganze Bereich der Farbgrafik und der Tonerzeugung werden genau beschrieben. Darüber hinaus gibt das Buch praktische Tips zur Programmierung eigener elektronischer Unterhaltungsspiele.

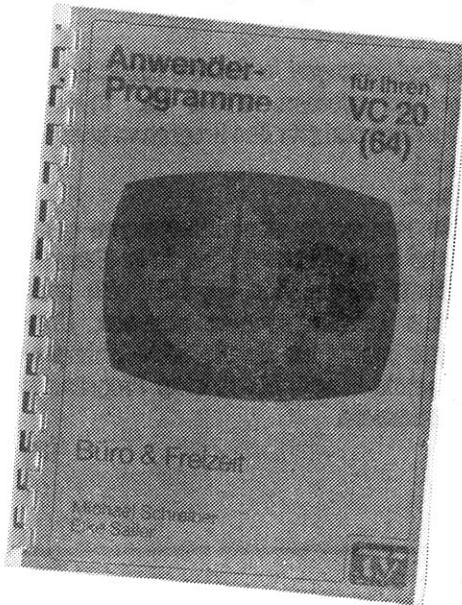
Es eignet sich sowohl für den Anfänger als schnelle Einführung als auch für den erfahrenen Anwender als Nachschlagewerk.

Aus dem Inhalt:

Vorstellung, Bedienung und Programmierung des VC-20 – Spielsteuerungen – Grafiken – Klänge – Peripheriegeräte – Systemarchitektur – Speicherbelegung – Fehlermeldungen – Basic-Statements und -Funktionen.

Die Daten: McGraw Hill Book Company, 388 S., DM 32,00, ISBN 3-89028-004-8

Schreiber/Saller: Anwenderprogramme für Ihren VC-20/64



Das Buch zielt darauf ab, die Möglichkeiten des VC-20 zu demonstrieren und anzuwenden. Im wesentlichen werden Spiele und kommerzielle Anwendungen programmiert. Die Palette reicht von Car-Race bis zur Finanzbuchhaltung. Zu jedem Programm gibt es kurze Beschreibungen, auffallend die gute Lesbarkeit der Steuerzeichen in den Listings. Ein Buch für den fortgeschrittenen Computer-Anwender. Die Daten: Luther-Verlag, 1984, 135 S., ISBN-3-88707-029-1



Brückmann: Der Commodore und der Rest der Welt

Ein wirklich tolles Buch für alle Bastler und Hobbyelektroniker. Hier werden technische Anwendungen mit dem C-64 beschrieben und mit Schaltbildern dokumentiert. Das Spektrum der Anwendungen reicht von dem Aufbau eines Computers bis zur Steuerung von Motoren mit dem C-64. Dabei wurde der Erläuterung-

stext ziemlich einfach gehalten, sodaß auch ein Laie nicht nur Bahnhof versteht. Damit auch alles gleich gesteuert werden kann, fehlen auch die Softwarelistings nicht. Die Daten: Data Becker, 1984, 229 S., ISBN 3-89011-015-0

(sr)

Computer im Fernsehen

Liebe Freunde von
COMPUTE MIT....!

Alles, was sich im Fernsehen mit dem unerschöpflichen Thema Homecomputer beschäftigt, nennen wir Ihnen jeweils an dieser Stelle, damit Sie genau zur rechten Zeit Ihren Fernseher einschalten können und nichts wesentliches versäumen. Ein toller **COMPUTE MIT...-Service** – denn **COMPUTE MIT...-Leser** wissen Bescheid und können mitreden.

05.10.84
Mikroprozessoren-
Microcomputer Teil I
3. Computer in der
Dienstleistung

SW3
21.15 h

08.10.84
Mikroprozessoren
3. Computer in der
Dienstleistung

ZDF
16.04 h

15.10.84
Mikroprozessoren
4. Struktur eines
Computers

ZDF
16.04 h

Süddeutscher Rundfunk

S III ab 21. 9. 1984,
jeweils Freitag 21.15 Uhr
und Samstag 17.30 Uhr (Wh.)

SENDEZEITEN: Westdeutsches Fernsehen

WDF III ab 9. 9. 1984,
jeweils Sonntag 15.30 Uhr

TV-Ecke

Bayerischer Rundfunk

BR III ab 19. 9. 1984,
jeweils Mittwoch 16.00 Uhr
und Freitag 9.00 Uhr (Wh.)

Zweites Deutsches Fernsehen

ZDF ab 24. 9. 1984,
jeweils Montag 16.04 Uhr
und Samstag 11.15 Uhr (Wh.)

Sprachsynthesizer für C-64

Unterhalten Sie sich mit SAM, einem der inzwischen zahllosen Sprachsynthesizern für den Commodore 64.

SAM, das steht für Software Automatic Mouth. Dieses Sprachprogramm der Firma Don't Ask Software auf Diskette ist wohl eines der vielseitigsten auf dem Markt. Der Satzsatz ist quasi unbeschränkt, mit richtig gesetzten Parametern kann SAM fast alles in die digitale Sprache umsetzen. Dabei ist der Vorgang ziemlich einfach, ein "Say"-Kommando mit dem entsprechenden Wort läßt den SID ansprechen. Geben Sie

z.B. Say "Hello", so hören Sie ein digitalisiertes, klar verständliches "Hello" von SAM. Mit der Eingabe von englischen Wörtern lassen sich ohne viel Parameter die meisten Experimente vollziehen. Bei deutschen Wörtern wird es schon schwieriger, sehr viel Parameter müssen gesetzt werden. So kann z. B. die Schnelligkeit, Tonlage usw. geändert werden. Um ein deutlicheres Ergebnis zu bekommen, besteht die Möglichkeit hinter Silben Registerwerte (1-8) zu setzen. Dies bekommt man durch Ausprobieren schnell in den Griff.

Mit der Option "Reciter" können Band-

aufzeichnungen in digitale Sprache umgesetzt werden, wobei ca. 11Kbyte Speicherplatz für das Programm benötigt wird.

Fazit:

Ein Programm, das Experimente geradezu herausfordert. Da kein Sprachschatz vorgegeben ist, läßt sich auch keine Leistungsgrenze absehen. Übrigens macht es Spaß, SAM-Unterprogramme in selbstgestrickte Programme einzubauen.

(sr)

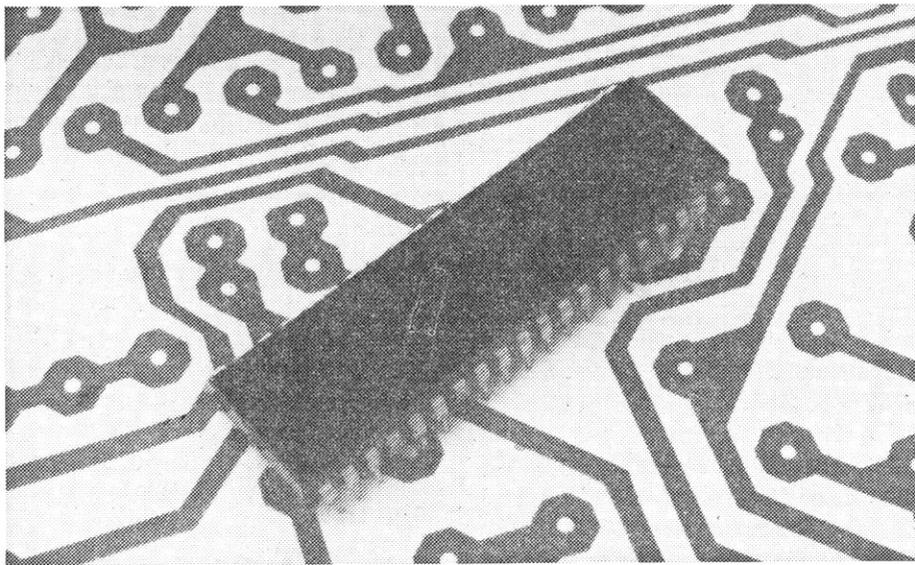
BASS-Compiler

Der neue EXBASIC-COMPILER von Interface Age ist ein sehr leistungsfähiges Werkzeug zur effizienten Programmierung: Aber – was ist das eigentlich ein Compiler?

Vereinfacht ausgedrückt, übersetzt ein Compiler (sprich: Kompeiler) ein in einer Hochsprache (z.B. BASIC) geschriebenes Programm in einen anderen, vom Computer leicht ausführbaren Code. Dies kann Maschinen- oder Assembler oder auch irgend ein anderer Code sein. In jedem Fall ist es einer, der vom 'Herz' des Computers, dem Mikroprozessor, leichter und somit schneller verstanden werden kann als die Hochsprache.

Der EXBASIC-COMPILER erzeugt aus Ihrem Standard-Basic- bzw. EXTENDED-BASIC-Programm 6502-Maschinencode, der direkt vom Mikroprozessor 6502 (oder 6510,6509), der in jedem Commodore Computer vorhanden ist, verarbeitet werden kann. Der Hauptvorteil bei der Benutzung eines Compilers liegt also darin, daß die compilierten Programme wesentlich schneller im Ablauf sind.

Der Hauptnachteil sei ohne allerdings auch nicht verschwiegen. Der Vorgang des compilierens ist äußerst komplex und dauert folglich einige Zeit. Dies kann zwischen einigen Minuten bis im Ex-



tremfall einigen Stunden liegen. Der EXBASIC-COMPILER benötigt insgesamt 4 Durchläufe durch ein Basic-Programm, bevor es komplett in Maschinencode umgewandelt ist. Andere Compiler sind in dieser Hinsicht weniger aktiv, dafür aber zum Teil im Compilerprozeß schneller (nicht im Ablauf!!).

Interface-Age hat seine Compiler so ausgelegt, daß er fast jedes Basic-Programm ohne Änderungen verarbeiten kann. Dennoch hat jeder Compiler seine Grenzen in dieser Hinsicht, einen, der jedes (100%) Basic-Programm problemlos compilieren kann, gibt es nicht. Es kann Ihnen also passieren, daß gewisse Basic-Befehle oder Strukturen nicht ver-

arbeitet werden. Da helfen nur noch Änderungen an Ihrem Basic-Programm. Das Kapitel 'UNTERSCHIEDE COMPILER/INTERPRETER', im 40 seitigem Handbuch, unterstützt Sie dabei.

Selbstverständlich ist, daß Sie fundierte Basic-Kenntnisse mitbringen müssen, um den Compiler in seiner Leistungsfähigkeit ausnutzen zu können. Wenn Sie hier noch Lücken haben, empfiehlt sich die Anschaffung des Buches 'BASIC KURS BEGINNER', ISBN 3-88623-015-5 das vom gleichen Hersteller erschienen ist.

Bezugsquelle: Exbasic-Compiler für C-64, Interface-Age-Verlag-GmbH in München (tm)

Textverarbeitung

für den Commodore 64

Das Programm ist einsetzbar in Verbindung mit dem Commodore 64, der Floppy VC 1541 und einem Matrix Printer (z.B. dem Commodore MPS 801).

Es werden cirka 14 KByte Speicherplatz belegt.

Nach dem Laden und Starten meldet sich 'Textverarbeitung' mit

dem Hauptmenue. Von hieraus werden die Unterprogramme und das Diskettenmenue angewählt. Eine Rückkehr ins Hauptmenue von den einzelnen Unterprogrammen ist durch Eingabe der 'Pfeil nach oben'-Taste möglich. Nun die Erläuterungen zu den einzelnen Unterprogrammen:

1. Texteingabe:

Das Programm fragt ab, ob neuer Text eingegeben oder an schon geschriebenen Text angehängen werden soll. Hier muß nur dann F1 gedrückt werden, wenn der Text von Diskette eingeladen wurde und nun weitergeschrieben werden soll. Wurde die Texteingabe nur unterbrochen, so setzt sich das Programm selbst einen Merker. Dann wird die Zeilennummer abgefragt, ab welcher der Text zu speichern ist. Bei neuem Text sollte eine 1 eingegeben werden, ansonsten die Zeilennummer (+1) bei welcher der schon geschriebene Text endet. Nun kann mit der Texteingabe begonnen werden. Die Cursorsteuertasten funktionieren im Eingabemodus nicht. Der Cursor kann aber mit Hilfe der Inst/Del-Taste zurückgeführt werden. Größere Fehler können im Unterprogramm 'Text ändern' beseitigt werden. Die Eingabe der Anführungszeichen ist nicht möglich, hier muß das hochgestellte Komma verwendet werden. Für die Texteingabe stehen 100 Zeilen mit maximal 70 Zeichen zur Verfügung. Bei Erreichen des 65. Zeichens, ertönt bei jeder weiteren Eingabe ein akustisches Signal. Sind 70 Zeichen eingegeben, wird ein automatischer Zeilenvorschub durchgeführt. Drückt man bei Beginn einer neuen Zeile die 'Pfeil nach oben'-Taste, gelangt man wieder ins Hauptmenue.

2. Text ändern:

Das Programm fragt die Zeilennummer ab, in welcher der zu ändernde Text gespeichert ist. Diese wird dann auf dem Bildschirm sichtbar gemacht, und kann darunter neu eingegeben werden. Drückt man die 'Cursor Right'-Taste, wird der alte Text in die neue Zeile kopiert. Betätigt man die F1-Taste, wird der gesamte Text gelöscht. Gelöschte Zeilen können sofort oder später wieder einge-

fügt werden. Dies muß aber ebenfalls im Untermenue Textändern vorgenommen werden. Werden 70 Zeichen eingegeben, wird keine Zeilenschaltung vorgenommen, sondern das Programm fragt, ob die Zeile übernommen werden soll (z.B. war das 70. Zeichen gerade der Schlußpunkt). In diesem Falle ist die 'Pfeil nach oben'-Taste zu drücken. Hat man die Zeile nicht korrekt eingegeben, drückt man F1 und gibt die Zeilennummer nochmals an. Der Text kann dann neu eingegeben werden. Bleiben dann Zeichen aus der vorherigen Eingabe stehen, können diese unberücksichtigt bleiben, das Programm "schneidet" sie ab.

3. Text löschen:

Durch Drücken von F1 wird der gesamte Text gelöscht.

stättigt werden. Auch von hieraus kann mit der 'PnOT' wieder ins Hauptmenue zurückgekehrt werden. So wird wirkungsvoll verhindert, daß ein falscher Tastendruck den bisher eingegebenen Text unwiederbringlich löscht.

Diskettenmenue:

1. Text speichern:

Es wird nach dem Namen des zu speichernden Textes gefragt. Nach dem Abspeichern kehrt das Programm zum Diskettenmenue zurück. Existiert bereits ein File mit dem gleichen Namen, wird es überschrieben. Ist man sich also über die vorhandenen Files nicht mehr im klaren, sollte man vorher die Directory lesen.

2. Text laden:

Nachdem der Textname abgefragt wurde, wird dieser mit Get eingelesen, und dem String Te\$(zn) zugeordnet. Bei längeren Texten dauert dies leider einige

Zeit. Nach dem Einlesen wird das Hauptmenue angezeigt.

3. Textdatei löschen:

Es wird der Name abgefragt und dann die entsprechende Datei gelöscht.

4. Textdatei umbenennen:

Es wird der alte und der neue Name abgefragt. Während dem Umschreiben wird dies sichtbar gemacht.

5. Formatieren:

Es wird Name und Id gefragt. Nach dem Formatieren Rückkehr ins Menue.

6. Directory:

Die Directory kann eingelesen werden, ohne daß das Programm oder der Text gelöscht werden. Das Einlesen geht sehr rasch, da hierfür ein Unterprogramm in Maschinensprache aufgerufen wird.

7. Hauptmenue:

Durch Drücken der 7 gelangt man wieder ins Hauptmenue zurück.

Fehlermeldungen:

Bei sämtlichen Operationen mit dem Diskettenlaufwerk wird der Fehlerkanal abgefragt. Tritt ein Fehler auf, wird dieser angezeigt und das Programm wartet auf einen Tastendruck und kehrt dann ins Hauptmenue zurück. Die Fehlermeldung 'Device Not Present' wird ebenfalls angezeigt. Der Benutzer wird gebeten, das angesprochene Gerät einzuschalten und eine Taste zu drücken. Nach dem Tastendruck fährt das Programm mit der Arbeit fort.

Wird bis zum Erreichen der möglichen 100 Zeilen geschrieben, springt das Pro-

sonsten können Sie mit diesem Programm so gut wie alles. Geschriebene Texte abspeichern, laden, editieren, Text-

dateien löschen oder umbenennen. Die Directory laden, natürlich, ohne daß das im Rechner befindliche Programm oder der Text gelöscht wird.

Haben Sie einen Text abgespeichert und wollen sofort einen weiteren eingeben, so genügt ein Tastendruck und der alte Text wird gelöscht.

Ich möchte nun doch nicht alle Möglichkeiten des Programmes hier in diesem Musterbrief niederschreiben, diese Angaben können Sie viel besser der Programmbeschreibung entnehmen.

Hartmut Götze

4. Textausgabe Bildschirm:

Der gesamte Text wird mit vorangestellter Zeilennummer aufgelistet. Mit der 'Pfeil nach oben'-Taste kann das Listen angehalten und wieder gestartet werden. Ist das Textende erreicht, wird dies angezeigt, und ein erneutes Drücken der 'Pfeil nach oben'-Taste zeigt wieder das Hauptmenue.

5. Druckerausgabe:

Das Programm fragt ab, ob der Absender mit ausgedruckt werden soll. Dann wird abgefragt, ob weitere Daten eingegeben werden. Wird die Frage mit 'Ja' beantwortet, so fragt das Programm folgendes ab:

- a) Anrede (z.B. Firma),
- b) Vorname,
- c) Name,
- d) Straße u. Nr.,
- e) PLZ u. Ort,
- f) Datum,
- g) Betreff.

Dann werden 3 Anredeformen zur Auswahl gestellt:

- a) Sehr geehrte Damen und Herren,
- b) Sehr geehrte Frau,
- c) Sehr geehrter Herr.

Trifft hiervon keine zu, kann eine eigene eingegeben werden.

Hinter b und c wird der Name mit ausgedruckt. Bei der eigenen Anrede muß er mit eingegeben werden. Hinter Anredeform einer gewählten oder der eigenen, wird ein Komma gedruckt. Der erste Satz sollte also kleingeschrieben beginnen. Dann wird abgefragt, ob ein Schlußwort ausgedruckt werden soll. Beantwortet man die Frage mit 'Ja', werden 4 Formen zur Auswahl angeboten:

- a) mit freundlichen Grüßen;.
- b) Mit freundlichen Grüßen;.
- c) Hochachtungsvoll;.
- d) Gezeichnet;.

Trifft keines zu, kann wieder eine eigene Eingabe gemacht werden. Hinter jedes

Schlußwort, gleich ob gewählt oder eingegeben, wird ein Doppelpunkt gesetzt. Nach dem Ausdruck fragt das Programm, ob ein weiterer Ausdruck gewünscht wird. Beantwortet man die Frage mit 'Ja', so wird gefragt, ob Daten geändert werden sollen. Bei Beantwortung mit 'Nein', wird ausgedruckt wie vorher. Bei 'Ja' werden die oben erwähnten Punkte abgefragt.

Es ist somit möglich, die Adresse oder andere Daten ständig zu ändern. Will man vorher gemachte Daten beibehalten, muß die Abfrage nur mit Return übersprungen werden. Dies ist möglich, da alle Stringzuweisungen ihre Werte solange behalten, bis kein Ausdruck mehr gewünscht oder diese neu eingegeben werden. Bisher wurde nur gesagt, was

gramm zum Hauptmenue, der Text kann dann noch editiert und abgespeichert werden. Die Run/Stop-Taste wurde außer Betrieb gesetzt, um ein Abstürzen durch versehentliches Drücken der Taste zu verhindern.

Noch ein kleiner Hinweis:

Auf ein DIN A4 Blatt haben incl. Absender, Anschrift und Schlußwort cirka 35 Zeilen Platz.

Die Benutzung der Anführungszeichen ist nicht möglich. Hier muß das hochgestellte Komma verwendet werden.

Um das Ende des Textes bei der Bildschirm und Druckerausgabe zu erkennen, setzt sich das Programm bei Beenden der Texteingabe einen Merker. Dies sind zwei aufeinanderfolgende **. Es sollte darum das Multiplikationszeichen nicht zweimal hintereinander eingegeben werden.

Da ich glaube, nun alles gesagt zu haben, bleibt mir nur noch, Ihnen viel Spaß beim Schreiben zu Wünschen.

Variablenliste:

- DN\$:** Dateiname
ND\$: Neuer Dateiname
AD\$: Alter Dateiname
ID\$: Id (Formatieren)
LD\$: Datei löschen

Drucker:

Folgende Variablen in den Zeilen 30-45 müssen mit den Daten des Anwenders belegt werden. Können natürlich auch ohne Belegung bleiben (z.B. Firmentyp wenn Privat).

- FT\$:** Firmentyp
SE\$: Straße und Nummer
TF\$: Telefon
BZ\$: Bankleitzahl

- KT\$:** Kontonummer
V\$: Vorname/Name
PS\$: Postleitzahl/Ort
BZ\$: Ist belegt

Folgende Variablen werden im Programm abgefragt:

- AS\$:** Anrede bei Adresse
VI\$: Vorname
NI\$: Name
Sy\$: Straße und Nummer
PI\$: Postleitzahl und Ort
DS\$: Datum
BT\$: Betreff
AS\$: Gewählte Anrede
SH\$: Gewählter Schluß

Sonstige:

- ZE\$:** Obere Maske
A,BL,BH\$: Directory einlesen
EN,FA,WA\$: Get Abfrage

das Programm macht, wenn mit 'Ja' geantwortet wird. Es dürfte jedem verständlich sein, daß der Absender nicht ausgedruckt wird, wenn man mit 'Nein' antwortet, gleiches gilt für die Anschrift, Datum, Betreff, Anrede und Schlußwort, wenn man die Frage nach weiteren Eingaben verneint. Es ist somit möglich, den eingegeben Text ohne weitere Daten auszudrucken, oder den ersten Ausdruck mit Daten und den zweiten (Kopie ohne). Es sind alle Kombinationen möglich. Beim Ausdruck ohne weitere Daten gibt es zwei Möglichkeiten, wenn ein weiterer Ausdruck gewünscht wird.

Wenn das Programm nach der mit 'Ja' beantworteten Frage nach dem zweiten Ausdruck fragt ob Daten geändert werden sollen, gibt man 'Nein' ein. Es werden dann zwischen dem bereits Ausgedruckten und dem zweiten Ausdruck, 4 Zeilenschaltungen vorgenommen. Soll aber der zweite Ausdruck (es kann sich hierbei auch um Text handeln, der neu eingeladen wurde, wenn z.B. 100 Zeilen nicht ausreichen) sofort an den 1. Ausdruck angehängen werden, geht man folgendermaßen vor. Zweiter Ausdruck JA, Daten ändern JA (obwohl eigentlich nichts zu ändern ist) Absender ausdrucken NEIN, weitere Daten eingeben NEIN. Der Text wird an den ersten angehängen. Wem dies nun alles etwas unverständlich vorgekommen ist, den möchte ich bitten, einen kleinen Text einzugeben und die Vielzahl der möglichen Druckersteuerungen auszuprobieren.

6. Diskettenmenue:

Wird gesondert erläutert.

7. Farbwechsel:

Mit den Funktionstasten F1, F3 und F5 können Hintergrund-, Rahmen- und Schriftfarbe gewählt werden. Mit F7 gelangt man wieder ins Hauptmenue.

Die beim Start eingestellte, für manchen wohl etwas 'poppige' Kombination von rotem Rahmen und Hintergrund mit weißer Schrift wurde gewählt, da diese einer der besten Kombinationen für Monochrome oder mit Grünumschalter bestückte Monitore ist.

8. Anweisungen:

Die Anweisungen sollen nur eine Hilfsfunktion erfüllen und sind deshalb in Kurzform gehalten. Die Punkte Text eingeben, Text ändern, Text einfügen und Text löschen sind kurz erläutert. Dies nur, damit bei dem anfänglichen Arbeiten die Zeitschrift nicht daneben liegen muß.

9. Programm beenden:

Das endgültige Programmende muß nochmals durch Drücken der F1 be-

Texteingabe:

TES: Text

NT\$: Eingabe

ZN\$: Eingabe Zeilennummer

ZN: Zeilennummer

ME: Merker

Weitere:

HF: Hintergrundfarbe

RF: Rahmenfarbe

SF: Schriftfarbe

DN%: Geräteadresse

DR%: Gerät EIN/AUS (0/1)

Programmaufbau:

25-100: Stringzuweisung, Dimensionierung, Farbe

130-580: Hauptmenue und Abfrage

600-850: Texteingabe

900-1260: Texteditor

1300-1460: Bildschirmausgabe

1500-1590: Textausdruck

1600-1680: Text löschen

1700-1770: Text speichern

1800-1910: Text laden

2000-2100: Farbwechsel

2200-2280: Programmabbruch

2300-2320: Ton

2400-2775: Eingabe weiterer Daten

2800-2990: Druck Absender und weiterer Daten

3000-3790: Anweisungen

4000-4040: Fehlerkanal abfragen

4100-4240: Disketten Menue und

Abfrage

4300-4355: Datei löschen

4400-4490: Datei umbenennen

4500-4570: Disk formatieren

4600-4690: Directory listen

5000-5120: Überprüfen ob Gerät EIN/AUS

5140: Fehlerkanal löschen

5200-5270: Maschinendatas für Direct u. Überprüfung EIN/AUS.

5300-5340: Copyright

5360-5370: Rücksetzen der Stringzuweisung

Ich möchte diesen Mustertext dazu benutzen, Ihnen, liebe Leser, viel Spaß mit diesem einfachen Textverarbeitungsprogramm zu wünschen. Es war mir schon lange ein Dorn im Auge, daß es zwar eine Vielzahl von veröffentlichten Textverarbeitungsprogrammen gibt, welche aber immer auf Schönschreibdrucker abgestimmt sind.

Was, so frage ich, nützt der schönste Zeichensatz mit deutschen Umlauten, wenn mein Printer sie nicht ausdrucken kann. Was nutzt es, wenn der Text in verschiedenen Schriftgrößen ausgedruckt werden kann, mein Printer kann es nicht.

Dieser kennt nur die Standard- und die Fettschrift. Das muß also reichen. An-

```

25 a=fre(0):clr:gosub5200:open5,8,15:Poke788,peek(788)+3:na=0:j=0:e=0
30 ft$="xxxxxxxx - xxxxxxxx - xxxxxxxx":v$="xxxxxxxx xxxxxxxx"
35 p$="xxxxx xxxxxxx/xxxxxx"
40 se$="xxxxxxxx-xxx.xx":tf$="Tel.:xxxxx / xxxxx"
45 bv$="Bankverbindung":bz$="BLZ xxx xxx xx":kt$="Kto.Nr. xxx xxx xxx"
50 :
60 ze$="☐Textverarbeitung                               Win-Soft☐"
70 :
80 dimte$(100):Poke54296,15
100 Poke53280,2:Poke53281,2:Poke646,1:gosub5300
110 :
120 :
130 gosub2300:Printchr$(147);chr$(14);ze$
160 Print:Print"  ☐1 ->☐ Texteingeben
190 Print:Print"  ☐2 ->☐ Textändern
220 Print:Print"  ☐3 ->☐ Textlöschen
250 Print:Print"  ☐4 ->☐ Textausgabe Bildschirm
280 Print:Print"  ☐5 ->☐ Textausgabe Drucker
310 Print:Print"  ☐6 ->☐ Diskettenmenue
320 Print:Print"  ☐7 ->☐ Farbwechsel
330 Print:Print"  ☐8 ->☐ Anweisungen
340 Print:Print"  ☐9 ->☐ Programm Beenden
400 Print:Print:Print:Printtab(15)"Ihre Wahl
430 :
460 :
490 getwa$:ifwa$=""then490
500 gosub2300
520 wa=val(wa$)
550 ifwa<1orwa>9then490
580 on wa9oto 600,900,1600,1300,1500,4100,2000,3000,2200
585 :
590 :
600 gosub2300:Printchr$(147);ze$
605 Print"  Text weiterschreiben F1 sonst ↑"
610 getwa$:ifwa$=""then610
615 ifwa$=chr$(133)thenfori=1to100:ifte$(zn)="**"thente$(zn)="":next:goto630
    
```

programme

```
620 goto630
625 goto610
630 inPut"   Ab welcher Zeilennummer ->   ↑■■■■";zn$:ifzn$=chr$(94)then130
635 zn=int(val(zn$)):ifzn<1orzn>99thenPrint"■■■■":goto630
640 te$(me)=""
645 Print
650 Printzn;": ";
655 az=0
660 get e9$
665 Printchr$(185);chr$(157);
670 ife9$=""then660
675 ife9$=chr$(29)ore9$=chr$(17)ore9$=chr$(145)then660
678 ife9$=chr$(157)ore9$=chr$(19)ore9$=chr$(34)then660
680 ife9$<>chr$(20)then730
685 if len(te$(zn))<1then660
690 Print" ";chr$(157);
700 Printchr$(157);" ";chr$(157);
710 te$(zn)=left$(te$(zn),len(te$(zn))-1)
720 goto660
730 if e9$<>chr$(94)then770
740 if len(te$(zn))<1then830
750 goto660
760 if az=0thente$(zn)="":az=1
770 Print" ";chr$(157);e9$;
780 if e9$=chr$(13)thenzn=zn+1:goto810
790 te$(zn)=te$(zn)+e9$
800 if len(te$(zn))>64 then90sub2300
805 iflen(te$(zn))=70thene9$=chr$(13):zn=zn+1
810 if zn=100then830
815 ife9$=chr$(13)then650
820 goto660
830 Print" ";chr$(157);
840 te$(zn)="*":me=zn
850 90sub2300:goto130
860 :
870 :
900 90sub2300:Printchr$(147):Poke211,0:Poke214,24:sys58732
905 Print"␣13=Loeschen ␣Cursor Right␣=Alten Text Einf.";chr$(19);ze$
910 inPut"   Welche Zeilennummer ->   ↑■■■■";zn$
912 ifr=1thenchr$(13)
915 if zn$=chr$(94)then130
920 zn=int(val(zn$)):ifzn<1orzn>99thenPrint"■■■■":goto910
930 Print
940 Printzn;": ";
950 Printte$(zn)
960 Print
965 Printzn;": ";
970 nt$=""
990 gete9$
1000 Printchr$(185);chr$(157);
1005 ife9$=chr$(17)ore9$=chr$(19)ore9$=chr$(145)ore9$=chr$(157)ore9$=chr$(34)the
n990
1010 ife9$=""then990
1015 ife9$=chr$(133)thente$(zn)="":goto900
1020 ife9$<>chr$(20)then1080
1030 if len(nt$)<1then 990
1040 Print" ";chr$(157);
1050 Printchr$(157);" ";chr$(157);
1060 nt$=left$(nt$,len(nt$)-1)
1070 goto990
1080 if e9$<>chr$(29)then1110
1090 iflen(nt$)<=len(te$(zn))thene9$=mid$(te$(zn),len(nt$)+1,1):goto1150
1100 goto990
1110 ife9$<>chr$(94)then1150
1120 if len(nt$)<1then130
1130 goto990
1140 if az=0thennt$="":az=1
1150 Print" ";chr$(157);e9$;
1160 if e9$=chr$(13)then1200
1170 nt$=nt$+e9$
1175 if len(nt$)>64then90sub2300
1180 if len(nt$)=70then:goto1230
1190 goto990
1200 if len(nt$)<1thennt$=te$(zn):Printnt$
```

```

1210 te$(zn)=nt$
1220 gosub2300:goto900
1230 Print:Print"█ Maximal 70 Zeichen █"
1235 Print:Print"█ Wenn Neue Zeile OK ↑ Druecken "
1240 Print"█ Wenn nochmals Rendern F1 Druecken █"
1245 getwa$:ifwa$=""then1245
1250 ifwa$=chr$(94)thente$(zn)="":te$(zn)=te$(zn)+nt$:goto900
1255 ifwa$=chr$(133)thenPrint"█":goto910
1260 goto1245
1270 :
1280 :
1300 gosub2300:Printchr$(147)
1310 zn=1
1320 ifleft$(te$(zn),2)="**"then1380
1330 Printzn;" ";
1340 Printte$(zn)
1350 gosub1420
1360 zn=zn+1:ifzn=100then1380
1370 goto1320
1380 Print
1390 Printtab(5)"-> █ DER TEXT IST ZU ENDE █ <-"
1400 gete9$:ife9$=""then1400
1405 ife9$=chr$(94)then130
1410 goto1400
1420 gete9$
1430 ife9$<>chr$(94)thenreturn
1440 gete9$
1450 ife9$<>chr$(94)then1440
1460 gosub2300:return
1470 :
1480 :
1500 gosub2300:gosub2400:dn%=4:gosub5000
1505 dn%=4:gosub5000
1508 Printchr$(147);ze$:Print:Printtab(5)"-> █Text wird Ausgedruckt█ <-"
1510 open4,4,7
1515 zn=1
1520 ifleft$(te$(zn),2)="**"then1545
1525 Print#4,chr$(16)"03"te$(zn)
1530 zn=zn+1
1535 ifzn=100then1545
1540 goto1520
1545 ife=1thengosub2970:close4:gosub1550:goto130
1550 gosub2300:Printchr$(147);ze$:Print
1555 Printtab(3):inPut"Noch ein Ausdruck";wa$
1560 ifwa$<>"j"andwa$<>"n"thenPrint"00":goto1555
1565 ifwa$="n"thenclose4:goto5360
1570 Print
1575 inPut"Sollen Daten geaendert werden";wa$
1580 ifwa$<>"j"andwa$<>"n"thenPrint"00":goto1575
1585 ifwa$="n"thenclose4:goto2800
1590 ifwa$="j"thenclose4:goto2400
1592 :
1595 :
1600 gosub2300:Printchr$(147);ze$:Print
1605 Printtab(3)"█TEXTLOESCHEN█"
1610 Print:Printtab(3)"Den im Speicher befindliche Text ":Print
1615 Printtab(3)"durch Druecken von F1 Loeschen.":Print
1620 Print:Printtab(9)"█ Bei Irrtum ↑ Druecken █"
1625 getwa$:ifwa$=""then1625
1630 ifwa$<>chr$(94)andwa$<>chr$(133)then1625
1635 ifwa$=chr$(94)then130
1640 ifwa$=chr$(133)then1650
1645 goto1625
1650 Print:Printtab(10)"Text wird Geloescht"
1655 foran=1to100
1660 te$(an)=" "
1670 next:Print:Printtab(15)"FERTIG":fori=0to700:next
1680 goto130
1685 :
1690 :
1700 gosub2300:dn%=8:gosub5000:ifdr%=1thengosub5140
1705 Printchr$(147);ze$
1710 Printtab(5)"█ TEXT SPEICHERN ":Print:Printtab(5)"Textname -> #####";
1720 inPut dn$:ifdn$=chr$(94)then4100

```

programme

```
1730 oPen1,8,2,"@:"+dn#+",s,w":gosub4000:forzn=1to100
1750 if te$(zn)=""thente$(zn)=chr$(13)
1760 Print#1,te$(zn)
1770 next:Print#1,chr$(13):close1:gosub2300:gosub4000:goto4100
1780 :
1790 :
1800 gosub2300:dn%=8:gosub5000:ifdn%=1thengosub5140
1805 Printchr$(147);ze$
1810 Printtab(5)"  TEXT LADEN  ":Print:Printtab(5)"Textname -> 11111";
1820 inPut dn#:ifdn#=chr$(94)then4100
1830 oPen1,8,2,dn#+",s,r":gosub4000:forzn=1to100
1850 te$=""
1860 get#1,e9$
1870 te$=te$+e9$
1880 ife9#=chr$(13)ore9#=""then1900
1890 goto1860
1900 te$(zn)=te$:te$(zn)=left$(te$(zn),len(te$(zn))-1)
1910 next:close1:gosub2300:goto130
1920 :
1930 :
2000 gosub2300:Printchr$(147);ze$
2010 Print:Print:Print"  F1 -> Hintergrundfarbe
2020 Print:Print:Print"  F3 -> Rahmenfarbe
2030 Print:Print:Print"  F5 -> Schriftfarbe
2040 Print:Print:Print"  F7 -> Menue
2050 getfa$:iffa#=""then2050
2060 iffa#=chr$(133)thenhf=hf+1:Poke53281,hf:ifhf>14thenhf=-1
2070 iffa#=chr$(134)thenrf=rf+1:Poke53280,rf:ifrf>14thenrf=-1
2080 iffa#=chr$(135)thensf=sf+1:Poke646,sf:Print"#####Schriftfarbe"
2085 ifsf>14thensf=-1
2090 iffa#=chr$(136)thengosub2300:goto130
2100 goto2050
2110 :
2120 :
2200 gosub2300:Printchr$(147);ze$:Print
2210 Print:Print"Bitte Bestaetigen sie den Endgueltigen
2220 Print:Print"Programmabbruch durch Druecken von F1 !
2230 Print:Printtab(8)"  Bei Irrtum ↑ Druecken  "
2240 getwa$:ifwa#=""then2240
2250 ifwa#<chr$(94)andwa#<chr$(133)then2240
2260 ifwa#=chr$(94)then130
2270 sys65129
2280 goto2240
2290 :
2300 si=54272:Pokesi+5,32:Pokesi+6,15*16:Pokesi+24,15
2310 Pokesi,59:Pokesi+1,139:Pokesi+4,33
2320 fora=1to100:next:Pokesi+4,0:return
2330 :
2340 :
2400 gosub2300:Printchr$(147);ze$
2405 Print:Print"Soll der Absender Ausgedruckt werden? ":Print
2410 Print"  Bei Irrtum ↑ Druecken"
2415 get en$:ifen#=""then2415
2420 ifen#<chr$(74)anden#<chr$(78)anden#<chr$(94)then2415
2425 ifen#="↑"then130
2430 ifen#="j"thema=0:goto2445
2435 ifen#="n"thema=1:goto2445
2440 goto 2415
2445 gosub2300:Printchr$(147);ze$
2450 Print:Print"Moechten Sie weitere Daten Eingeben ? ":Print
2455 get en$:ifen#=""then2455
2460 ifen#<chr$(74)anden#<chr$(78)then2455
2465 ifen#="j"thenj=2:goto2480
2470 ifen#="n"andna=1andj=0then1505
2472 ifen#="n"andj=2orj=0then2800
2475 goto2455
2480 Print"  RETURN WENN KEINE EINGABE ":Print
2485 inPut"Anrede      ";a$
2490 inPut"Vorname     ";v1$
2495 inPut"Name        ";n1$
2500 inPut"Strasse u.Nr. ";s1$
2510 inPut"Plz Ort     ";p1$
2515 Print
2520 Print:inPut"Datum      ";d$
```

```

2525 Print:input"Betreff          ";bt$:iflen(bt$)>0thenbt$="Betreff:"+bt$
2530 gosub2300:Printchr$(147);ze$:Print
2535 Print"Waehlen Sie nun die Anrede aus :":Print
2540 Print"  1 -> Sehr geehrte Damen und Herren.":Print
2545 Print"  2 -> Sehr geehrte Frau _____":Print
2550 Print"  3 -> Sehr geehrter Herr _____":Print
2555 Print"  4 -> Eigene Eingeben
2560 get wa$:ifwa$=""then2560
2565 wa=val(wa$):ifwa<lorwa>4then2560
2570 onwa goto 2630,2640,2650,2580
2575 goto2560
2580 Print:Print"Namen selber eingeben,Komma wird angeh.
2585 Print:input"Ihre Eingabe ->";as$:as$=as$+","":goto2670
2630 as$="Sehr geehrte Damen und Herren.":goto2670
2640 as$="Sehr geehrte Frau "+nl$+","":goto2670
2650 as$="Sehr geehrter Herr "+nl$+","":goto2670
2670 gosub2300:Print:Print"Schlusswort Eingeben "
2680 gets$:ifs$<"j"ands$<"n"then2680
2685 ifs$="j"thene=1:goto2700
2690 ifs$="n"thene=0:goto2800
2695 goto2680
2700 Printchr$(147);ze$:Print:Print"  1 -> mit freundlichen Gruessen:
2705 Print:Print"  2 -> Mit freundlichen Gruessen:
2710 Print:Print"  3 -> Hochachtungsvoll:
2715 Print:Print"  4 -> Gezeichnet:
2720 Print:Print"  5 -> Eigene Eingeben
2725 get wa$:ifwa$<"1"orwa$>"5"then2725
2730 wa=val(wa$):onwa goto 2760,2770,2775,2780,2740
2735 goto2725
2740 Print:Print" Doppelpunkt wird nachgestellt"
2750 Print:input" Ihre Eingabe ->";sh$:sh$=sh$+","":goto2800
2760 sh$="mit freundlichen Gruessen.":goto2800
2770 sh$="Mit freundlichen Gruessen.":goto2800
2775 sh$="Hochachtungsvoll.":goto2800
2780 sh$="Gezeichnet.":goto2800
2790 :
2800 dnZ=4:gosub5000:open4,4,7:ifna=1then2850
2805 Printchr$(147);ze$:Print:Printtab(5)"->  TEXT WIRD AUSGEDRUCKT <-"
2810 Print#4,chr$(14)chr$(16)"03"v$chr$(15)chr$(16)"55"p$
2820 Print#4,chr$(16)"55"se$
2830 Print#4,chr$(16)"03"ft$chr$(16)"55"tf$
2832 Print#4,chr$(13)chr$(16)"55"bv$
2834 Print#4,chr$(16)"55"bz$
2836 Print#4,chr$(16)"55"kt$
2840 Print#4,chr$(13)chr$(16)"55"d$
2850 ifj=2thengoto2900
2860 fori=1to 4:Print#4,chr$(13):nexti
2870 close4:goto1510
2900 Print#4,chr$(13)chr$(13)
2910 Print#4,chr$(16)"03"a$
2920 Print#4,chr$(16)"03"v1$" "nl$
2930 Print#4,chr$(16)"03"s1$
2940 Print#4,chr$(13)chr$(16)"03"p1$
2950 Print#4,chr$(13)chr$(13)chr$(13)chr$(16)"03"bt$
2960 Print#4,chr$(13)chr$(13)chr$(16)"03"as$chr$(13)
2965 close 4:goto1510
2970 :
2975 :
2980 Print#4,chr$(13)chr$(13)chr$(16)"03"sh$:Print#4,chr$(16)"03"v$
2990 return
2992 :
2995 :
3000 gosub2300:Printchr$(147);ze$:Print
3010 Print"  HELP bei ":Print
3020 Print:Print"  1 -> Texteingabe
3030 Print:Print"  2 -> Texteingeben
3040 Print:Print"  3 -> Textaendern
3050 Print:Print"  4 -> Textloeschen"
3060 Print:Print:Printtab(15)"Ihre Wahl"
3070 getwa$:ifwa$=""then3070
3080 wa=val(wa$)
3090 ifwa<lorwa>4then3070
3100 onwa goto 3120,3280,3410,3590
3110 goto3070

```

programme

```
3120 gosub2300:Printchr$(147);ze#:Print
3130 Print" 2 Texteingabe: 2"
3140 Print:Print" Es stehen ihnen 99 Zeilen mit maximal
3150 Print" 70 Zeichen zur Verfuegun9.Neuer Text
3160 Print" sollte darum immer mit Zeilennummer 1
3170 Print" beginnen.Fehler koennen wie gewohnt
3180 Print" mit der Ins/Del Taste sofort korri-
3190 Print" giert werden.
3200 Print" Moechten Sie die Eingabe beenden
3210 Print" geben sie als erstes Zeichen in einer
3220 Print" neuen Zeile ^t ein und sie gelangen
3230 Print" wieder ins Hauptmenue.
3240 Print" Tonsignal beginnt bei Anbruch der
3250 Print" letzten fuenf Zeichen.Automatisches
3260 Print" Return bei Zeilenende.
3270 goto3780
3280 gosub2300:Printchr$(147);ze#:Print
3290 Print" 2Texteinfuegen: 2
3300 Print:Print" Waehlen sie Punkt 2 Textaendern.
3310 Print" Text.kann nur dann eingefuegt werden
3320 Print" wenn eine Zeile geloescht wird oder
3330 Print" sich zwischen den Zeilen noch eine
3340 Print" befindet die vorher mit Return ueber-
3350 Print" sprungen wurde.
3360 Print" Geben sie die Nummer der noch freien
3370 Print" oder vorher geloeschten Zeile ein.
3380 Print" Eingaben im Modus 2 Textaendern vor-
3390 Print" nehmen und mit Return abschliessen.
3400 gosub 3780
3410 gosub2300:Printchr$(147);ze#:Print
3415 Print" 2 Textaender: 2":Print
3420 Print:Print" Geben sie im Programmmodus zwei die
3430 Print" Nummer der Zeile an welche geaendert
3440 Print" werden soll.Die Zeile wird auf dem
3450 Print" Bildschirm ausgegeben und kann darun-
3460 Print" ter neu eingeegeben werden.
3470 Print" Druucken sie die Cusortaste ^rechts^
3480 Print" so wird der alte sich an der gleichen
3490 Print" stelle befindende Text in die Neue
3500 Print" Zeile uebernommen.Druucken sie F1
3510 Print" so wird die gesamte Zeile geloescht.
3520 Print" Wurden 70 Zeichen eingeegeben ist eine
3530 Print" weitere Eingabe nicht mehr moeglich.
3540 Print" Druucken Sie nun ^t^ so wird die
3550 Print" Zeile so uebernommen.Druucken sie
3560 Print" ^F1^ kann die Zeile Neu eingeseben
3570 Print" werden.
3580 goto3780
3590 gosub2300:Printchr$(147);ze#:Print
3600 Print" 2Textloeschen: 2":Print
3610 Print" Einzelne Zeilen koennen Geloescht
3620 Print" werden indem Sie Punkt 2 Textaendern
3630 Print" Waehlen.
3670 Print" Geben sie die Nummer der Zeile ein
3680 Print" welche sie Loeschen moechten.Wenn
3690 Print" diese Anzeigt wird Druucken sie
3700 Print" ^F1^.Soll die Zeile Neu Geschrieben
3710 Print" werden so ebenfalls im Modus 2. nicht
3720 Print" im Programmteil 1.
3730 Print" Wollen sie den gesamten Text Loeschen
3740 Print" Waehlen sie Punkt 3 und Druucken F1.
3750 Print" Der Text wird geloescht und sie
3760 Print" koennen einen Neuen Text Eingeben.
3770 goto3780
3780 Poke211,12:Poke214,22:sys58732:Print"-> 2Taste 2 <-"
3790 Poke198,0:wait198,1:goto130
4000 :
4020 inPut#5,w,x$,y,z :ifw=0thenreturn
4025 Print:Print:Printtab(5)w,x$,y,z
4030 Print:Print:Printtab(12)"-> 2 TASTE 2 <-" :poke198,0:wait198,1
4040 close 1
4050 :
4060 :
4100 gosub2300:Printchr$(147);ze#:Print
```

```

4110 Print:Print"  1 -> Text speichern
4120 Print:Print"  2 -> Text Laden
4130 Print:Print"  3 -> Textdatei Loeschen
4140 Print:Print"  4 -> Datei umbenennen
4150 Print:Print"  5 -> Diskette Formatieren
4160 Print:Print"  6 -> Directory Lesen
4170 Print:Print"  7 -> Hauptmenue
4180 Print:Print:Printtab(15)" Ihre Wahl
4190 :
4200 get wa$:ifwa$=""then4200
4210 ifwa$(chr$(49)orwa$)chr$(55)then4200
4220 wa=val(wa$)
4230 onwa goto 1700,1800,4300,4400,4500,4600,1300
4240 goto4200
4250 :
4260 :
4300 gosub2300:dn%=8:gosub5000:ifdr%=1thengosub5140
4305 Printchr$(147);ze$
4310 Printtab(3)"  TEXT LOESCHEN  ":Print:Print
4320 Printtab(3):input"Textname ->  ";ld$:ifld$=chr$(94)then4100
4330 dn$="s0:"+ld$
4340 Print:Printtab(3)ld$" wird geloescht "
4350 open 1,8,15:Print#1,dn$:close1
4355 gosub4000:goto4100
4360 :
4370 :
4400 gosub2300:dn%=8:gosub5000:ifdr%=1thengosub5140
4405 Printchr$(147);ze$
4410 Printtab(3)"  DATEI UMBENENNEN  ":Print
4420 Printtab(3)"Alter Dateiname ->  ";
4430 input ad$:ifad$=chr$(94)then4100
4450 Print:Printtab(3)"Neuer Dateiname";
4460 input nd$
4465 dn$="r0:"+nd$+"="+ad$
4470 Print:Printtab(3)"ad$"=> "nd$"
4480 open1,8,15:Print#1,dn$:close1:gosub4000
4490 goto4100
4495 :
4498 :
4500 gosub2300:dn%=8:gosub5000:ifdr%=1thengosub5140
4505 Printchr$(147);ze$
4510 Printtab(3)"  Diskette Formatieren  ":Print
4520 input"  Welchen Namen ->  ";nd$:ifnd$=chr$(94)then4100
4530 Print:input"  Welche ID ->  ";id$:ifid$=""thenid$=","+id$
4540 dn$=nd$+id$:Print:Printtab(3)"  Diskette eingelegt ?"
4550 Print:Printtab(8)"-> Taste Druucken <- "
4560 open1,8,15,"n:"+dn$:close1:gosub4000
4570 goto4100
4580 :
4590 :
4600 gosub2300:dn%=8:gosub5000:ifdr%=1thengosub5140
4605 Printchr$(147);ze$
4610 Printtab(3)"  DIRECTORY LESEN  "
4620 open3,8,0,"#0":get#3,q#,q#
4630 get#3,q#,q#,bl#,bh#
4640 ifq$=""thenclose3:gosub4690:goto4100
4650 bl#=bl#+chr$(0):bh#=bh#+chr$(0)
4660 Print256*asc(bh#)+asc(bl#);
4670 sys832:goto4630
4680 gosub4690:goto4100
4690 Printtab(7)"-> TASTE <-":poke198,0:wait198,1:return
4695 :
4698 :
5000 poke 2,dn$:sys49152:ifpeek(2)=0thendr%=1:return
5010 :
5020 dr%=0
5030 gosub2300:Printchr$(147);ze$:Print:Print
5040 ifdr%=4thenxy$="DRUCKER"
5050 ifdr%=8thenxy$="FLOPPY "
5060 for i=0to5
5070 Printtab(4)"-> -> -> ";xy$;" <- <- <-  ":fora=0to40:nexta:
5080 Printtab(4)"-> -> -> ";xy$;" <- <- <-  "
5090 nexti
5100 Print:Print:Print:Print:Printtab(4)"  Gerat Bitte Einschalten !! "

```

```

5110 Print:Printtab(4) "  Taste Druecken wenn OK "
5120 Poke198,0:wait198,1:return
5130 :
5135 :
5140 oPen1,8,15,"ui":close1:return
5200 fori=49152to49185:readx:Pokei,x:next
5210 data 169,1,160,192,162,33,32,189,255,169,1,160,15,166,2,32,186,255,32
5220 data 192,255,176,2,169,0,133,2,169,1,32,195,255,96,32
5230 fori=0to23:readx:Poke832+i,x:next
5240 data 162,3,32,198,255,32,207,255,32
5250 data 210,255,208,248,169,13,32,210
5260 data 255,76,204,255,0,0,0
5270 return
5280 :
5290 :
5300 gosub2300:Printchr$(147)chr$(14):ze#
5310 Print:Print:Print:Printtab(13)"COPYRIGHT 1984"
5320 Print:Print:Printtab(19)"by"
5330 Print:Print:Printtab(13)"HARTMUT GOETZE"
5340 fori=0to3500:next:return
5350 :
5360 v1#="":n1#="":s1#="":a#="":P1#="":bt#="":as#="":
5370 na=0:j=0:goto130
:d#="":sh#=""

```

JOYSTICK-GRAFIK für den VC-20 + 16 K

Das Programm ist menuegesteuert, und nach dem Einladen erscheint zuerst das Menue.

Zeichnen

Es erscheint eine kurze Beschreibung.

F1 = Das Programm kehrt zum Menue zurück.

Das Gezeichnete ist damit nicht verloren; es kann jederzeit durch 'Wiederholen' zurückgerufen werden.

F3+F5 = Mit diesen beiden Funktionstasten ist es möglich, eine Zeichenroutine zu erstellen und diese später an jeder beliebigen Stelle des Bildschirms abzurufen.

F3 = Drücken, gewünschte Routine zeichnen, wiederum F3 drücken.

An jeder beliebigen Stelle mit **F5** abrufen. Die Routine verschwindet bei Erstellung einer neuen oder bei Programmunterbrechung, nicht aber bei Rückkehr zum Menue.

Restore = Hardcopy – nach Drücken der Restore-Taste ertönt ein Ton und der Computer erwartet das Drücken der Taste 1 bzw. der Taste 2. Die Tasten bedeuten: 1 \$ Ausdruck normal – 2 = Ausdruck 4-fach vergrößert.

Danach ertönt noch ein Ton. Mit drücken

der Taste 9 wird im RVS ON MODUS ausgedruckt. 0 = RVS OFF.

Das Hardcopyprogramm kann in allen Phasen des Grafik-Prog. aufgerufen werden (Vorsicht).

1+2 = Mit diesen Tasten kann die Geschwindigkeit des Cursors variiert werden. Um schwierige Figuren zu malen, ist es erforderlich, eine langsame Geschwindigkeit zu wählen, da eine größere Genauigkeit so erzielt werden kann. Die Tasten sind nur bei Stillstand des Cursors zu betätigen.

Gezeichnet wird mit dem Joystick. Dabei ist unbedingt zu beachten, daß der Rand rechts und links (durch Striche markiert) nicht überschritten wird – sonst ist es möglich, daß das Maschinenprogramm versehentlich gelöscht wird.

Der Feuerknopf dient zum An- und Ausschalten der Zeichenfarbe. Auf diese Einstellung ist beim Aufrufen einer Routine zu achten. Wichtig: Soll eine Linie gelöscht werden, so ist sie einfach mit dem 'ausgeschaltetem Punkt' nachzuziehen. Das Programm fragt zu einem bestimmten Zeitpunkt die Zeichen- und Bildschirmfarbe ab. Dazu ist die Farbtabelle aus dem Handbuch heranzuziehen;

bei der Zeichenfarbe gelten die auf den Tasten stehenden Zahlen vermindert um eins. Das Programm baut den Bildschirm um, und der Spaß kann beginnen.

Wiederholen

Die Zeichen- und Bildschirmfarbe ist, wie oben beschrieben, festzulegen; daraufhin erscheint das eben Gezeichnete wieder, und ebenfalls wieder der Cursor in der oberen Hälfte des Bildschirms, so daß weitergezeichnet werden kann (so ist es auch möglich, eine Zeichnung in beliebigen Farbkombinationen zu betrachten und zu ändern). Der Bildschirminhalt geht auch bei Programmunterbrechung nicht verloren. Es kann einfach mit Run wieder gestartet werden, und das Gezeichnete erscheint nach dem Menue durch Druck auf die 2 wieder. Drückt man in so einem Fall den Menue-Punkt zeichnen, geht der alte Bildschirminhalt jedoch verloren.

Bild einspielen

Bei Druck auf die Taste 3 kann ein auf Band gespeichertes Bild in den Computer geladen werden (Ein vorher abgespeichertes Bild läßt sich also nach Wochen und Monaten noch vervollständigen).

gen, wenn man will).

1. Band auf entsprechende Position bringen
2. Farbkombination wählen
3. Play on Tape drücken

(Der Bildschirm wird umgebaut, und das Bild Stück für Stück auf den Bild-

schirm gezeichnet.)

Bild aufnehmen

Natürlich kann ein soeben gezeichnetes Bild auch auf Band abgespeichert werden.

1. Band auf leere Bandstelle spulen
2. Die 4 drücken
3. Press Record and Play on Tape

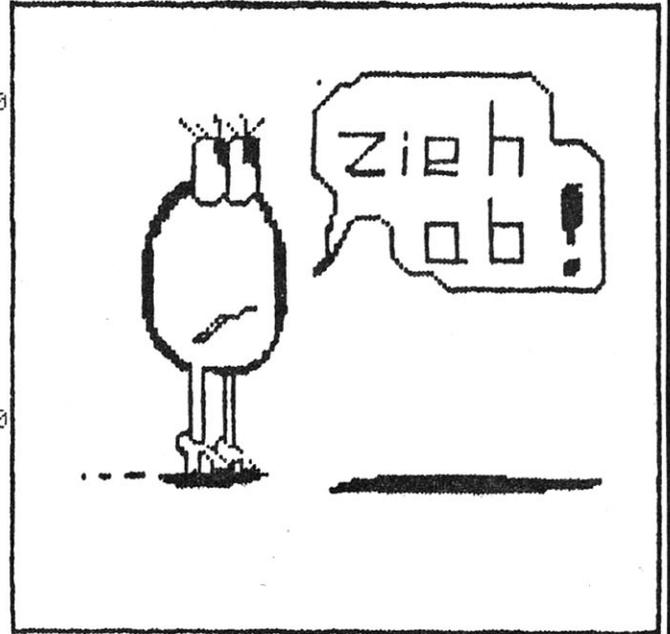
Leider dauert das Laden und Speichern der Dateien von bzw. auf Band sehr lange (134 Umdrehungen des Counters). Die entsprechenden Routinen können jedoch sehr einfach durch Änderung der Geräte-Nummer für ein Diskettenlaufwerk umgeschrieben werden.

JOYSTICK-GRAFIK 1

```

10 REM:VORPROGRAMM
20 GOTO660
30 REM:HARDCOPY-FROG.
40 IFPEEK(648)=30THEN80
50 POKE1024,85:IFPEEK(1024)=85THENA=4:GOTO100
60 A=PEEK(56)-2:IFPEEK(55)THENA=A-1
70 POKE56,A:POKE52,A:POKE55,0:POKE51,0:GOTO100
80 IFPEEK(56)<16THEN30
90 A=PEEK(44)-2
100 B=A*256:C=0
110 FORX=0TO489:READD:C=C+D:POKEB+X,D:NEXT
120 IFC<>49587THENPRINT"OFEHLER":END
130 POKEB+6,A:POKEB+10,A:POKEB+130,A
140 POKEB+164,A+1:POKEB+169,A:POKEB+408,A
150 SYSB
160 DATA169,11,141,24,3,169,4,141,25,3
170 DATA96,72,138,72,152,72,32,44,4,76
180 DATA170,254,169,15,141,14,144,169,176,141
190 DATA12,144,170,168,136,208,253,202,208,250
200 DATA142,14,144,96,216,160,0,162,17,181
210 DATA0,149,0,72,202,16,248,173,5,144
220 DATA170,41,7,10,10,10,133,8,138,41
230 DATA8,208,1,56,102,8,173,2,144,168
240 DATA41,127,133,9,152,41,128,240,1,56
250 DATA138,41,112,144,2,9,8,74,133,17
260 DATA138,41,128,208,1,56,102,17,173,3
270 DATA144,41,127,74,38,12,240,1,10,10
280 DATA10,10,133,10,169,128,133,2,32,22
290 DATA4,32,159,255,165,197,201,56,240,25
300 DATA201,0,240,21,201,15,240,14,201,62
310 DATA208,235,169,173,141,24,3,169,254,141
320 DATA25,3,76,213,5,133,13,32,22,4
330 DATA32,159,255,165,197,201,4,240,4,201
340 DATA60,208,243,133,11,24,169,4,170,160
350 DATA255,32,186,255,169,0,133,198,32,189
360 DATA255,32,192,255,162,4,32,201,255,176
370 DATA207,169,8,32,210,255,169,13,32,210
380 DATA255,24,165,9,133,7,165,3,101,0
390 DATA133,5,197,10,144,4,169,0,240,92
400 DATA166,12,240,1,74,74,74,74,133,6
410 DATA169,0,133,15,133,16,162,8,70,7
420 DATA144,3,24,101,6,106,102,15,202,208
430 DATA243,24,101,17,133,16,24,165,1,101
440 DATA15,133,15,138,101,16,133,16,161,15
450 DATA133,15,134,16,24,169,3,101,12,168
460 DATA138,70,5,106,6,15,38,16,136,208
470 DATA246,164,12,208,1,74,74,74,74,74
480 DATA101,15,133,15,165,8,101,16,133,16
490 DATA161,15,24,36,2,240,1,56,102,4
500 DATA230,3,160,6,196,3,176,58,134,3
510 DATA165,4,166,13,240,22,24,169,4,101
520 DATA14,170,70,4,8,106,40,106,202,208
530 DATA247,166,14,208,8,70,4,106,162,4
540 DATA228,11,208,2,73,255,56,106,133,4
550 DATA32,210,255,166,13,240,5,165,4,32
560 DATA210,255,70,2,176,3,76,221,4,102
570 DATA2,230,1,165,9,197,1,208,243,160
580 DATA0,132,1,169,13,32,210,255,165,13
590 DATA240,14,165,14,208,6,169,4,133,14
600 DATA208,220,169,0,133,14,24,165,0,105
610 DATA7,133,0,165,5,197,10,144,203,169
620 DATA15,32,210,255,169,13,32,210,255,32
630 DATA204,255,169,4,32,195,255,162,0,134
640 DATA198,104,149,0,232,224,18,48,248,96
650 RETURN
660 POKE36879,25
670 PRINT"VIC-JOYSTICK-GRAFIK"
680 PRINT"BY"
690 PRINT"J. M. HEISE"
700 PRINT"AND"
710 PRINT"H. AHRENSFELD"
720 PRINT"BITTE WARTEN."
730 PRINT"DAS HAUPTPROGRAMM"
740 PRINT"WIRD EINGELADEN."
750 GOSUB40,760
760 IFPEEK(44)=10THENPOKE44,34:POKE8704,0
770 POKE631,131:POKE198,1
780 END

```



JOYSTICK-GRAFIK 2

```

10 GOSUB1440
20 REM:HAUPTPROGRAMM
30 REM:JOYSTICK-GRAFIK
40 REM:  VON
50 REM:H. AHRENSFELD
60 REM:  UND
70 REM:J. M. HEISE
80 REM:3100 CELLE
90 REM:FUER VC-20 +16K
100 REM:+ 1525 +1530
110 REM: DATEN FUER
120 REM:MASCHINENPROG.
130 FORT=0T0137:READA:POKE4600+T,A:NEXTT
140 DATA169,16,141,0,144,169,46,141,1,144
150 DATA169,0,141,2,144,169,21,141,3,144
160 DATA169,0,133,253,169,16,133,254,160,0
170 DATA162,0,165,240,157,0,148,138,145,253
180 DATA24,152,105,19,168,201,190,208,10,160
190 DATA0,230,253,165,253,201,19,240,4,232
200 DATA76,32,18,96,169,0,133,253,169,20
210 DATA133,254,162,12,160,0,169,0,145,253
220 DATA136,208,249,202,240,7,160,0,230,254
230 DATA76,76,18,169,205,141,5,144,169,19
240 DATA141,2,144,169,255,133,253,169,19,133
250 DATA254,160,160,169,128,145,253,136,208,249
260 DATA169,63,133,253,169,31,133,254,160,160
270 DATA169,1,145,253,136,208,249,96
280 FORT=0T0215:READA:POKE4750+T,A:NEXTT
290 DATA169,128,133,249,169,160,133,253,169,25
300 DATA133,254,169,0,133,163,169,1,133,250
310 DATA160,20,169,127,141,34,145,173,32,145
320 DATA41,128,133,251,169,255,141,34,145,173
330 DATA31,145,41,28,24,101,251,133,251,173
340 DATA31,145,41,32,133,252,132,165,165,249
350 DATA133,166,165,253,133,167,165,254,133,168
360 DATA165,251,201,152,208,3,32,138,19,201
370 DATA24,208,6,32,138,19,32,140,19
380 DATA201,28,208,3,32,140,19,201,20,208
390 DATA6,32,140,19,32,136,19,201,148,208,3
400 DATA32,136,19,201,132,208,6,32,136,19
410 DATA32,173,19,201,140,208,3,32,173,19
420 DATA201,136,208,6,32,173,19,32,138,19
430 DATA165,252,201,0,208,23,165,250,201,0
440 DATA208,6,169,1,133,250,208,4,169,0
450 DATA133,250,173,31,145,201,94,240,249
460 DATA165,250,201,1,208,13,132,164,164,165
470 DATA56,177,167,229,166,145,167,164,164,177
480 DATA253,5,249,145,253,165,197,201,0,208
490 DATA2,230,163,201,56,208,2,198,163,201
500 DATA63,208,3,76,206,19,96
510 FORT=0T069:READA:POKE5000+T,A:NEXTT
520 DATA200,96,136,96,165,249,201,1,240,3
530 DATA70,249,96,169,128,133,249,165,253,201
540 DATA96,48,2,230,254,201,224,208,2,230
550 DATA254,24,105,160,133,253,96,165,249,201
560 DATA128,240,3,6,249,96,169,1,133,249
570 DATA165,253,201,160,16,2,198,254,201,0
580 DATA208,2,198,254,56,233,160,133,253,96
590 FORT=0T044:READA:POKE5070+T,A:NEXTT
600 DATA132,164,200,192,160,240,13,177,253,5
610 DATA249,209,253,240,5,145,253,76,208,19
620 DATA164,164,136,192,255,240,13,177,253,5
630 DATA249,209,253,240,5,145,253,76,228,19
640 DATA164,164,32,140,19,96
650 REM: BASIC-PROG.
660 PRINT"J+ JOYSTICK-GRAFIK +"
670 PRINT"001=ZEICHNEN"
680 PRINT"02=WIEDERHOLEN"
690 PRINT"03=BILD EINSPIELEN"
700 PRINT"04=BILD AUFNEHMEN"
710 GETA#
720 IFA#="1"THEN770
730 IFA#="2"THEN1080
740 IFA#="3"THEN1130
750 IFA#="4"THEN980

```

```

760 GOTO710
770 REM: ZEICHNEN
780 PRINT"J+ JOYSTICK-GRAFIK +"
790 PRINT"☐F1 MENUE":PRINT"F3 ROUTINE ERSTELLEN":PRINT"F5 ROUTINE ABFRAGEN"
800 PRINT"☐1 LANGSAMER":PRINT"2 SCHNELLER"
810 PRINT"TRIGGER CURSOR ON/OFF"
820 PRINT"☐RESTORE HARDICOPY    ☐☐- NICHT UEBER RAND -":GOSUB1520,830
830 INPUT"☐ZEICHENFARBE 0☐☐☐☐":A:POKE240,A
840 INPUT"SCHIRMFARBE 25☐☐☐☐":A
850 POKE36879,A
860 SYS(4608):SYS(4672)
870 SYS(4750)
880 SYS(4772)
890 FORT=0TOPEEK(163):NEXTT
900 GETA$
910 IFA$="☐"THEN950
920 IFA$="☐"THEN1250
930 IFA$="☐"THEN1360
940 GOTO880
950 PRINT"J":POKE36869,192:POKE36867,46
960 POKE36866,22:POKE36864,12:POKE36865,38
970 GOTO650
980 REM: BILD AUF-
990 REM: NEHMEN
1000 PRINT:PRINT:PRINT
1010 OPEN1,1,1,"BILD"
1020 FORT=5120T08159
1030 B=PEEK(T)
1040 PRINT#1,B
1050 NEXTT
1060 CLOSE1
1070 GOTO650
1080 REM: WIEDERHOLEN
1090 INPUT"ZEICHENFARBE 0☐☐☐☐":A:POKE240,A
1100 INPUT"SCHIRMFARBE 25☐☐☐☐":A
1110 POKE36879,A
1120 SYS(4608):POKE36869,205:POKE36866,19:GOTO870
1130 REM: BILD EIN-
1140 REM: SPIELEN
1150 INPUT"ZEICHENFARBE 0☐☐☐☐":A:POKE240,A
1160 INPUT"SCHIRMFARBE 25☐☐☐☐":A:POKE36879,A
1170 OPEN1,1,0,"BILD"
1180 SYS(4608):SYS(4672)
1190 FORT=5120T08159
1200 INPUT#1,B
1210 POKET,B
1220 NEXTT
1230 CLOSE1
1240 GOTO870
1250 :
1260 C=0
1270 FORT=0T0499
1280 SYS(4772)
1290 GETA$: IFA$="☐"THEN880
1300 IFPEEK(251)=156THEN1280
1310 POKE8200+T,PEEK(251):C=C+1
1320 FORR=0TOPEEK(163):NEXTR
1330 NEXTT
1340 GETA$: IFA$="☐"THEN1340
1350 POKE8200+T,156:GOTO880
1360 :
1370 FORT=1T0C
1380 POKE251,PEEK(8200+T)
1390 SYS(4806)
1400 NEXTT
1410 GOTO880
1420 REM: ENDE
1430 END          1500 PRINT"☐BITTE WARTEN, DAS MASCHINENPROGRAMM WIRD EINGELADEN."
1440 POKE36879,25          1510 FORT=1T0500:NEXT:GOTO130
1450 PRINT"☐☐ VIC-JOYSTICK-GRAFIK"          1520 PRINT"☐HIT A KEY"
1460 PRINT"☐☐☐☐☐☐☐☐☐ BY"          1530 GETA$: IFA$=""GOTO1530
1470 PRINT"☐☐ J. M. HEISE"          1540 RETURN
1480 PRINT"☐☐☐☐☐☐☐☐ AND"
1490 PRINT"☐ H. AHRENSFELD"          READY.

```

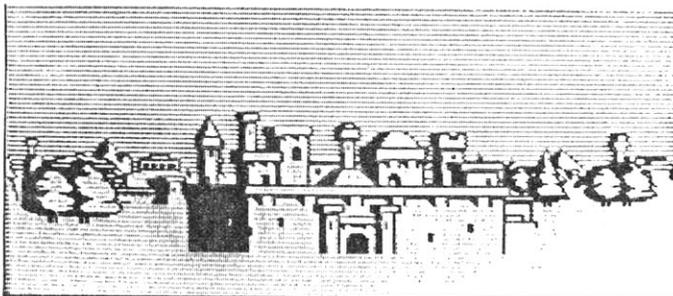
Magdomnad

für den Commodore 64

In einer Burg muß man mit dem Hauptdarsteller des Spieles –Magdomnad–genannt entkommen.
 Zu Beginn des Spieles befindet sich das Männchen rechts unten vor einer Zelle.
 Die Aufgabe besteht nun darin, so schnell wie möglich den Schlüssel zu erreichen.
 Aber Vorsicht! Spinnen, Fässer, Feuerbälle und Fallgruben versuchen einen daran zu hindern.
 Gespielt wird mit einem Joystick an Controllport 2.

Programmaufbau:

- 1- 390: Vorspann Titelbild
- 1000- 1020: Cursor setzen
- 1500- 1530: SID Parameter setzen
- 1600- 1540: Verzweigung Info/Play
- 2000- 2290: Bildschirmmaske
- 3000- 3099: Mann bewegen
- 3100- 3150: V+ 16 prüfen
- 5000- 5999: Mann klettert hoch
- 6000- 6999: Mann klettert runter
- 7000- 7110: Mann springt links
- 8000- 8110: Mann springt rechts
- 10000-10499: Spinne bewegen
- 10500-10999: Faß bewegen
- 11000-11499: Feuerball bewegen
- 20000-20100: Mann stürzt ab
- 30000-30030: Kletterbewegung
- 40000-40050: Sprites einlesen
- 40060-40099: Sprites setzen
- 50060-50220: Info Magdomnad
- 60000-60030: Spielauswertung
- 60050-60160: Pause oder neues Spiel



```

1 REM *****
2 REM *
3 REM * M A G D O M N A D *
4 REM *
5 REM * (C) Z+K SOFTWARE *
6 REM *
7 REM *****
8 REM *** VORSPANN TITELBILD ***
9 REM *****
10 CLR:PRINTCHR$(147)CHR$(19)CHR$(129);:POKE53281,0:POKE53280,0:POKE53265,11
20 V=53248:POKEV:21,4
30 POKE2042,13:POKEV+37,6:POKEV+38,2:POKEV+41,7
40 FORN=0TO62:READQ:POKE832+N,Q:NEXT
50 POKEV+4,160:POKEV+5,50
60 POKEV+23,4:POKEV+29,4:POKEV+28,4
70 FORN=1TO5:FORM=1TO30
80 PRINTCHR$(226);:IFM=15THENFORS=1TO10:PRINTCHR$(32);:NEXTS
90 NEXTM:POKE646,PEEK(646)-1:NEXTN
100 DATA35,85,85,85,85,85,64,0,1,64,40,1,64,170,1,66,170,129,74,170,161,66,170
110 DATA129,64,130,1,64,130,1,66,130,129,64,0,1,79,195,49,64,195,49,67,3,193,76
120 DATA3,49,76,3,49,79,195,49,64,0,1,85,85,85,85,85,85
130 ZE=10:SP=5:GOSUB1000
140 PRINTCHR$(159)"M A G"
150 SP=30:GOSUB1000
160 PRINTCHR$(159)"N A D"
170 ZE=8:SP=12:GOSUB1000
    
```


programme

```
2260 SP=7:ZE=2:GOSUB1000:PRINT"DRUCKE FIRE BUTTON TO PLAY"
2270 J=PEEK(56320):IF(JAND16)<>0THEN2270
2280 GOSUB1000:PRINT"                                ":REM 25 SPACE
2290 TI$="000000":HC$="0000":MA=3:ET=0:M0=3:M1=5
3000 REM *****
3001 REM *** MANN BEWEGEN ***
3002 REM *****
3010 J=PEEK(56320)
3011 IFPEEK(V+31)=60RPEEK(V+31)=140RPEEK(V+30)<>0THENGOTO20000
3012 ON(ET+1)GOSUB1000,10500,11000,11000
3014 IFET=2ANDPEEK(V)=155THENGOTO3030
3015 IFET=4ANDPEEK(V)>=30ANDPEEK(V+16)=1THENGOTO60000
3020 IF(JAND8)=0ANDPEEK(2040)<>14THENPOKEV,14:POKEV,PEEK(V)+5:GOSUB3100
3025 IF(JAND8)=0ANDPEEK(2040)<>13THENPOKEV,13:POKEV,PEEK(V)+5:GOSUB3110
3027 IFET=3ANDPEEK(V)=95ANDPEEK(V+1)=110THENGOTO3040
3028 IFET=2ANDPEEK(V)=210THENGOTO3040
3030 IF(JAND4)=0ANDPEEK(2040)<>15THENPOKEV,15:POKEV,PEEK(V)-5:GOSUB3100
3035 IF(JAND4)=0ANDPEEK(2040)<>11THENPOKEV,11:POKEV,PEEK(V)-5:GOSUB3110
3040 IF(JAND1)=0THENGOSUB5000
3050 IF(JAND2)=0THENGOSUB6000
3055 IFET=2ANDPEEK(V)>155ORPEEK(V+16)=1THENGOTO3010
3060 IFJ=107ANDPEEK(V+16)=0ANDPEEK(V)>=95THENGOSUB7000
3065 IFET=2ANDPEEK(V)=155THENGOTO3010
3070 IFJ=103ANDPEEK(V+16)=0ANDPEEK(V)<200THENGOSUB8000
3080 GOTO3010
3100 REM *****
3101 REM ***V+16 PRUEFEN***
3102 REM *****
3105 FORP=1TO15:NEXTP:POKEV+1,50:POKEV+1,0
3106 IFPEEK(V+31)=60RPEEK(V+31)=140RPEEK(V+30)<>0THENGOTO20000
3110 IFPEEK(V)<40ANDPEEK(V+16)=0THENPOKEV,35
3120 IFPEEK(V)>50ANDPEEK(V+16)=1THENPOKEV,50
3130 IFPEEK(V)=255THENPOKEV+16,1:POKEV,10:RETURN
3140 IFPEEK(V)=0THENPOKEV+16,0:POKEV,250:RETURN
3150 RETURN
5000 REM *****
5001 REM *** MANN KLETTERT HOCH ***
5002 REM *****
5020 IFET=0ANDPEEK(V+16)=0ANDPEEK(V)=35THEN5500
5030 IFET=1ANDPEEK(V+16)=1ANDPEEK(V)=20THEN5600
5040 IFET=2ANDPEEK(V+16)=0ANDPEEK(V)=210THEN5700
5050 IFET=2ANDPEEK(V+16)=0ANDPEEK(V)=155THEN5800
5060 IFET=2ANDPEEK(V)=95THENPOKEV,11:GOTO5900
5081 IFET=3ANDPEEK(V)=95THENPOKEV,11:GOTO5900
5070 IFET=3ANDPEEK(V)=45THEN5950
5100 RETURN
5500 FORKL=214TO182STEP-4:POKEV+1,KL:GOSUB30000:NEXTKL:ET=1:RETURN
5600 FORKL=182TO142STEP-4:POKEV+1,KL:GOSUB30000:NEXTKL:ET=2:POKEV+4,255:RETURN
5700 POKEV,215:FORKL=142TO110STEP-4:POKEV+1,KL:GOSUB30000:NEXTKL:ET=3:RETURN
5800 FORKL=142TO110STEP-4:POKEV+1,KL:GOSUB30000:NEXTKL:ET=3:RETURN
5900 FORKL=PEEK(V+1)TO102STEP-1:POKEV+1,KL:POKEV+15,200-KL:NEXTKL:ET=3:POKEV+15,0:RETURN
5950 FORKL=102TO62STEP-4:POKEV+1,KL:GOSUB30000:NEXTKL:ET=4:RETURN
6000 RETURN
6000 REM *****
6001 REM *** MANN KLETTERT RUNTER**
6002 REM *****
6020 IFET=1ANDPEEK(V+16)=0ANDPEEK(V)=35THEN6500
6030 IFET=2ANDPEEK(V+16)=1ANDPEEK(V)=20THEN6600
6040 IFET=3ANDPEEK(V+16)=0ANDPEEK(V)=215THEN6700
6050 IFET=3ANDPEEK(V+16)=0ANDPEEK(V)=155THEN6800
6060 IFET=2ANDPEEK(V)=95THEN6900
6081 IFET=3ANDPEEK(V)=95THENPOKEV,11:GOTO6900
6070 IFET=4ANDPEEK(V)=45THEN6950
6100 RETURN
6500 FORKL=182TO142STEP4:POKEV+1,KL:GOSUB30000:NEXTKL:ET=0:POKEV+4,255:RETURN
6600 FORKL=142TO102STEP4:POKEV+1,KL:GOSUB30000:NEXTKL:ET=1:POKEV+6,217:POKEV+7,85:RETURN
6700 FORKL=110TO142STEP4:POKEV+1,KL:GOSUB30000:NEXTKL:ET=2:POKEV,215:RETURN
6800 FORKL=110TO142STEP4:POKEV+1,KL:GOSUB30000:NEXTKL:ET=2:RETURN
6900 FORKL=PEEK(V+1)TO142:POKEV+1,KL:POKEV+15,200-KL:NEXTKL:ET=2:POKEV+15,0:RETURN
6950 FORKL=62TO102STEP4:POKEV+1,KL:GOSUB30000:NEXTKL:ET=3:RETURN
6990 RETURN
```

```

7000 REM *****
7001 REM *** MANN SPRINGT LINKS ***
7002 REM *****
7020 S1=PEEK(V):S2=PEEK(V+1)
7030 FORN=1TO2
7040 POKE2040,15:S2=S2-5.5:POKEV+1,S2
7045 FORP=30TO32:POKESI+15,P:NEXTP
7050 POKE2040,15:S1=S1-15:POKEV,S1
7055 FORP=33TO35:POKESI+15,P:NEXTP
7056 ON (ET+1) GOSUB10000,10500,11000,11000
7060 NEXTN
7070 FORN=1TO2
7080 POKE2040,15:S2=S2+5.5:POKEV+1,S2
7085 FORP=35TO33STEP-1:POKESI+15,P:NEXTP
7090 POKE2040,15:S1=S1-10:POKEV,S1
7095 FORP=32TO30STEP-1:POKESI+15,P:NEXTP
7096 ON (ET+1) GOSUB10000,10500,11000,11000
7100 NEXTN
7105 POKE2040,11:POKESI+15,0
7110 RETURN
8000 REM *****
8001 REM *** MANN SPRINGT RECHTS***
8002 REM *****
8020 S1=PEEK(V):S2=PEEK(V+1)
8030 FORN=1TO2
8040 POKE2040,14:S2=S2-5.5:POKEV+1,S2
8045 FORP=30TO32:POKESI+15,P:NEXTP
8050 POKE2040,14:S1=S1+15:POKEV,S1
8055 FORP=33TO35:POKESI+15,P:NEXTP
8056 ON (ET+1) GOSUB10000,10500,11000,11000
8060 NEXTN
8070 FORN=1TO2
8080 POKE2040,14:S2=S2+5.5:POKEV+1,S2
8085 FORP=35TO33STEP-1:POKESI+15,P:NEXTP
8090 POKE2040,14:S1=S1+10:POKEV,S1
8095 FORP=32TO30STEP-1:POKESI+15,P:NEXTP
8096 ON (ET+1) GOSUB10000,10500,11000,11000
8100 NEXTN
8105 POKE2040,13:POKESI+15,0
8110 RETURN
10000 REM *****
10001 REM ***ET=0 SPINNE ***
10002 REM *****
10010 IFPEEK(V+2)>250THENM0=-3
10020 IFPEEK(V+2)<30THENM0=3
10030 POKEV+2,PEEK(V+2)+M0
10499 RETURN
10500 REM *****
10501 REM ***ET=1 FASS ***
10502 REM *****
10510 IFPEEK(V+4)=35THENPOKEV+4,255
10520 POKEV+4,PEEK(V+4)-M1
10999 RETURN
11000 REM *****
11001 REM ***ET=2 FEUER ***
11002 REM *****
11005 IFPEEK(V)<160ANDPEEK(V+16)=0THENPOKEV+7,85:POKEV+6,217:RETURN
11010 Q=PEEK(V+7):R=PEEK(V+6):Q=Q+4:R=R+2:POKEV+21,15
11015 IFPEEK(V+7)>140THENR=255
11020 IFPEEK(V+7)>150THENPOKEV+7,85:POKEV+6,217:GOTO11010
11025 POKEV+7,Q:POKEV+6,R
11499 RETURN
20000 REM *****
20001 REM ***MANN STUERZT AB***
20002 REM *****
20005 FORN=1TO15:POKEV+39,N:POKESI+15,N:FORP=1TO30:NEXTP,N:POKESI+15,0
20010 Q=PEEK(V+1)
20020 FORN=QTO217STEP1
20030 POKEV+1,N:POKESI+15,N:NEXT:POKEV+21,1:POKESI+15,0
20040 POKE2040,251:POKEV+39,12:POKEV+21,15
20045 SP=6:ZE=2:GOSUB1000:PRINT"  MAGDOMNAD  BRUHE IN FRIEDEN!"
20050 FORP=1TO1500:NEXTP:MA=MA-1
20051 IFMA=0THENSP=6:ZE=2:GOSUB1000:PRINT"  G A M E   O V E R   "
8 SPACE

```

programme

```
20052 IFMA=0THENFORP=1TO1000:NEXTP
20053 IFMA=0THENSP=6:ZE=2:GOSUB1000:PRINT"          ":GOTO60000:REM
      SPACE
20055 SP=6:ZE=2:GOSUB1000:PRINT"          ":REM 27 SPACE
20060 POKEV+21,0:POKEV+16,1:POKEV,30:POKEV+37,6:POKEV+38,8:POKEV+39,1
20070 POKE2040,11:POKEV+1,214
20080 POKEV+6,217:POKEV+7,85
20081 POKEV+2,55:POKEV+4,255
20085 POKEV+31,0:POKEV+30,0
20090 POKEV+21,15:ET=0
20091 SP=30:ZE=5:GOSUB1000:PRINT"  "MA
20095 SP=11:ZE=2:GOSUB1000:PRINT"  "PRESS FIRE BUTTON"
20096 J=PEEK(56320):IF(JAND16)<>0THEN20096
20098 GOSUB1000:PRINT"          ":REM 17 SPACE
20100 GOTO3000
30000 REM *****
30001 REM KLETTTERBEWEGUNG
30002 REM *****
30010 IFPEEK(2040)<>255THENPOKE2040,255
30015 FORP=1TO75:NEXTP:POKESI+15,20:POKESI+15,0
30016 ON (ET+1) GOSUB10000,10500,11000,11000
30020 IFPEEK(2040)<>254THENPOKE2040,254
30025 FORP=1TO75:NEXTP:POKESI+15,30:POKESI+15,0
30026 ON (ET+1) GOSUB10000,10500,11000,11000
30030 RETURN
40000 REM *****
40001 REM *** SPRITES LESEN ***
40002 REM *****
40010 FORN=0TO62:READQ:POKE704+N,Q:NEXT
40020 FORN=0TO62:READQ:POKE832+N,Q:NEXT
40030 FORN=0TO62:READQ:POKE896+N,Q:NEXT
40040 FORN=0TO62:READQ:POKE960+N,Q:NEXT
40045 FORN=0TO62:READQ:POKE16320+N,Q:NEXT
40046 FORN=0TO62:READQ:POKE16256+N,Q:NEXT
40047 FORN=0TO62:READQ:POKE16192+N,Q:NEXT
40048 FORN=0TO62:READQ:POKE16128+N,Q:NEXT
40049 FORN=0TO62:READQ:POKE16064+N,Q:NEXT
40050 FORN=0TO62:READQ:POKE16000+N,Q:NEXT
40060 REM *****
40061 REM *** INITIALISIERUNG ***
40062 REM *****
40065 V=53248:POKEV+23,0:POKEV+29,0
40068 POKE2040,11:POKE2041,253:POKE2042,252:POKE2043,250
40067 POKEV+37,6:POKEV+38,8:POKEV+39,1:POKEV+40,0:POKEV+41,3:POKEV+42,2
40068 POKEV,30:POKEV+1,214:POKEV+2,55:POKEV+3,228:
40069 POKEV+4,255:POKEV+5,131:POKEV+6,217:POKEV+7,85
40070 POKEV+30,0:POKEV+31,0
40080 POKEV+16,1:POKEV+21,15:POKEV+28,11
40089 RETURN
40100 REM MANN STEHT LINKS
40101 DATA0,168,0,2,170,0,0,254,0,0,60,0,3,252,0,0,252,0,0,48,0,0,84,0,1,17,0,
40102 DATA17,0,3,17,0,3,81,0,3,5,0,1,85,0,0,84,0,0,20,0,0,20,0,0,20,0,0,20,0,0,
40103 DATA148,0,0,168,0
40200 REM MANN STEHT RECHTS
40201 DATA0,42,0,0,170,128,0,191,0,0,60,0,0,63,192,0,63,0,0,12,0,0,85,0,0,68,6
40202 DATA0,68,64,0,68,192,0,69,192,0,64,192,0,85,64,0,21,0,0,20,0,0,20,0,0,20
40203 DATA0,0,20,0,0,22,0,0,42,0
40300 REM MANN GEHT RECHTS
40301 DATA0,42,0,0,170,128,0,191,0,0,60,0,0,63,192,0,63,0,0,12,0,0,85,0,0,68,6
40302 DATA0,68,64,0,68,64,0,68,64,0,127,64,0,93,64,0,21,0,0,21,0,0,81,64,1,64,
40303 DATA1,0,16,2,0,32,0,128,40
40400 REM MANN GEHT LINKS
40401 DATA0,168,0,2,170,0,0,254,0,0,60,0,3,252,0,0,252,0,0,48,0,0,85,0,1,17,0,
40402 DATA17,0,1,17,0,1,17,0,1,253,0,1,117,0,0,84,0,0,84,0,1,69,0,5,1,64,4,0,6
40403 DATA8,0,128,40,2,128
40500 REM MANN KLETTERT L
40501 DATA0,168,0,2,170,0,14,170,0,12,252,0,4,252,0,4,252,192,4,48,192,5,85,64
40502 DATA5,85,64,1,85,0,1,85,0,1,85,0,1,85,0,1,85,0,0,84,0,0,132,0,0,132,0,0,
40503 DATA0,0,4,0,0,8,0,0,8,0
40600 REM MANN KLETTERT R
40601 DATA0,168,0,2,170,0,2,170,192,0,252,192,12,252,64,12,252,64,4,48,64,5,85
40602 DATA64,5,85,64,1,85,0,1,85,0,1,85,0,1,85,0,1,85,0,0,84,0,0,72,0,0,72,0,6
40603 DATA64,0,0,64,0,0,128,0,0,128,0
40700 REM SPINNE
40701 DATA0,0,0,0,0,0,60,0,0,255,0,3,255,192,3,125,192,3,255,192,0,195,0,0,
```

```

40702 DATA0,2,40,128,8,130,32,34,0,136,34,0,136,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
40703 DATA0,0,0,0,0,0,0,0,0
40800 REM FASS
40901 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
40902 DATA52,0,0,120,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
40900 REM GRAB
40901 DATA0,85,0,0,125,0,0,125,0,0,125,0,0,125,0,5,125,80,7,255,208,7,255,208,7
40902 DATA255,208,5,125,80,0,125,0,0,125,0,0,125,0,0,125,0,2,125,128,10,125,160
40903 DATA42,125,168,170,85,170,170,170,170,170,170,170,170,170,170,170,170,170
40950 REM FEUERBALL
40951 DATA0,0,0,0,10,0,3,13,0,48,220,48,12,92,192,14,166,192,18,251,16,17,238,16
40952 DATA7,158,84,2,215,0,1,119,0,0,212,0,0,16,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
40953 DATA0,0,0,0,0,0,0,0,0
50000 REM *****
50001 REM *** INFO MADDOMNAD***
50002 REM *****
50010 ZE=8:SP=0
50020 FORN=ZE TO ZE+14
50030 GOSUB1000
50040 PRINT"                                ":ZE=ZE+1:NEXTN:REM 40 SPAC
E
50050 ZE=6:SP=05:GOSUB1000
50060 PRINTCHR$(5)"INFORMATION M A G D O M N A D"
50070 PRINT"Sie SIE BEFINDEN SICH IM KELLER EINER BURG"
50080 PRINT"          VERSUCHEN SIE ZU ENTKOMMEN"
50090 PRINT" GESPIELT WIRD MIT DEM JOYSTICK PORT 2"
50100 PRINT"Drücken DRUCKEN SIE DEN FEUERKNOPF"
50200 J=PEEK(56320)
50210 IF(JAND16)=0THEN RETURN
50300 GOTO50200
60000 REM *****
60001 REM *** SPIELAUSWERTUNG ***
60002 REM *****
60010 SP=3:ZE=2:GOSUB1000
60011 T=VAL(TI#):T=T*1.5
60015 SC=1000-T:SC=SC*10
60016 IFMA=0THENSC=VAL(TI#)*10:IFSC>5000THENSC=5000
60017 PRINT"YOUR SCORE"SC+"(100*MA)"BONUS":SC=SC+(100*MA)
60018 SC#=STR$(SC):IFVAL(SC#)>VAL(HC#)THENHC#=SC#
60020 IFLEN(HC#)=3THENHC#="0"+HC#
60022 FORN=1TO15:ZE=12:SP=2:GOSUB1000:PRINTHC#":POKES1+15,50:POKE646,N
60023 POKES1+15,25:NEXTN:POKES1+15,0:POKE646,1
60025 FORN=1TO10:FORM=1TO50:POKES1+15,M:NEXTM,N:POKES1+15,0
60026 SP=4:ZE=21:GOSUB1000:PRINT"PRESS FIRE BUTTON FOR NEW GAME"
60030 J=PEEK(56320):IF(JAND16)<>0THENB0030
60050 REM *****
60051 REM ***NEW GAME - RESET***
60050 REM *****
60100 SP=4:ZE=21:GOSUB1000:PRINT"                                ":REM 30 SPACE
60110 SP=8:ZE=2:GOSUB1000:PRINT"                                ":REM 28 SPACE
60120 POKEV+21,0:POKEV,30:POKEV+1,214:POKEV+16,1:POKEV+2,55:POKEV+4,255
60130 POKEV+6,217:POKEV+7,85:POKE2040,11
60140 ET=0:MA=3:POKEV+21,15:POKEV+30,0:POKEV+31,0
60145 SP=30:ZE=5:GOSUB1000:PRINT" "MA
60150 TI#="000000"
60160 GOTO0000
READY.

```

Der ideale Schönschreibdrucker für Ihren C 64* von OLYMPIA electronic compact 2

- eingebautes Interface für seriellen Port
- Adresse 4 oder 5 einstellbar
- Schreibmaschine mit Korrekturspeicher
- ON-OFF-Line Taste trennt Betriebsarten
- Zeichenabstand (10, 12 und 15 Z/Zoll) und Zeilenabstand (1, 1 1/2 und 2 zeilig) vom Rechner einstellbar
- ohne Tastatur als compact 2 RO
- 100 Zeichentypenrad
- 14 Zeichen/Sekunde
- Leerstellenunterdrückung
- sofort lieferbar
- enorm preiswert
- viele Typenräder, Gewebe-Karbon-, Multikarbonbänder
- bundesweiter Werkkundendienst

* auch für andere Rechner mit Centronics, V.24 und IEC-Bus.

Alle OLYMPIA-Drucker (ESW 102, 103 u. 3000) für den C 64 lieferbar

 -Datentechnik

Telemannstraße 18
7250 Leonberg (Höfingen)
☎ 07152/6305



Bomber

für den Commodore 64

Sie müssen versuchen, die radioaktiv verseuchten Weg schicken. Da Ihr Flugzeug immer tiefer sinkt, Hochhäuser einer Stadt mit Ihren Bomben zu zerstören. Sie haben tausende von Bomben, die ein Haus so Sie an einem radioaktiv verseuchten Haus. Eine lange zerstören, bis Sie die nächste Bombe auf den Bombe wird mit "F1"-Taste abgeworfen.

```

0 POKES3280,14:POKES3281,14
1 READA$: IFA$="G" THEN GOTO 100
2 GOSUB 100000:POKE 49152+S,C:S=S+1:GOTO 1
10 DATA 0E,DC,29,FE,8D,0E,DC,AS,01,29,FB,85,01,A9,D0,85,23,A9,00,85,22,85,24
12 DATA 9,30,85,25,A0,00,B1,22,91,24,AS,22,C9,FF,F0,07,E6,22,E6,24,4C,1E,C0,AS
14 DATA 23,C9,D9,F0,0D,A9,00,85,22,85,24,E6,23,E6,25,4C,1E,C0,AS,01,09,04,85,01
16 DATA 0E,DC,09,01,8D,0E,DC,AD,18,D0,29,F0,09,0C,8D,18,D0,60,0
100 PRINT "##### BOMBER"
110 PRINT "DAS BELIEBTE SPIEL FUER JUNG UND ALT":PRINT ""
120 PRINT "
130 PRINT "      ADAPTED FOR THE      ":PRINT
140 PRINT "      COMMODORE 64      ":PRINT
150 PRINT "      BY      ":PRINT
160 PRINT "      CHRISTIAN LAENTZSCH      ":PRINT
170 PRINT "
180 PRINT:PRINT "#####PRESS 'P' FOR INSTRUCTIONS"
190 SI=54272:POKESI+24,15:POKESI+5,170:POKESI+6,170:POKESI+4,35
200 POKESI+12,170:POKESI+13,170:POKESI+11,35
210 POKESI+19,170:POKESI+20,170:POKESI+18,35
220 X=1
230 POKESI+1,X:POKESI+8,X-1:POKESI+15,X+1:X=X+1:IFX=253 THEN GOTO 310
290 GETA$: IFA$="P" THEN 310
300 GOTO 230
310 PRINT "":POKESI+4,32:POKESI+11,32:POKESI+18,32
320 PRINT "#####TAB(17)#####BOMBER"
330 PRINT "#####YOU HAVE TO DESTROY THE RADIOACTIVE"
340 PRINT "#####CONTAMINATED CITY WITH YOUR 'BOMBER'"
350 PRINT "#####TO SCRAB THE CITY YOU HAVE THOUSENDS"
360 PRINT "#####OF SMART BOMBS WHICH WILL DASH THE "
370 PRINT "#####OLD HOUSES AS LONG AS NO OTHER BOMB"
380 PRINT "#####IS ON ITS' WAY ."
390 PRINT "#####PRESS 'P' TO BEGIN":PRINT "#####F1 = FIRE"
400 GETA$: IFA$="P" THEN 400
410 PRINT "":SYS 49152:FORX=0 TO 39:READ D:POKE 12520+X,D:NEXT X:U=500
415 D=3
420 FORR=1 TO 25
430 A=INT(RND(1)*38)+1184:A=A+40:IFPEEK(A+40)=30 THEN 430
440 FORT=1 TO 20
450 POKE 55296+(A-1024),0:POKER,30
460 A=A+40
470 NEXT T:NEXTR
550 J=1064:L=0:K=0:F=0:M=32:FA=55336
555 PRINT "#####PUNKTE"PU"#####J"D
560 GETC$:IFC$=" " THEN K=1:POKEJ+L,M:L=F
570 POKEJ+F-1,M:POKEJ+F,29:POKEFA+F,1:IFK=1 THEN GOSUB 10200
580 F=F+1:IFK=0 THEN L=F
590 IFPEEK(J+F)=30 THEN GOTO 10300
600 IFPU=X THEN GOTO 11000
610 GOTO 555
10000 A=ASC(A$)-48:IFA>16 THEN A=A-7
10010 B=ASC(RIGHT$(A$,1))-48:IFB>16 THEN B=B-7
10020 C=A*16+B:RETURN
10200 L=L+40:IFPEEK(J+L)=30 THEN PU=PU+1:POKEJ+L-40,32:POKEJ+L,31:POKEFA+L,1
10210 IFJ+L>2023 THEN K=0:POKEJ+L-40,M:RETURN

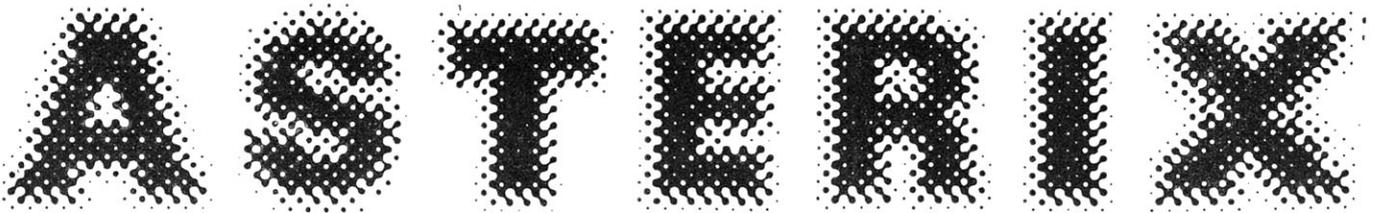
```

```

10220 POKEJ+L-40,M:POKEJ+L,31:POKEFA+L,1:RETURN
10250 DATA16,24,92,127,127,92,24,16,255,153,153,153,255,153,153,153
10260 DATA0,24,255,126,60,24,0,0,0,0,0,0,0,0,68,0,89,192,90,18,8,66
10300 POKEJ+L,M:POKEF+J-1,M:POKEJ+F,33:POKEFA+F,2
10310 POKESI+24,31:POKESI+5,62:POKESI+6,0:POKESI+22,1:POKESI+23,1
10320 POKESI+1,2:POKESI+4,129
10330 FORX=1TO6000:NEXT
10340 POKESI+4,4:POKEJ+F,M
10350 PU=PU+1:IFPU=XUTHEN11000
10360 D=D-1:IFD=0THEN10400
10370 GOT0550
10400 PRINT"#####POINTS : "PU:PRINT"#####YOUR RANK IS ";
10410 IFPU<150THENPRINT"LOOSER":GOTO10500
10420 IFPU<500THENPRINT"TOURIST":GOTO10500
10430 IFPU<999THENPRINT"COOK":GOTO10500
10440 IFPU<1250THENPRINT"CHIEFCOOK":GOTO10500
10450 IFPU<2000THENPRINT"EXPERT":GOTO10500
10460 IFPU<2750THENPRINT"SEMIPRO":GOTO10500
10470 IFPU<3499THENPRINT"PRO":GOTO10500
10480 IFPU<4750THENPRINT"SUPERPRO":GOTO10500
10490 IFPU>4750THENPRINT"HERO"
10500 PRINT"#####TRY AGAIN":PRINT"#####PRESS ANY KEY":POKE198,0
10510 GETA$:IFA#=""THEN10510
10520 PRINT"J":PU=0:XU=500:GOTO415
11000 PRINT"#####*BONUSGAME*#####"
11005 FOREE=54272TO54300:POKEEE,0:NEXTEE
11010 SI=54272:POKESI+4,4:POKESI+4,0
11020 POKESI+24,15:POKESI+5,170:POKESI+6,170
11030 POKESI+4,21
11040 FORAS=1TO255:POKESI+1,AS:POKESI+15,AS/7:NEXTAS:PRINT"J":XU=XU+500
11050 POKESI+4,8:D=D+1:GOTO420

```

READY.



für den Commodore 64

Asterix benötigt alle 8 Sprites. Alle Sprites sollen zu einem Bild zusammengesetzt werden. Das Spielfeld besteht aus einer 3*3 Matrix. Es können immer nur die Bilder verschoben werden, die direkt neben dem freien Raum liegen. Zum Verschieben stehen die Tasten 1-9 zur Verfügung.

Die Sprites in die richtige Reihenfolge zu bringen, ist gar nicht so einfach, wenn man bedenkt, daß es sich nur um eine 3*3 Matrix handelt.



programme

```
100 REM *****
110 REM ** COPYRIGHT **
120 REM ** BY **
130 REM ** HEINZ PAUDITZ **
160 REM *****
170 POKE53281,5:POKE53280,5:HI=100
180 REM *** PROGRAMMBESCHREIBUNG ***
190 PRINT"STELLEN SIE BITTE DIE KLEINGRAPHIK
200 PRINT"RICHTIG ZUSAMMEN. ES HANDELT SICH UM
210 PRINT"ASTERIX. ALS ZUSAEZTLICHE SCHWIERIGKEIT";
220 PRINT"AM ANFANG IST DIE GRAPHIK BEREITS
230 PRINT" DURCHMISCHT.
240 PRINT"VERSCHIEBEN DER SPRITES (3*3 MATRIX)
250 PRINT"MIT:
260 PRINT" | | |
270 PRINT" | 4-5-6
280 PRINT" | | |
290 PRINT" | 7-8-9
300 PRINT"BITTE TASTE DRUECKEN"
310 POKE198,0:WAIT198,1
320 REM *** UMRÄHMUNG VON ASTERIX ***
330 PRINT"SPC(3)"
340 FORI=0TO14
350 PRINT" "
360 NEXT
370 PRINT" "
380 PRINT"BESTE ANZAHL DER VERSUCHE: ";HI
390 PRINT"SPC(28)"1-2-3"
400 PRINTSPC(28)"| | |"
410 PRINTSPC(28)"4-5-6"
420 PRINTSPC(28)"| | |"
430 PRINTSPC(28)"7-8-9"
440 REM *** VARIABLEN DEFINIEREN ***
450 U=11:ZB=3:X=0
460 V=53248
470 REM *** RICHTIGE ZUSAMMENSTELLUNG VON ASTERIX ***
480 C(0)=57:C(1)=105:C(2)=57:C(3)=105:C(4)=153:C(5)=57:C(6)=105:C(7)=153
490 D(0)=80:D(1)=80:D(2)=122:D(3)=122:D(4)=122:D(5)=164:D(6)=164:D(7)=164
500 REM *** LESEN DER 8 SPRITES ***
510 FORE=200TO207:POKE2040+E-200,E:FORD=64*ET064*E+62:READS
520 POKED,C:NEXT:POKEV+39+E-200,U:NEXT
530 POKEV+21,255
540 REM *** FARBEN DER SPRITES ***
550 POKEV+39,0:POKEV+40,0:POKEV+41,0:POKEV+42,0:POKEV+43,0:POKEV+44,0:POKEV+45,0
560 POKEV+46,0:POKEV+23,255:POKEV+29,255
570 GOSUB2410:REM *** VERTAUSCHEN DER SPRITES ***
580 REM *** HAUPTPROGRAMM ***
590 POKEV,A(0):POKEV+1,B(0)
600 POKEV+2,A(1):POKEV+3,B(1)
610 POKEV+4,A(2):POKEV+5,B(2)
620 POKEV+6,A(3):POKEV+7,B(3)
630 POKEV+8,A(4):POKEV+9,B(4)
640 POKEV+10,A(5):POKEV+11,B(5)
650 POKEV+12,A(6):POKEV+13,B(6)
660 POKEV+14,A(7):POKEV+15,B(7)
670 REM *** ABFRAGE OB RICHTIGE ZUSAMMENSTELLUNG ***
680 FF=0
690 FORI=0TO7
700 IFC(I)=A(I)ANDD(I)=B(I)THENFF=FF+1
710 IFFF=8THEN2290
720 NEXT
730 PRINT"SPC(25)"VERSUCHE:";X
740 REM *** TASTENABFRAGUNG ***
750 GETA$
760 IFA$=""THEN750
770 IFA$="1"THENZA=1:GOSUB880
780 IFA$="2"THENZA=2:GOSUB990
790 IFA$="3"THENZA=3:GOSUB1100
800 IFA$="4"THENZA=4:GOSUB1210
810 IFA$="5"THENZA=5:GOSUB1320
820 IFA$="6"THENZA=6:GOSUB1430
830 IFA$="7"THENZA=7:GOSUB1540
840 IFA$="8"THENZA=8:GOSUB1650
850 IFA$="9"THENZA=9:GOSUB1760
860 GOT0580
```

```

870 REM *** VERSCHIEBUNG DER SPRITES ***
880 IFZB<>2ANDZB<>4THENRETURN
890 ZD=2:ZC=2:X=X+1
900 IFZB<4THENZD=0:GOTO920
910 IFZB<7THENZD=1
920 IFZB=10RZB=40RZB=7THENZC=0:GOTO940
930 IFZB=20RZB=50RZB=8THENZC=1
940 ZB=ZA
950 FORI=0TO7
960 IFA(I)=57ANDB(I)=80THEN:A(I)=57+ZC*48:B(I)=80+ZD*42:RETURN
970 NEXT
980 RETURN
990 IFZB<>1ANDZB<>3ANDZB<>5THENRETURN
1000 ZD=2:ZC=2:X=X+1
1010 IFZB<4THENZD=0:GOTO1030
1020 IFZB<7THENZD=1
1030 IFZB=10RZB=40RZB=7THENZC=0:GOTO1050
1040 IFZB=20RZB=50RZB=8THENZC=1
1050 ZB=ZA
1060 FORI=0TO7
1070 IFA(I)=105ANDB(I)=80THEN:A(I)=57+ZC*48:B(I)=80+ZD*42:RETURN
1080 NEXT
1090 RETURN
1100 IFZB<>2ANDZB<>6THENRETURN
1110 ZD=2:ZC=2:X=X+1
1120 IFZB<4THENZD=0:GOTO1140
1130 IFZB<7THENZD=1
1140 IFZB=10RZB=40RZB=7THENZC=0:GOTO1160
1150 IFZB=20RZB=50RZB=8THENZC=1
1160 ZB=ZA
1170 FORI=0TO7
1180 IFA(I)=153ANDB(I)=80THEN:A(I)=57+ZC*48:B(I)=80+ZD*42:RETURN
1190 NEXT
1200 RETURN
1210 IFZB<>1ANDZB<>5ANDZB<>7THENRETURN
1220 ZD=2:ZC=2:X=X+1
1230 IFZB<4THENZD=0:GOTO1250
1240 IFZB<7THENZD=1
1250 IFZB=10RZB=40RZB=7THENZC=0:GOTO1270
1260 IFZB=20RZB=50RZB=8THENZC=1
1270 ZB=ZA
1280 FORI=0TO7
1290 IFA(I)=57ANDB(I)=122THEN:A(I)=57+ZC*48:B(I)=80+ZD*42:RETURN
1300 NEXT
1310 RETURN
1320 IFZB<>2ANDZB<>4ANDZB<>6ANDZB<>8THENRETURN
1330 ZD=2:ZC=2:X=X+1
1340 IFZB<4THENZD=0:GOTO1360
1350 IFZB<7THENZD=1
1360 IFZB=10RZB=40RZB=7THENZC=0:GOTO1380
1370 IFZB=20RZB=50RZB=8THENZC=1
1380 ZB=ZA
1390 FORI=0TO7
1400 IFA(I)=105ANDB(I)=122THEN:A(I)=57+ZC*48:B(I)=80+ZD*42:RETURN
1410 NEXT
1420 RETURN
1430 IFZB<>3ANDZB<>5ANDZB<>9THENRETURN
1440 ZD=2:ZC=2:X=X+1
1450 IFZB<4THENZD=0:GOTO1470
1460 IFZB<7THENZD=1
1470 IFZB=10RZB=40RZB=7THENZC=0:GOTO1490
1480 IFZB=20RZB=50RZB=8THENZC=1
1490 ZB=ZA:X=X+1
1500 FORI=0TO7
1510 IFA(I)=153ANDB(I)=122THEN:A(I)=57+ZC*48:B(I)=80+ZD*42:RETURN
1520 NEXT
1530 RETURN
1540 IFZB<>4ANDZB<>8THENRETURN
1550 ZD=2:ZC=2
1560 IFZB<4THENZD=0:GOTO1580
1570 IFZB<7THENZD=1
1580 IFZB=10RZB=40RZB=7THENZC=0:GOTO1600
1590 IFZB=20RZB=50RZB=8THENZC=1
1600 ZB=ZA
1610 FORI=0TO7

```

programme

```
1620 IFA(I)=57ANDB(I)=164THENA(I)=57+ZC*48:B(I)=80+ZD*42:RETURN
1630 NEXT
1640 RETURN
1650 IFZB<>5ANDZB<>7ANDZB<>9THENRETURN
1660 ZD=2:ZC=2:X=X+1
1670 IFZB<4THENZD=0:GOTO1690
1680 IFZB<7THENZD=1
1690 IFZB=10RZB=40RZB=7THENZC=0:GOTO1710
1700 IFZB=20RZB=50RZB=8THENZC=1
1710 ZB=ZA
1720 FORI=0TO7
1730 IFA(I)=105ANDB(I)=164THENA(I)=57+ZC*48:B(I)=80+ZD*42:RETURN
1740 NEXT
1750 RETURN
1760 IFZB<>6ANDZB<>8THENRETURN
1770 ZD=2:ZC=2:X=X+1
1780 IFZB<4THENZD=0:GOTO1800
1790 IFZB<7THENZD=1
1800 IFZB=10RZB=40RZB=7THENZC=0:GOTO1820
1810 IFZB=20RZB=50RZB=8THENZC=1
1820 ZB=ZA
1830 FORI=0TO7
1840 IFA(I)=153ANDB(I)=164THENA(I)=57+ZC*48:B(I)=80+ZD*42:RETURN
1850 NEXT
1860 RETURN
1870 REM *** SPRITE 1 ***
1880 DATA124,0,0,131,192,0,64,62,0,48,1,192,14,0,60,1,249,3,0,17,0,0,24
1890 DATA144,0,6,72,0,1,244,0,0,34,0,0,33,0,0,17,0,0,15,0,0,1,0,0,1,0
1900 DATA0,3,0,0,4,0,0,8,0,0,7,0,0,1
1910 REM *** SPRITE 2 ***
1920 DATA28,0,0,34,0,0,33,0,0,16,192,0,8,96,0,196,16,0,59,8,0,7,132,0
1930 DATA1,194,0,0,161,0,0,192,132,224,160,68,0,144,36,1,12,36,2,10,84
1940 DATA4,8,108,207,8,104,120,236,44,232,20,42,144,14,114,16,31,97
1950 REM *** SPRITE 3 ***
1960 DATA0,0,3,0,0,2,0,0,6,0,15,196,0,16,62,0,96,97,0,65,128,0,159,0,0
1970 DATA161,120,1,78,134,1,146,134,0,35,9,0,33,8,0,65,9,0,65,9,0,65,6
1980 DATA0,130,130,0,130,102,0,133,120,0,133,8,0,69,8
1990 REM *** SPRITE 4 ***
2000 DATA160,249,208,199,177,184,24,172,120,32,170,56,64,110,88,130,153
2010 DATA92,130,185,187,67,57,144,67,57,208,82,58,240,82,26,232,172,11
2020 DATA255,84,4,0,152,4,128,19,196,60,7,132,64,7,8,0,6,244,0,3,4,0,12
2030 DATA6,3,24,2,0
2040 REM *** SPRITE 5 ***
2050 DATA128,0,0,128,0,112,64,0,80,64,0,80,128,0,112,128,0,80,0,0,0,0
2060 DATA0,48,0,0,64,0,0,32,0,0,16,252,0,96,2,0,0,1,0,112,0,128,32,0,128
2070 DATA32,0,128,32,0,128,32,1,0,0,1,0,112,2,0,64
2080 REM *** SPRITE 6 ***
2090 DATA0,36,132,0,20,132,0,10,68,0,6,62,0,0,1,0,0,2,0,0,4,0,0,4,0,0
2100 DATA4,0,0,8,0,0,8,0,0,9,0,0,10,0,0,14,0,0,6,0,0,6,0,0,0,0,0,0
2110 DATA0,0,0,0,0,0,0
2120 REM *** SPRITE 7 ***
2130 DATA32,4,128,64,5,64,128,9,48,128,31,15,0,63,2,0,127,2,0,255,2,3
2140 DATA126,2,13,125,2,53,121,2,196,113,4,4,97,4,132,67,4,131,133,8,128
2150 DATA201,168,64,50,176,32,195,192,31,7,128,0,0,0,0,0,0,0,0,0
2160 REM *** SPRITE 8 ***
2170 DATA4,0,96,8,0,64,48,0,112,192,0,0,0,0,112,0,0,80,0,0,112,0,0,96
2180 DATA0,0,80,0,0,0,0,0,32,0,0,32,0,0,32,0,0,32,0,0,32,0,0,0,0,0,80
2190 DATA0,0,80,0,0,32,0,0,80,0,0,80
2200 REM ** VERTAUSCHEN DER SPRITES **
2210 DATA5,3,4,0,1,6,2,7,-1,6,2,3,1,4,7,0,5,-1,2,3,5,0,4,6,7,1,-1
2220 DATA3,5,2,7,4,6,1,0,-1
2230 DATA2,3,7,5,1,6,4,0,-1,7,2,4,6,1,3,0,5,-1,0,7,2,6,1,4,3,5,-1
2240 DATA7,3,1,0,6,2,4,5,-1
2250 DATA6,0,4,1,2,7,5,3,-1,0,2,6,4,5,7,3,1,-1,5,6,3,2,4,0,7,1,-1
2260 DATA6,4,7,3,1,5,2,0,-1
2270 DATA6,1,7,4,2,5,0,3,-1,1,7,6,3,2,5,4,0,-1,1,5,7,4,2,6,0,3,-1
2280 REM *** ENDE ***
2290 PRINT"XXXXXXXXXX"SPC(25)"SIE HABEN
2300 PRINTSPC(25)" "
2310 PRINTSPC(25)"ES GESCHAFFT "
2320 PRINTSPC(25)" "
2330 PRINTSPC(25)"IN ";X;" "
2340 PRINTSPC(25)"VERSUCHEN.
2350 PRINTSPC(25)"NOCHMAL ?
2360 IFHI>XTHENHI=X
```

```

2370 GETA$: IFA$="J"THENRESTORE:GOTO330
2380 IFA$="N"THENPOKEV+21,0:PRINT"NO BIS ZUM NAECHSTEN MAL !":END
2390 GOTO2370
2400 REM ** VERTAUSCHEN DER SPRITES **
2410 TR=INT(RND(TI)*15)*6
2420 IFTR=0THEN2460
2430 FORI=0TOTR:READP:NEXT
2440 READP:IFP=-1THEN2460
2450 GOTO2440
2460 FORI=0T07:READP
2470 A(I)=C(P)
2480 B(I)=D(P)
2490 NEXT
2500 RETURN
READY.

```

THINK

für den Commodore 64

Bei dem Spiel THINK (eine Version von Superhirn) kommt es darauf an, eine vom Computer erdachte Farbkombination in möglichst wenig Versuchen zu erraten. Nach ca. 30 Sekunden erscheint das Titelbild. Wenn Sie nun eine Taste drücken, wird nach dem Schwierigkeitsgrad (1 oder 2) gefragt. Bei Schwierigkeitsgrad 1 muß man eine Kombination von 5 Steckern, die aus den Farben weiß, schwarz, rot, violett und türkis bestehen, erraten. Bei Schwierigkeitsgrad 2 kommen noch die Farben grün und blau dazu. Um die Stecker zu setzen, geben Sie jeweils den Anfangsbuchstaben der entsprechenden Farbe ein, achten Sie aber darauf, daß ein gesetzter Stecker nicht zurückgenommen werden kann. Nachdem der fünfte Stecker gesetzt ist, folgt auf der rechten Bildschirmhälfte die Bewertung wie beim Original-Spiel (schwarzer Stecker für richtige Farbe und richtige Position,

weiße Stecker für richtige Farbe und falsche Position). Wenn die Kombination des Spielers mit der des Computers übereinstimmt, ist das Spiel zu Ende und es folgt die Bewertung der Leistung.

Programmaufbau:

80-90: Dimensionieren und festlegen der Variablen
100-146: Neue Zeichen und Musikdatas einlesen, Titelbildaufruf
150-162: Tastaturabfrage
990-1900: Stecker zeichnen und Ton ausgeben
2400-2700: Bewertung der Kombination
2800-3900: Schwarze Stecker zeichnen und Ton erzeugen
4500-9900: Weiße Stecker zeichnen und Ton erzeugen
9950-9999: Unterprogramm aufrufen, wenn Kombination erraten
11000-12000: Spielfeld zeichnen

14990-16100: Frage nach neuem Spiel
20000-22400: Wahl des Schwierigkeitsgrades
60000-60500: Datas für neue Zeichen
61000-62040: Datas für Lied 1 und 2

Variablenliste:

ZU: Nummer des Zuges
SI: Basisadresse des SID
FL, FH, TL, TH, W, H, A, L: Einzelne Soundadressen
SW: Anzahl schwarzer Stecker
X(I), Y(I), D(I): Noten für erstes Lied
X2(I), Y2(I), D2(I): Noten für zweites Lied
A\$: Buchstabe der aktuellen Farbe
F(I): Farbcode der aktuellen Farbe
PS: Position des nächsten Steckers
S(I): Kombination des Computers
RV, RF: Positions- u. Farbangaben für das Spielfeld
GR, GR\$: Schwierigkeitsgrad
I, RR, B, Q, AA, BB: Zählvariable

programme

```
5 REM *****
10 REM ** **
12 REM ** THINK **
14 REM ** **
16 REM ** EINE VERSION DES SPIELES **
18 REM ** **
20 REM ** SUPERHIRN **
22 REM ** **
23 REM *****
25 REM ** **
26 REM ** VON: **
28 REM ** OLAF AMBLANK **
34 REM ** **
37 REM *****
40 :
50 POKE649,1:REM *TASTATURPUFFER AUF 1*
80 DIMX(30),Y(30),D(30),X2(15),Y2(15),D2(15)
90 ZU=1:SI=54272:FL=SI:FH=SI+1:TL=SI+2:TH=SI+3:W=SI+4:H=SI+6:L=SI+24
100 POKE53281,7:POKE53280,9
103 PRINT"*****"SPC(7)"BITTE EINEN MOMENT GEDULD":REM"*** P=COMMODORE+2"
110 GOSUB60000
111 A=SI+5:SW=0
115 FORI=1TO30:READX(I),Y(I),D(I):NEXT
116 FORI=1TO15:READX2(I),Y2(I),D2(I):NEXT
118 GOSUB20000
120 FORR=1TO5
125 IFGR=1THEN132
130 S(R)=INT(RND(1)*7)+0:GOTO135
132 S(R)=INT(RND(1)*5)+0
138 S2(R)=S(R)
140 NEXT
146 SW=0:PS=801
147 GOSUB11000
148 FORI=1TO5:REM ***** 5 EINGABEN *****
149 REM***** EINGABE *****
150 GETA$:IFGR=1ANDAS$<>"S"ANDAS$<>"W"ANDAS$<>"R"ANDAS$<>"T"ANDAS$<>"V"THEN150
151 IFA$<>"S"ANDAS$<>"W"ANDAS$<>"T"ANDAS$<>"V"ANDAS$<>"G"ANDAS$<>"B"ANDAS$<>"R"THEN150
152 IFA$="S"THEN F(I)=0
153 IFA$="R"THEN F(I)=2
154 IFA$="W"THEN F(I)=1
156 IFA$="T"THEN F(I)=3
158 IFA$="V"THEN F(I)=4
160 IFA$="G"THEN F(I)=5
162 IFA$="B"THEN F(I)=6
990 REM***** KREISE POKEN *****
1000 POKE1024+PS,65:POKE55296+PS,F(I)
1010 POKE1025+PS,66:POKE55297+PS,F(I)
1020 POKE1026+PS,67:POKE55298+PS,F(I)
1030 POKE1064+PS,68:POKE55336+PS,F(I)
1040 POKE1065+PS,69:POKE55337+PS,F(I)
1050 POKE1066+PS,70:POKE55338+PS,F(I)
1060 POKE1104+PS,71:POKE55376+PS,F(I)
1070 POKE1105+PS,72:POKE55377+PS,F(I)
1080 POKE1106+PS,73:POKE55378+PS,F(I)
1100 PS=PS+4
1150 REM ***** TON BEIM SETZEN *****
1200 POKEL,15:POKEA,16+9:POKEH,4*16+4:POKEFH,29:POKEFL,69:POKEW,17
1210 FORM=1TO300:NEXT:POKEW,0:POKEA,0:POKEH,0
1900 NEXT
2400 PS=824
2450 REM ***** VERGLEICH FUER *****
2452 REM ***** SCHWARZE STECKER *****
2500 FORB=1TO5
2510 IF F(B)=S(B)THENS(B)--1:F(B)--2:SW=SW+1:GOSUB3000
2520 NEXT
2530 REM ***** VERGLEICH FUER *****
2531 REM ***** WEISSE STECKER *****
2532 FORAA=1TO5:FORBB=1TO5
2550 :
2551 IFF(AA)=S(BB)THENGOSUB5000
2580 NEXT:NEXT
2583 :
2590 GOTO4500
2700 PS=824
2800 REM***** SCHWARZE UND WEISSE*****
2801 REM***** STECKER POKEN *****
3000 POKE1024+PS,74:POKE55296+PS,0
3010 POKE1025+PS,75:POKE55297+PS,0
3020 POKE1026+PS,76:POKE55298+PS,0
3030 POKE1064+PS,77:POKE55336+PS,0
3040 POKE1065+PS,78:POKE55337+PS,0
3050 POKE1066+PS,79:POKE55338+PS,0
3060 POKE1104+PS,80:POKE55376+PS,0
3070 POKE1105+PS,81:POKE55377+PS,0
3080 POKE1106+PS,82:POKE55378+PS,0
3500 PS=PS+3
```


Teil 5

Logik im Basic-Programm

Wie man Systematik in sein Selbstgestricktes bringt

Während wir in den ersten Folgen stets am Rechner arbeiteten und anhand kleinerer Unternehmungen erste Erfahrungen im Programmieren sammelten, müssen wir heute ein bißchen Theorie einschalten, um uns auf die vor uns liegenden größeren Aufgaben vorzubereiten. Wenn wir bei der Lösung komplexerer Probleme nicht wahllos an's Werk gehen wollen, müssen wir Methoden ins Spiel bringen, die den Aufbau unserer Listings von Anfang an systematisieren.

Ich gebe zu, daß sich das so dahin gesagt recht schwierig anhört, doch werden wir sehen, daß richtig aufgezümt das Pferd schon wesentlich bessere Startchancen haben wird.

Wir wollen uns daher ein paar Gedanken über das Programmieren machen, weil man nur dasjenige gut tun kann, was zuvor auch verstanden ist. Sehen wir uns also einmal an, wie ein Programm entsteht.

1. Das Problem:

Ohne Problem, ohne Fragestellung, wird in der Regel kein Programm zustande kommen: zuerst ist da eine Aufgabe, die es mit Hilfe des Computers zu lösen gilt. Die Vorbereitung dieser Lösung ist ein geistiger Prozess, der umso langwieriger ist, je weiter das Problem im menschlichen, kreativen Bereich angesiedelt ist, d.h. je weniger es sich durch mathematische Formalismen beschreiben läßt.

Da aber ein Computer nicht etwa ein künstliches, doch von höherer Macht mit übermenschlicher Intelligenz ausgestattetes Wesen ist (wie mancher Laie auch heute noch meint), sondern ein Apparat, der lediglich unter Zuhilfenahme elementarer logischer Verknüpfungen eine Folge von Primitiv-Handlun-

gen ausführt, ist es Aufgabe des programmierenden Menschen, sich erstens über das Problem und seine Strukturen bewußt zu werden und es zweitens sprachlich zu übersetzen. Somit dürfte die erste Frage des Programmierens die nach der Art des zu bewältigenden Problems sein.

Er stellt sich dann die Erwartungen zusammen, die er in die Lösung durch den Rechner setzt: Was soll mein Programm leisten? – Was muß es alles machen, damit mein Problem gelöst bzw. gemindert wird?

2. Der Plan:

Nun wird in Form einer Skizze oder eines Struktogrammes ein Plan erstellt, der die verschiedenen noch ein wenig durcheinander gewürfelten Punkte zu einem logischen Ganzen ordnet. Dabei ist es durchaus nicht möglich, einzelne Bezirke abzuspalten, um sie quasi als Sonderfälle in späteren Teilprogrammen getrennt zu bearbeiten. Dieser Plan kann zu einer detaillierten Beschreibung, aber

auch in einer Grafik oder einem Struktogramm bestehen.

3. Der Algorithmus:

Aus dem im Plan enthaltenen Lösungsentwurf wird dann durch Präzisierung und Abfassung in einer dem Computer verständlichen Sprache der eigentliche Programmablauf entwickelt, d.h. die Folge aller vom Rechner verwertbaren Programmierbefehle.

Dabei ist der Erfolg des jeweiligen Programmes maßgebend durch die Qualität des Programmierens, aber auch der Maschine bestimmt. Wir können den Algorithmus als Gradmesser für die Kompromißfähigkeit des jeweils Programmierenden ansehen – Kompromiß dergestalt, daß vom menschlichen Partner verlangt werden muß, daß dieser mit der Darstellung der Befehlsfolge auf die Ebene der Maschine "Computer" hinabsteigen kann.

4. Die Eingabe

Das Eintippen eines Programmes ist sicher der lästigste

Teil der gesamten Programmierarbeit. Dennoch stellt es einen wesentlichen Schritt dar, nicht zuletzt deshalb, weil sich während des oft langwierigen und monotonstumpfsinnigen Vorganges leicht Fehler einschleichen, sei es, daß der Programmierer die Tastatur nicht beherrscht, sei es, daß es an der nötigen Konzentration mangelt.

5. Die Prüfung:

Programme, die geschrieben und eingetippt werden, gestartet und dann fehlerfrei auf Anhieb abgearbeitet werden, sind höchst selten. In 99,9% aller Fälle beginnt nun erst ein besonders nervenaufreibender Abschnitt, nämlich die Fehlersuche.

Zunächst ist die völlige Identität des gespeicherten Befehlsvorrats mit den handschriftlichen Unterlagen zu prüfen, damit wir sicher sein können, daß nicht ein unsinniger Tippfehler Schuld an unserem Maleur ist.

Die im Interpreter "festverdrahteten" Fehler-Meldungen können hier nur bedingt als Orientierungshilfe dienen, da

sie auch indirekt verursacht werden können. Wenn wir sicher sind, keinen Eingabefehler gemacht zu haben, nehmen wir uns das abgespeicherte Listing Zeile für Zeile genau unter die Lupe und überlegen zu welchen Ergebnissen die jeweiligen Anweisungen, angeforderten Daten und Rechenoperationen führen. Eine Vielzahl von Fehlern kann auf diese Weise erkannt und berichtigt werden. Wunder wirkt auch bisweilen die Unterstützung eines anderen: wenn man sich mit einer bestimmten Problematik über lange Zeit hinweg intensiv beschäftigt, wird ganz automatisch der Blickwinkel eingengt. Logische Fehlleistungen können dann selbst kaum noch gefunden werden, während ein außen-

stehender Beobachter manchmal auf den ersten Blick das Richtige trifft.

6. Der erste Lauf:

Irgendwann – manchmal dauert es Wochen, ja Monate – ist dann der Augenblick gekommen, in dem das selbst erstellte Programm von Anfang bis Ende läuft. Sofern irgend möglich auch jetzt noch gewisse Vorsicht walten lassen und mit Daten arbeiten, deren "processing" man zur Not mit Papier und Bleistift nachverfolgen kann. Es passiert bisweilen, daß ein Programm anscheinend wunderbar funktioniert, Werte anfordert, verarbeitet und auch Ergebnisse ausspuckt: beim zehnten oder gar erst zwanzigsten Durchlauf stellt man durch Zufall fest, daß die

Resultate teilweise oder auch immer unsinnig sind.

Die Optimierung:

"optimus" ist lateinisch und bedeutet "der Beste". Und weil jeder Programmierer von Natur aus Perfektionist ist, gibt er sich mit der erstbesten Lösung nicht zufrieden sondern versucht, sein Werk zu vervollkommen. Dabei können verschiedene Kriterien eine Rolle spielen, je nachdem, für wen bzw. was unser Programm taugen soll. Solche Optimierungs-Gesichtspunkte können zum Beispiel sein:

- höhere Auflaufgeschwindigkeit
- bessere Bedienungsführung
- geringerer Speicherumfang
- extreme Rechengenauigkeit usw.

Allgemein verbindliche Grund-

sätze gibt es nicht: dem Programmierer (bzw. dessen Auftraggeber) bleibt es überlassen, in welchem Bereich Schwerpunkte zu setzen sind.

Erst dann, wenn ein Programm alle sieben Etappen mit Bravour genommen hat, darf es als rundherum gelungen archiviert werden. Ein Tip: auch bei Ihrer normalen Gebrauchssoftware sollten Sie sich nicht mit Halb- oder Mittelmäßigkeiten zufrieden geben. Das Tüfteln und Ausfeilen sollte erst dann ein Ende haben, wenn Sie sicher sind, alles Ihnen und Ihrem Commodore Mögliche zur Optimierung getan zu haben.

(sk)

programme

VIER gewinnt

```

5 PRINT " "
10 POKE36878,15
15 DIMA(9):DIMB(7,9)
20 POKE36879,25
21 PRINT "J":FORA=1TO5
30 PRINT " "
31 FORB=1TO300:NEXT
32 PRINT "J":FORC=1TO300:NEXTC,A
33 PRINT "J"
34 PRINT " "
60 V=INT(RND(1)*2)+1
65 PRINT " " AN.1 SPIELT MIT "X" AN.2 SPIELT MIT "O"
70 PRINT " " ANSPIELER N. "V" BEGINNT! V=2/V
71 PRINT " " PROGRAMM VON: "
73 PRINT " " FRÄNKE BURKI
75 PRINT " "
77 PRINT " " OLIVIER PROBST
80 PRINT " " TASTE -"
85 GETC$:IFC$=""THEN85
86 PRINT " "
87 PRINT " "
89 PRINT " "
90 GETC$:IFC$<" "THEN90
92 POKE36879,25
94 GOSUB1000
95 V=2/V
96 PRINT " " ANSPIELER N. "V" IST DRAN
98 POKE7766,85+V
100 PRINT " " CHIP IN KOLONNE?
103 ED$=K$:K$=""
105 GOSUB2000
110 K$=LEFT$(K$,1):K=VAL(K$):K=K-1:IFK$=""GOTO800
115 IFK<0ORK>7GOTO800
120 IFA(K)+1>8GOTO800
130 A(K)=A(K)+1
140 B(K,A(K))=V:GOTO170
170 POKE101+K*2-44*A(K),85+V
180 C=A(K):Z=K
190 G=0:IFC=1GOTO205
195 IFG=3GOTO950

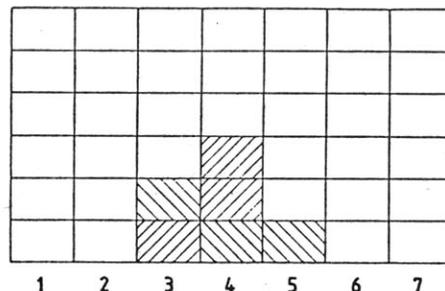
```

für den VC-20 ohne Erweiterung

Das Spielfeld umfaßt 7 Spalten und 6 Reihen, besteht insgesamt also aus 42 Feldern. Die Spalten sind von links nach rechts mit 1 bis 7 numeriert.

Zwei Spieler sind abwechselnd am Zuge. Jeder Spieler besitzt 21 farblich gekennzeichnete Spielsteine (im obigen Schema durch unterschiedliche Schraffierung gekennzeichnet). Ein Zug besteht darin, einen Spielstein von oben in eine Spalte zu werfen, d.h. eine Spalte wird immer von unten aufgefüllt.

Jeder Spieler muß nun versuchen, vier eigene Spielsteine waagrecht, senkrecht oder diagonal in einer Reihe anzuordnen.



Assemblerkurs

Teil 5

In diesem Teil werden wir zeigen wie man den Bildschirm löscht und Programme optimiert. Außerdem zeigen wir Ihnen wie man Programme schreibt, die sich selbst modifizieren.

Um beim Editor eine Hardcopy des Bildschirms zu bekommen, ergänzen Sie bitte den Editor um folgende Zeilen:

```

295 IFA$="T"THENGOSUB11000:GOTO200
READY.

11000 OPEN4,4
11005 FORV=1TO25
11010 QC$=""
11020 FORW=1TO40
11030 QC=PEEK<1024+(V-1)*40+W-1>
11033 IFQC<128THENGOSUB11100:QC$=QC$+CHR$(QC)
11035 IFQC>127THENGOSUB12000
11040 NEXTW
11050 PRINT#4,QC$
11060 NEXTV
11070 CLOSE4
11080 RETURN
11100 REM.SET1 -> ASCII

11131 IFQC<32THENQC=QC+64:GOTO11134
11132 IFQC>63ANDQC<96THENQC=QC+32:GOTO11134
11133 IFQC>95ANDQC<128THENQC=QC+64
11134 RETURN
12000 REM REVERSE
12010 QC#=QC#+CHR$(27)
12020 QC$=QC$+" "
12030 QC=QC-128
12040 GOSUB11100
12045 QC#=QC#+CHR$(QC)
12050 QC$=QC$+CHR$(27)
12060 QC#=QC#+""
12070 RETURN
READY.
    
```

Geben Sie nun, wenn der Computer nach dem Befehl fragt, ein "T" ein, so erhalten Sie auf dem Drucker eine Bildschirmcopy. Geben Sie nun folgendes Programm ein und lassen Sie es assemblieren. Was passiert? Nun das können Sie selbst mit einem Sys 828 testen.

```

1DFILE EQU 1024
2 ORG 828
3 LDA #32
4 LDX #0
5LAB1 STA DFIL,X
6 INX
7 CPX #0
8 BNE LAB1
9LAB2 STA DFIL+256,X
10 INX
11 CPX #0
12 BNE LAB2
13LAB3 STA DFIL+512,X
14 INX
15 CPX #0
16 BNE LAB3
17 LDX #232
18LAB4 STA DFIL+768,X
19 DEX
    
```

```

20 CPX #0
21 BNE LAB4
22 RTS
    
```

Na, was passierte? Richtig der Bildschirm wird gelöscht, aber wie funktioniert das? Wir werden sehen.

Wir fangen mit dem 1. Befehl an
DFILE EQU 1024

Dies ist ein Pseudobefehl für den Assembler. Er soll ein Label anlegen, d.h. jedesmal wenn ich die Adresse DFILE benutze, wird automatisch 1024 dafür eingesetzt. (DFILE ist die Abkürzung für Display File)

Der 2. Befehl
ORG 828

Dies kennen wir schon aus unserem letzten Kurs. Das Programm soll bei Adresse 828 beginnen.

Der 3. Befehl:
LDA #32

Dieser Befehl lädt in den Akku 32.

Der 4. Befehl
LDX #0

In das X-Register wird eine 0 geladen.

Der 5. Befehl
LAB1 STA DFIL,X

Das LAB1 ist ein Label. Dies bedeutet, daß man um bis hier zu springen nur noch LAB1 als Adresse eingeben muß.

Nun zum **STA DFIL,X**

Dies ist ein schwieriger Befehl. Das STA bedeutet Store Akku; na gut aber wohin. Natürlich nach - DFIL,X. Aber was bedeutet das? Ganz einfach: Zu der Adresse DFILE addiert man den Inhalt des X-Registers und schon hat man die Endadresse.

Endadresse = Basicadresse + Inhalt von X-Register (Index)

In unserem Beispiel: Beim 1. Durchlauf steht das X-Register auf 0, (siehe vorherigen Befehl). Die Basicadresse ist DFILE, also 1024. Nach unserer Formel:

Endadresse = 1024 + 0 also 1024

Es wird in die Endadresse 1024 der Inhalt des Akku (32) gebracht. 32 bedeutet ein Space. Die 1. Bildschirmadresse wird gelöscht. Das Ganze nennt man übrigens **Indizierte Adressierung**.

Nun weiter: **INX**

Dies bedeutet des X-Register wird inkrementiert. In Basic würde man folgendes schreiben: X=X+1. Der entsprechende Gegenbefehl heißt übrigens DEX (X=X-1)

Der 7. Befehl

CPX # 0

CPX# Vergleiche der X-Register mit nachfolgendem Wert, nämlich mit 0. Beim 1. Durchlauf steht das X-Register nach dem Inkrementieren auf 1. Der 1. Vergleich endet also mit "ungleich". Dazu gibt es ein Bit das gesetzt oder rückgesetzt wird, je nach Vergleichsausgang. Dieses Bit ist das Zero-Flag. Es ist gesetzt als 1, wenn beide Operanden, also X-Register und der folgende Wert gleich sind. Der nachfolgende Befehl fragt dieses Bit ab.

BNE LAB1

BNE bedeutet Branch on not equal – auf deutsch; Springe wenn ungleich. Nun, es ist ungleich. Springe also zu der Adresse LAB1 zurück. Nun kommt wieder der indizierte Befehl. Dieses Mal steht aber das X-Register nicht auf 0 sondern auf 1. Die Endadresse ist also $1024 + 1 = 1025$. Danach wird das X-Register wieder inkrementiert usw. bis das X-Register auf 255 steht. Bei einem erneuten Inkrementieren steht das X-Register wieder auf 0. 255 dezimal = 1111 1111 binär. Zählt man nun 1 dazu erhält man folgendes

$$\begin{array}{r} 1111\ 1111 \\ + 0000\ 0001 \\ \hline = 10000\ 0000 \end{array}$$

Da das X-Register aber nur 8 Bit besitzt, sitzt darin nur 0000 0000. Die "herausfallende" 1 wird in den sogenannten Carryflag (Übertragflag) übernommen. Auch diese könnte man mit einem BCC (**Branch on Carry Clear** = Verzweigung bei nicht gesetztem Übertragflag) oder mit einem BCS (**Branch on Carry Set** = Verzweigung bei gesetztem Flag) ausnutzen. Das Zero-Flag wird jetzt auch zurückgesetzt, da das X-Register gleich mit 0 ist. Es wird nicht mehr zu LAB1 verzweigt. Vergleich Assembler – Basic bis hierher

Assemblerlisting

```
1DFILE EQU 1024
2   ORG 828
3   LDA #32
4   LDX #0
5LAB1 STA DFIL,X
6   INX
7   CPX #0
8   BNE LAB1
```

Basic-Listing:

```
10 DFIL=1024
20 A=32
30 X=0
40 POKE (DFIL+X),A
50 X=X+1 : IF X=256 THEN X=0
60 IF X<>0 THEN GOTO 40
READY.
```

Die nächsten 4 Programmschritte tun im Prinzip das Gleiche, mit dem Unterschied, daß nicht ab D-File sondern ab D-File + 256-Bytes gespeichert wird. Das ganze nochmal mit D-File + 512. Wir haben nun 768 Bytes mit Blanks aufgefüllt. Der Bildschirm hat aber 1000 Bytes, d.h. wir müssen nur noch 232 Bytes füllen. Dieses Problem lösten wir folgendermaßen ab Zeile 17:

Das X-Register wird mit 232 geladen. Die Basicadresse wieder um 256 auf D-File + 768 erhöht. Aber statt INX benutzen wir den DEX-Befehl und wiederholen die Schleife wie gewohnt.

Danach wie gewohnt der **RTS**-Befehl, um wieder zum Basic-Interpreter zurückzukehren. Zum besseren Verständnis das entsprechende Basicprogramm, startet man es, funktioniert es deutlich wie unser MC-Programm.

```
1 DFIL=1024
3 A=32
4 X=0
5 POKE (DFIL+X),A
6 X=X+1 : IF X=256 THEN X=0
7 IF X<>0 THEN GOTO 5
9 POKE (DFIL+256+X),A
10 X=X+1 : IF X=256 THEN X=0
11 IF X<>0 THEN GOTO 9
13 POKE (DFIL+512+X),A
14 X=X+1 : IF X=256 THEN X=0
15 IF X<>0 THEN GOTO 13
17 X=232
18 POKE (DFIL+768+X),A
19 X=X-1 : IF X=(-1) THEN X=255
20 IF X<>0 THEN GOTO 18
21 END
```

Nun wollen wir daran gehen, unser MC-Programm zu optimieren, d.h. wir wollen versuchen, ob sich das Programm speichermäßig ver-

kleinern läßt.

Dazu folgendes: Der CPX-Befehl setzt, genau wie CMP (Vergleich mit Akku) oder CPY (Vergleich mit Y-Register), die Flags. Eins davon wird beim BNE-Befehl abgefragt. Bei vielen Befehlen werden auch die Flags gesetzt, ohne daß man dafür extra vergleichen muß. Die CPU prüft dann immer auf "0". Dieses passiert

auch bei dem INX und DEX-Befehl.

Wir könnten also einfach hingehen und die CPX # Befehle weglassen. Das Programm sähe dann so aus:

```
1DFILE EQU 1024
2   ORG 828
3   LDA #32
4   LDX #0
5LAB1 STA DFIL,X
6   INX
7   BNE LAB1
8LAB2 STA DFIL+256,X
9   INX
10  BNE LAB2
11LAB3 STA DFIL+512,X
12  INX
13  BNE LAB3
14  LDX #232
15LAB4 STA DFIL+768,X
16  DEX
17  BNE LAB4
18  RTS
```

Außerdem sind die Zeilen 5,6,7 fast gleich mit 8,9,10 und 11,12,13. Könnte man nicht versuchen während des Programmablaufs nach einem Schleifenende die Basisadresse DFIL und 256 zu erhöhen und danach die **gleiche Schleife** noch einmal zu durchlaufen. Dazu muß man wissen, wie der Sta Befehl in Maschinencode aussieht. In unserem indizierten Beispiel 9D Hex +2 Byte Basicadresse

Das 1. Byte der Basicadresse ist das sogenannte Low-Byte, das 2. das Highbyte.

In unserem Falle: Basicadresse ist 1024

= Hex 0400

Hex 04 – Highbyte

Hex 00 – Lowbyte

Da das Lowbyte aber zuerst gebraucht wird sieht der "Befehl 5" STA DFIL,X in Ma-

schinencode so aus:

9D 00 04

Erhöht man nun die 04 um 1 so erhält man Hexadezimal die Adresse, 0500 und das ist dezimal 1280.

Das ist aber auch eine Erhöhung der Basisadresse und 256, also genau das was wir wollen. Da wir aber 3 x die Adresse erhöhen müssen, brauchen wir einen Zähler. Dazu eignet sich das Y-Register. Sehen wir uns das Programm an:

```
1DFILE EQU 1024
2   ORG 828
3   LDA #32
4   LDX #0
5   LDY #3
6LAB1 STA DFIL,X
7   INX
8   BNE LAB1
9   INC LAB1+2
10  DEY
11  BNE LAB1
12  LDX #232
13LAB2 STA DFIL+768,X
14  DEX
15  BNE LAB2
16  RTS
```

Dieses Programm hat allerdings einen Fehler: Ist das Programm einmal gelaufen, so sollte man es kein 2. Mal starten, da sonst die Adresse DFIL nicht mehr 1024 ist. Wir müssen also wieder dafür sorgen, daß der ursprüngliche Zustand wieder hergestellt wird. Dies erreichen wir durch folgende Einfügungen ab Zeile 15.

```
16  LDA #0
17  STA LAB1+1
18  LDA #4
19  STA LAB1+2
```

Diese Einfügungen stellen sicher, daß die Basisadresse nach Ablauf des Programms wieder die ist, die sie zu Anfang war. Sie werden nun vielleicht bemängeln, daß das Programm nun doch durch diese Änderung nicht kürzer geworden ist. Da haben Sie recht. Aber stellen Sie sich vor, die Schleife ist statt 3 Zeilen 20 oder noch mehr Zeilen. Dann rentiert sich diese Art der Programmierung. Man nennt dies übrigens: Eigenmodifikation eines Programms. Das Programm ändert sich also selbst. (hf)

Liebe Compute mit Leser,

an dieser Stelle kommt wieder unser Preisrätsel. Auch diesmal gibt es wieder attraktive Preise zu gewinnen. Wie bereits letzte Woche angekündigt, wollen wir die Form unserer Rätsel ein bißchen variieren. Wir haben uns einige Gedanken gemacht und wollen in dieser Ausgabe ein Denk- und Ratespiel beginnen.

In Anlehnung an eine Fernsehsendung, werden wir Ihnen eine kurze Umschreibung des gesuchten Begriffes geben. Ebenfalls geben wir Ihnen die Anzahl der Buchstaben und eine kleine Hilfe.

Beispiel: Weiches Produkt

_____ T _____

Die Lösung lautet Software (soft für weich, Produkt = ware)

Sie sehen, ganz so ernst wollen wir es natürlich nicht nehmen. Also, dann lassen Sie Ihrer Phantasie und Ihrem Humor freien Lauf, wir wünschen auf jeden Fall viel Spaß und selbstverständlich viel Glück!!

P.S.: Es handelt sich auf jeden Fall um einen Begriff aus der Computerwelt.

Umschreibung des gesuchten Begriffes:

Körperteillektüre

_____ N _____ U _____

Einsendeschluß ist der 13.10.84

Einsendungen bitte unter Angabe des Kennwortes "Suchrätsel Nr. 41/84" an die Adresse des Verlages.

Bitte geben Sie an, für welchen Commodore Sie eine Kasette wünschen.

Der Rechtsweg ist wie immer ausgeschlossen. Nicht teilnahmeberechtigt sind Mitarbeiter des Roeske-Verlages und deren Angehörigen.

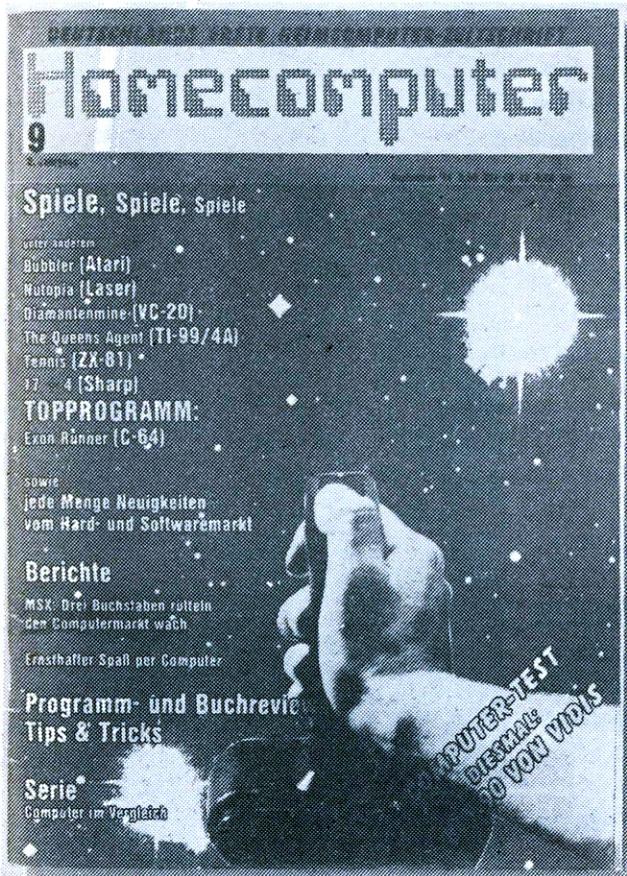
1. Preis
5
Kassetten

2. Preis
3
Kassetten

3. Preis
2
Kassetten

4.-25. Preis
1
Kassette

... und hier noch zwei Zeitschriften aus dem ROESKE Verlag, die Sie unbedingt kennenlernen sollten!



»HOMECOMPUTER«

erscheint am 1. Montag jeden Monats.

»CPU«

erscheint monatlich in der Mitte eines Monats.

In beiden Zeitschriften finden Sie jeden Monat circa 18 tolle Spiel- und Anwenderprogramme, für die bekanntesten Heimcomputer: zum Beispiel den ZX-81, TI-99/4A, Commodore 64, VC-20, Dragon 32, TRS-80, Sharp, Laser, Atari, Apple, Bit 90 und andere.

Außerdem jede Menge Informationen über Neuerscheinungen aus der Soft- und Hardware-Branche, von Messen, Ausstellungen, Computertests, Tips & Tricks.

Sie finden unter unseren Rubriken regelmäßig die Nußknackerseiten, Leserbriefe, Interviews, Clubinfos, das Wörterbuch, HOMECOMPUTER-Bücherkiste, CPU-Bibliothek, den Kassettenservice und den Kleinanzeigenteil.

Haben Sie Fragen zu unseren Heften; rufen Sie uns an:

ROESKE VERLAG

Redaktion

☎ 05651/8558

