

monatlich

# Compute mit

## COMMODORE & SCHNEIDER

5/85

VC-64, VC-20, C-16, CPC-464

2,80 DM  
24 öS  
2,80 sFr

Unabhängiges Magazin für Anwender von Commodore- und Schneider-Computern

**Aktuell +++ Aktuell +++ Aktuell +++ Aktuell**

„COMPUTE MIT...“ haben wir, lieber Leser, speziell auf die Commodore- und Schneider- (hier der CPC 464)-Computer zugeschnitten. Nachdem sich schon im Herbst 1984 auf dem deutschen Computer-Markt abzuzeichnen schien, daß der CPC 464 nach einem nahezu kometenhaften Aufstieg gute Chancen besitzt, dem Giganten Commodore ein ernstzunehmender Konkurrent zu werden, wurde unser Team nahezu gezwungen, diesen Trend aufzunehmen und auch weiterhin zu verfolgen. D. h., für beide populären Homecomputer-Riesen werden wir in jeder Ausgabe von „COMPUTE MIT...“ ein attraktives Angebot bereithalten. Außerdem bietet unsere Redaktion einen *S u p e r*-Service an, um Ihnen eventuelle schwierige Abtippereien zu ersparen. Mehr darüber auf Seite 11!



### Spiel und Spaß mit dem Commodore 64

Ron Jeffries, Glen Fisher & Brian Sawyer

Diese Sammlung von 35 Spielen garantiert Commodore 64-Benutzern viele unterhaltende Stunden. Programmlisten und Erläuterungen der Spiele machen die Eingabe einfach.

ISBN 3-89028-014-5  
182 Seiten, DM 29,80



### Tips & Tricks für CPC-464

ROM-Calls ohne Einsprungsbedingungen  
mehr darüber auf Seite 55

**Report – Werkstatt – Hardware – Bücher –  
Software – Tips & Tricks – Assemblerkurs Teil 1 –  
Jungle – Sequenzer – Olympia – Cobra – Krümi – Q-Bert u. v. mehr**



## Report

Computer Schau '85	3
Schule baut industrielle Abfüllanlage	4
DIDAC, das Diskettenverwaltungssystem für C-64	5
Schneider Personal-Computer „CPC 464“	40

## Werkstatt

Prozedur „Autostart“ für Commodore 64	
Ein Programm, das automatisch in den Speicher Ihres C-64 geladen wird	16

## Hardware

Erweiterung des Systems Commodore 64	25
Neue Standard-Matrixdrucker für Homecomputer: Commodore 64	26
Schneider CPC 464	57
Der Farbbildschirm CTM 640	63

## Bücher

Strategiespiele auf dem C-64	5
Handbuch für Anwenderprogramme	15
VC-20 Anwenderhandbuch	36
Grundkurs in Basic	36
Grundkurs in Pascal	42
Fans des Schneider CPC 464	57

# Software

## Commodore

Senso (C-16)	6
Jungle (C-64)	7
Sequencer (C-64)	20
Sufo (C-64)	27
Olympia (VC-20)	33
Flowers (VC-20)	35

## Schneider CPC 464

Cobra	42
Krümi	46
Haunted Castle	51
Q-Bert	56

## Tips & Tricks

ROM-Calls ohne Einsprungadressen für CPC 464	55
--	----

## Assembler-Kurs 31

## Software-Service 38

## Kleinanzeigen 62

**Die nächste „Compute mit“ erhalten Sie ab 20. Mai bei Ihrem Zeitschriftenhändler.**

## Impressum

„Compute mit“ erscheint monatlich im Tronic-Verlag, 3444 Wehretal 1

### Redaktion:

Axel Grede (verantwortlich)  
Frank Brall, Siegfried Gork, Hartmut Wendt, Holger Grede,  
Ottfried Schmidt

### Freie Mitarbeiter:

Volker Becker, Rolf Freitag

### Gesamtherstellung:

Druckhaus Dienchs Kassel, Frankfurter Str. 168, 3500 Kassel

### Vertrieb:

Inland (Groß-, Einzel- und Bahnhölbuchhandel) sowie Österreich und Schweiz  
Verlagsunion  
Friedrich-Bergius-Straße 20  
6200 Wiesbaden  
Telefon 0 61 21 / 26 50

**Anfragen nicht an Vertrieb oder Druckerei, sondern nur an den Verlag!**

### Anzeigenleitung:

Annelie Kratzenberg, Heike Lux

### Erscheinungsweise:

Erstverkaufstag von „Compute mit“.

### Anzeigenpreise:

Bitte Mediaunterlagen anfordern.

### Anzeigenannahmeschluss:

Jeweils 3 Wochen vor Erscheinungstermin.

### Urheberrecht:

Alle in „Compute mit“ veröffentlichten Beiträge sind urheberrechtlich geschützt. Alle Rechte, auch Übersetzungen, vorbehalten.

Reproduktionen jeder Art (Fotokopien, Microfilm, Erfassung in Datenverarbeitungsanlagen usw.) bedürfen der schriftlichen Genehmigung des Verlages. 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.

### Bezugspreis:

Einzelheft 2,80 DM  
Abonnement Inland 55,- DM im Jahr (12 Ausgaben)  
Ausland: Europa 80,- DM, USA 110,- DM  
**ohne Kassetten!**

### 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 120,- DM 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 Kosten. Zusendungen von Software zur Veröffentlichung sollten 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 und ausführliche Programmbeschreibung (Erklärung der Variablenliste, Beschreibung des Bildschirmaufbaues, Farbe, Grafik usw.). Für eingesandte Programmunterlagen kann keinerlei Haftung übernommen werden.



## COMPUTERSCHAU '85

Die diesjährige Hobby-tronic-Messe in Dortmund vom 20.-24. März stand völlig im Zeichen der ersten Ausstellung von Computern, Computerzubehör und Computersoftware. Die Ausstellungsleitung hatte sich dem allgemeinen Trend des Freizeitmarktes angepaßt und für den immer größer werdenden Kreis der Computerfreunde eine gesonderte Schau in Halle 4 des Messegeländes aufgebaut.

An Ort und Stelle konnten wir uns überzeugen, wie groß das Interesse der Messebesucher an diesen Ständen war. Leider wurden wir, was das Neueste in der Mikroelektronik betrifft, arg enttäuscht. Zwar wurde eine Vielzahl von Produkten vorgestellt und zum Verkauf angeboten, ein absoluter „Messerenner“ jedoch fehlte. So war es dann auch nicht verwunderlich, daß hauptsächlich die Stände mit Software-Angeboten stark umlagert waren. Hier fand der Kunde für alle populären Heimcomputer eine Vielzahl von Spiel- und Anwendungsprogrammen, sodaß einem beinahe schon der Überblick über dieses Riesensortiment verloren ging. Daneben waren besonders stark verschiedene Kaufhäuser vertreten. An diesen

Ständen wurden Fachabteilungen gebildet und jedes Computersystem wurde gesondert vorgestellt und erläutert. Der „Newcomer“ in der Homecomputer-Szene hatte hier die Gelegenheit, sich genauestens über Vor- und Nachteile der einzelnen Systeme zu informieren und konnte Preis-, Leistungsvergleiche anstellen. Die Vielzahl der vertretenen Händlerfirmen entfachten während der Ausstellung einen starken Wettbewerb untereinander. Lachender

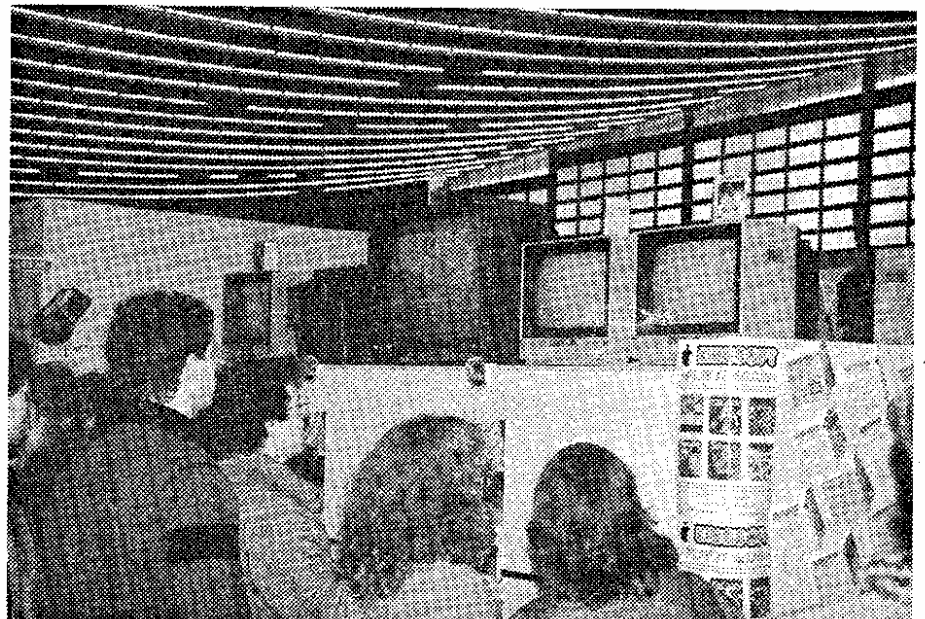
Dritter waren hier die Messebesucher, die zu günstigen Preisen Computerzubehör wie Software etc. erwerben konnten.

Ein anderer häufig vertretener Bereich auf dieser Ausstellung waren die Buch- und Zeitschriftenverlage. Auch hier hatte der Verbraucher die Möglichkeit, sich vor Ort über die Fülle von Fachliteratur zu informieren.

Ein positiver Aspekt dieser Messe waren die zahlreichen Informationsveranstaltungen und der Wettbewerb „Jugend programmiert“. Hier hatte sich die Veranstaltungsleitung Mühe gegeben und versucht, die Jugendlichen verstärkt in den Messebetrieb mit einzubeziehen.

Als äußerst negativ zu bemerken ist, daß die großen Homecomputer-Hersteller wie Commodore, Atari oder Sinclair nicht durch eigene Stände vertreten waren. Wie wir erfahren haben, sind alle prominenten Hersteller erst wieder auf der Hannover-Messe vertreten, und haben wohl deshalb auf die Publikation der neuesten Produkte in Dortmund verzichtet.

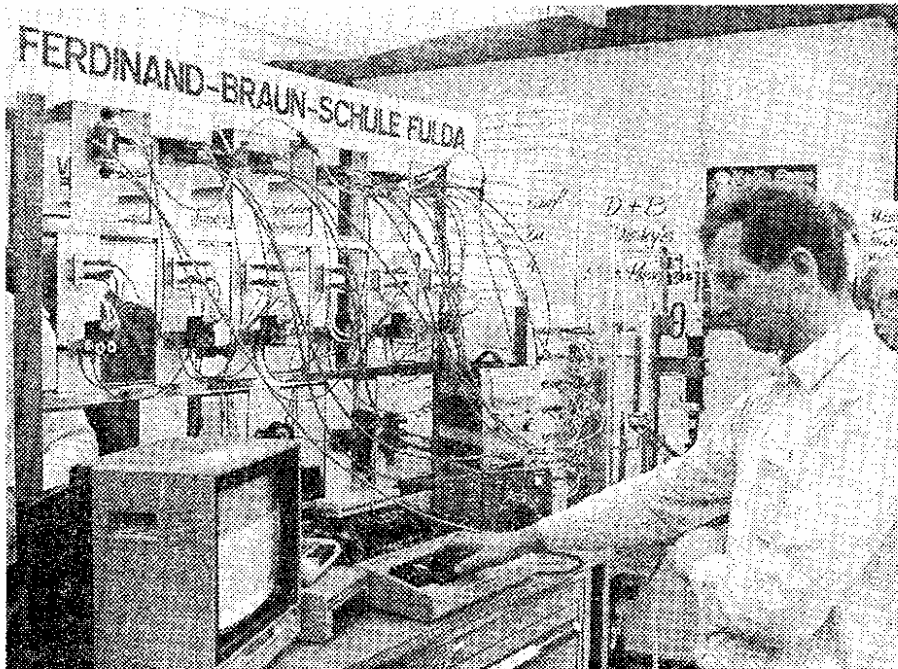
**Fazit:** Wer als Laie diese Veranstaltung aufsuchte, konnte sich einige wichtige Informationen aus der Homecomputer-Branche mitnehmen. Wer billig Computerzubehör erwerben wollte, konnte ebenfalls zufrieden sein. Wir waren es nicht.



Fürs Computerleben lernen:

## Schule baut industrielle Abfüllanlage

Steuerung mit Commodore C 64 / Schneller als übliche SPS



Eine computergesteuerte pneumatische Abfüllanlage hat eine Projektgruppe der Ferdinand-Braun-Schule in Fulda entworfen und gebaut. Der Computer ist ein C 64 von Commodore: Berühmt geworden als Heimcomputer, von dem in Deutschland weit mehr als eine halbe Million Stück verkauft worden sind. Elektronik-Entwickler haben ihn längst auch als ebenso preiswertes wie professionell leistungsfähiges Steuerungssystem entdeckt. Darauf legen die Lehrkräfte der Schule besonderen Wert, denn gerade am Computer sollen die Schüler nicht für die Schule, sondern für das Berufsleben lernen. Die Ferdinand-Braun-Schule ist die

Gewerblich-Technische Berufsschule der Stadt Fulda. Ihr angegliedert sind eine Berufsfach- und Berufsaufbauschule, eine Fachoberschule, ein Berufliches Gymnasium sowie eine Fachschule für Technik bis zur Fachhochschulreife. Die Projektleitung für den Modellversuch liegt in den Hän-

den des Schulleiters, Oberstudiendirektor W. Imhof.

Ziel des Modellversuchs sollte die stufenweise Entwicklung einer komplexen, anwendungsbezogenen Steuerungsanlage sein, die nicht nur die praxisnahe konkrete Anwendung im berufsbildenden Unterricht gestattet, sondern darüber hinaus als Lehrmittel ein Übungssystem der modernen Steuerungstechnik darstellt, das im Schwierigkeitsgrad den unterschiedlichen Erfordernissen der verschiedenen Schulformen und Berufsziele (u. a. Maschinenschlosser, Betriebsschlosser, Maschinenbauer, Elektro-Installateure, Energieanlagen-Elektroniker, Maschinenbautechniker, Techniker der Informations-Elektronik, Steuer- und Regelungstechnik sowie künftige Ingenieurschüler) fachgerecht angepaßt und deshalb gleichermaßen von den Auszubildenden der Berufsschule, den Schülern der Fachoberschule und des Beruflichen Gymnasiums und von Studierenden der Fachschule für

Technik genutzt werden kann. Zu diesem Zweck wurden sechs Laborwagen konstruiert, die alle notwendigen Bauteile für die verschiedenen Schülerübungen bis zum Aufbau der kompletten funktionsfähigen Anlage enthalten. So wird Lernenden die selbständige Lösung realer Steuerungsaufgaben, die Konstruktion und praktische Erprobung, der Nachvollzug und Aufbau im Zusammenhang mit neuformulierten Lehrzielen und der Transfer auf eine Vielzahl von Anwendungen ermöglicht.

Gesteuert wird die sechszylindrige pneumatische Abfüllanlage von einem handelsüblichen Mikrocomputer Commodore C 64, der sich hinsichtlich Preis, Leistung und Bedienungskomfort nach eingehenden Vergleichen und probeweiser praktischer Anwendung als optimales Gerät anbietet. Nicht nur erwies sich der Commodore C 64 als preiswerte und überaus leistungsfähige Alternative zu wesentlich aufwendigeren Speicherprogrammierbaren Steuerungsanlagen (SPS) der Großindustrie, die er in der Geschwindigkeit der Signalverarbeitung überdies um ein Vielfaches übertrifft; er zeigte sich nach Angaben der Projektgruppe anderen Mikrocomputern auch insofern überlegen, als er eine große Speicherkapazität und hohe Computerleistung bei unzähligen Anwendungsmöglichkeiten zu einem erstaunlich günstigen Preis zu bieten vermag. Die Entscheidung für den Commodore C 64 ist vor allem deshalb interessant und überzeugend, weil die Steuerung der Anlage durch Mikrocomputer nicht von vornherein beabsichtigt, der vorliegende Modellversuch also keineswegs darauf angelegt war. Die Steuerung wurde ursprünglich zu Lehrzwecken mit austauschbaren rein pneumatischen Steuerelementen ver-

schiedener Lieferfirmen sowie in Ansätzen mit Speicherprogrammierbaren Steuerungen (SPS) vorgenommen. Erst der technische Fortschritt und die jüngste Entwicklung auf dem Computermarkt führten dazu, die Anlage vor rund einem Jahr mit einem Mikrocomputer auszustatten. Leicht zu bedienen, mit seiner ausgereiften

Technik und die Vielzahl möglicher eigenständiger Programme ist der Commodore C 64 geradezu prädestiniert als Lehr- und Unterrichtsinstrument. Der Preis für den Commodore C 64 liegt unter 700 Mark. Die Gesamtkosten der betriebsfertigen Computerausstattung mit Monitor, Drucker und Schnittstellen beliefen

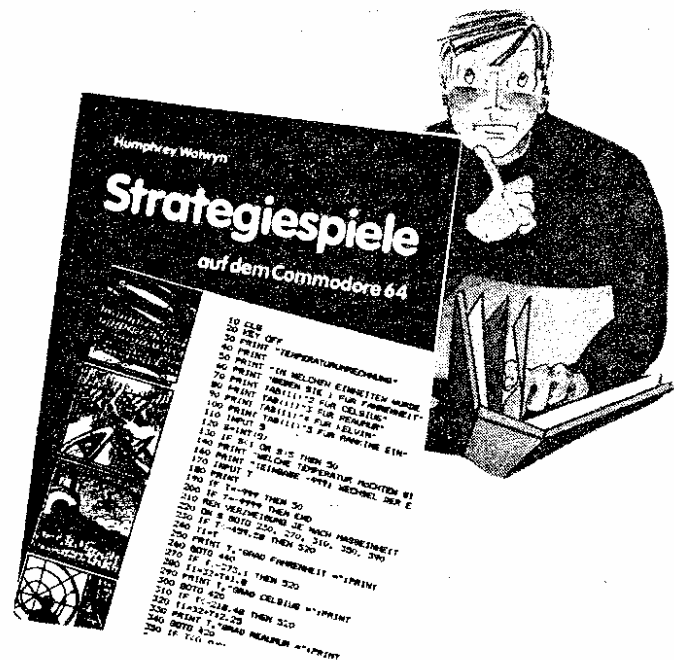
sich auf ca. 2500 Mark. Geliefert wurde der Mikrocomputer samt Zubehör von der Firma Weinrich, Büro-Organisation, in Fulda, die sich als wohlsortierter und fachkundiger Schulausstatter für den osthessischen Raum besonders für den Bereich der Elektronischen Datenverarbeitung anbietet.

## DIDAC, das Diskettenverwaltungssystem für den C 64

Auf der MICRO COMPUTER MESSE '85 in Frankfurt stellt die Firma CPL Computer plus Soft GmbH ihr Diskettenverwaltungssystem DIDAC vor. Mit DIDAC kann endlich Ordnung in die Diskettensammlung gebracht werden. Bis zu 4500 Programme und Dateien verwaltet das System pro Datendiskette. Alle Angaben zu den erfaßten Programmen werden aus dem Inhaltsverzeichnis der Diskette gelesen. Jedes Programm kann in eine von 26 möglichen Kategorien eingeordnet und mit zusätzlichen Bemerkungen versehen werden. Die kategorisierten Programme können nach den verschiedensten Kriterien sortiert und in zwei Druckformaten auf den Drucker ausgegeben werden, wobei Drucker mit seriellem sowie CENTRONICS Interface unterstützt werden. Das Suchen von Programmen bereitet keine Schwierigkeiten, dafür stehen verschiedene Suchkriterien zur Verfügung. DIDAC ist zur Zeit in deutscher Sprache erhältlich und kostet für den Endverbraucher DM 89,-. Englische, französische und italienische Versionen sind in Vorbereitung.

## Strategiespiele auf dem Commodore 64

Humphrey Walwyn



An diesen Strategiespielen werden vor allem Geschichts-Interessierte ihren Spaß haben: 6 historische Schlachten werden simuliert, darunter Waterloo und der Amerikanische Bürgerkrieg. Der Anwender kann die Strategie-Entscheidungen selbst nachvollziehen und Alternativen ausprobieren. Spielregeln und der histo-

rische Hintergrund jedes Spiels werden eingehend erläutert. Auch die Anpassung der Programme an einen PET-Computer wird erklärt.

ISBN 3-89028-021-8  
Dezember 1984  
ca. 256 Seiten, DM 29,80

# Senso

Das Programm „SENSO“ wurde auf dem Commodore 16 ohne Speichererweiterung geschrieben und getestet

Bei diesem Spiel müssen Sie versuchen, die Signale, die der Computer optisch und akustisch ausgibt, korrekt nachzuspielen

Dabei haben Sie die Möglichkeit, zwischen verschiedenen Schwierigkeitsstufen zu wählen. Die Eingabe der Töne erfolgt per Tastendruck, was in der Hi-Res-Graphik des Pro-

gramms deutlich zu sehen ist. Aber Vorsicht: Warten Sie mit Ihren Eingaben nicht zu lange, denn auch die Zeit spielt in diesem Programm eine Rolle. Es kommt also auf Schnellig-

keit und ein gutes Gehör an. Hier wird, anders als bei den meisten Spielprogrammen, Ihre Reaktion und Intelligenz auf den Prüfstein gestellt.

```

20 DIMB(40),F(40)
25 DA=1:VOL8
30 GRAPHIC0,1:COLOR0,5,4:COLOR1,1:COLOR4,5,2
35 PRINTTAB(83)"
40 PRINTTAB(3)"1SEN50"
45 PRINTTAB(3)"1SEN50"
50 PRINTTAB(3)"1SEN50"
55 PRINTTAB(3)"1SEN50"
60 PRINTTAB(3)"1SEN50"
65 COLOR1,15,2:PRINTTAB(87)"(C) 1985 BY ANDREAS BACHLER"
70 COLOR1,3,2:PRINTTAB(88)" S ■ = SPIELBEGINN"
75 PRINTTAB(46)" E ■ = SPIELEND"
80 PRINTTAB(46)" B ■ = SCHWIERIGKEITSGRAD ( 1 )"
85 GETA$:IFA$="S"THEN105
90 IFA$="E"THENGGRAPHIC0,1:END
95 IFA$="B"THENGOSUB210
100 GOTO85
105 GOSUB245
110 A=0
115 A=A+1:X=A:IFA>DA*8THEN195
120 B(A)=INT(RND(1)*4+1)
125 FURC=1TUA
130 ONB(C)GOSUB225,230,235,240
135 NEXT
140 A=0
145 IFA=XTHENFORI=1TO500:NEXT:GOTO115
150 A=A+1:T1$="000000"
155 GETA$
160 IFT1$="000010"THEN205
165 IFA$="T"THENF(A)=1:GOSUB225:GOTO190
170 IFA$="U"THENF(A)=2:GOSUB230:GOTO190
175 IFA$="G"THENF(A)=3:GOSUB235:GOTO190
180 IFA$="J"THENF(A)=4:GOSUB240:GOTO190
185 GOTO155
190 IFF(A)<>B(A)THEN205:ELSE145
195 FORI=1TO6: SOUND1,834,5: SOUND1,810,5
200 NEXT:FORI=1TO1000:NEXT:GOTO25
205 SOUND1,0,70:FORI=1TO1000:NEXT:GOTO25
    
```

Variablendefinitionen:

- A Laufvariable für Tonanzahl
- C,I,K Schleifenvariablen
- X Anzahl der gespielten Töne
- DA Schwierigkeitsgrad
- B(x) Tonausgabe des Computers
- F(x) String für Get-Abfrage
- A\$ Kontrolliert Eingabezeit

```

210 DA=DA+1:IFDA>5THENDA=1
215 PRINT"@";FORI=1TO16:PRINT"@";:NEXT
220 PRINTTAB(32);DA:RETURN
225 SOUND1,770,10:COLDR4,8,6:FORK=1TO500:NEXT:RETURN
230 SOUND1,810,10:COLDR4,7,3:FORK=1TO500:NEXT:RETURN
235 SOUND1,854,10:COLDR4,3,3:FORK=1TO500:NEXT:RETURN
240 SOUND1,881,10:COLDR4,6,4:FORK=1TO500:NEXT:RETURN
245 COLDR0,3,6:COLOR1,1:GRAPHIC1,1
250 CIRCLE1,160,100,75:CIRCLE1,160,100,74
255 CIRCLE1,135,75,15:CIRCLE1,185,75,15
260 CIRCLE1,135,125,15:CIRCLE1,185,125,15:CIRCLE1,160,100,65
265 COLOR1,8,6:PAINT1,135,75
270 COLOR1,7,3:PAINT1,185,75
275 COLOR1,3,3:PAINT1,135,125
280 COLOR1,6,4:PAINT1,185,125
285 COLOR1,1:CHAR1,16,9,"T":CHAR1,22,15,"J"
290 CHAR1,22,9,"U":CHAR1,16,15,"G"
295 CHAR1,17,12,"SENSO"
300 RETURN
    
```

# JUNGLE

für  
Commodore 64

## Laden und Starten

Tippen Sie Jungle 1 ab, SAVEN Sie es und starten Sie anschließend. Dann tippen Sie Jungle 2 ab und SAVEN Sie es auf Ihrer Cassette hinter Jungle 1. Zum späteren LADEN der beiden Programmteile laden Sie nur Jungle 1 und starten Sie. Der 2. Programmteil wird dann automatisch nachgeladen und gestartet.

Wie schon der Name dieses Spieles verrät, hat es Sie in den Urwald verschlagen. Sie sammeln im Urwald verborgene Schätze ein. Dazu müssen Sie an Lianen Ihre Kletterkünste unter Beweis stellen und im Sprung verschiedene Ebenen erreichen. Das Ganze klingt aber etwas einfacher, als

es in Wirklichkeit ist. An den Lianen klettern Feuerballen hoch und runter und behindern Sie beim Klettern. Auf den Plattformen wandern pfannenförmige Gebilde hin und her und von oben fallen ständig große Holzfässer auf Sie nieder. Hinzu kommt noch, daß der Bonus-

zähler unbarmherzig rückwärts zählt. Haben Sie aber eine Runde geschafft, bekommen Sie den Bonuszähler zu den von Ihnen erreichten Punkten hinzugezählt. Geschwindigkeit, Anfangsbild und Anzahl der Leben können vor dem Spiel gewählt werden. Die Steuerung erfolgt über







538 DATA7,169,8,141,5,192,173,5,192,201,5,144,6,206,1,208,206,1,208,32,94  
 539 DATA19,206,5,192,208,235,169,0,141,32,192,238,2,192,206,3,192,32,117  
 540 DATA19,169,8,141,5,192,32,94,19,206,5,192,208,248,238,2,192,32,117,19  
 541 DATA169,8,141,5,192,238,1,208,32,94,19,206,5,192,208,245,169,0,141,32  
 542 DATA192,238,3,192,238,2,192,173,2,192,201,39,144,3,76,113,21,32,117,19  
 543 DATA32,139,20,96,238,0,208,173,0,208,208,8,173,16,208,9,1,141,16,208  
 544 DATA32,150,16,96,0,0,0,173,2,192,201,39,208,3,104,104,96,173,2,192,201  
 545 DATA255,208,3,104,104,96,173,2,192,141,0,192,173,3,192,141,1,192,206  
 546 DATA1,192,32,164,16,240,3,76,188,19,238,1,192,32,164,16,240,7,201,32  
 547 DATA240,3,76,191,19,169,134,141,248,7,104,104,169,0,141,32,192,96,238  
 548 DATA1,192,238,1,192,32,164,16,240,25,201,32,240,21,76,225,19,173,4,192  
 549 DATA41,128,240,6,173,4,192,141,248,7,32,149,20,104,104,96,206,0,192,32  
 550 DATA164,16,240,7,201,32,240,3,76,206,19,96,0,0,0,0,173,2,192,201,1  
 551 DATA208,1,96,32,239,21,169,131,141,4,192,169,132,141,248,7,169,8,141  
 552 DATA5,192,173,5,192,201,5,144,6,206,1,208,206,1,208,32,114,20,206,5,192  
 553 DATA208,235,169,0,141,32,192,206,2,192,206,3,192,32,117,19,169,8,141  
 554 DATA5,192,32,114,20,206,5,192,208,248,206,2,192,32,117,19,169,8,141,5  
 555 DATA192,238,1,208,32,114,20,206,5,192,208,245,169,0,141,32,192,238,3  
 556 DATA192,206,2,192,173,2,192,201,255,144,3,76,113,21,32,117,19,96,0,0  
 557 DATA0,206,0,208,173,0,208,201,255,208,8,173,16,208,41,254,141,16,208  
 558 DATA32,150,16,96,0,0,0,173,3,192,201,23,144,3,76,113,21,173,2,192,141  
 559 DATA0,192,172,3,192,200,140,1,192,169,1,141,6,192,32,164,16,240,21,201  
 560 DATA32,240,17,206,0,192,32,164,16,240,9,201,32,240,5,169,0,141,6,192  
 561 DATA234,234,234,173,2,192,141,0,192,172,3,192,200,140,1,192,32,164,16  
 562 DATA240,17,201,32,240,13,206,0,192,32,164,16,240,5,201,32,240,1,96,206  
 563 DATA1,192,206,1,192,173,2,192,141,0,192,32,164,16,240,3,76,14,21,238  
 564 DATA1,192,32,164,16,240,7,201,32,240,3,76,14,21,96,0,0,0,169,2,141,7  
 565 DATA192,169,8,141,5,192,173,4,192,41,128,240,23,173,4,192,201,128,208  
 566 DATA3,32,94,19,173,4,192,201,131,208,3,32,114,20,76,57,21,32,150,16,238  
 567 DATA1,208,206,5,192,208,215,238,3,192,173,4,192,41,128,240,27,173,4,192  
 568 DATA201,128,208,3,238,2,192,201,131,208,3,206,2,192,173,1,192,201,24  
 569 DATA208,3,76,113,21,32,137,19,206,7,192,208,165,234,234,234,173,4,192  
 570 DATA41,128,208,9,238,1,208,32,150,16,76,155,21,173,4,192,201,128,208  
 571 DATA6,238,1,208,32,94,19,173,4,192,201,131,208,6,238,1,208,32,114,20  
 572 DATA173,16,208,41,1,208,5,173,0,208,240,10,173,1,208,201,255,240,3,76  
 573 DATA113,21,104,104,104,104,96,0,0,0,169,0,141,4,212,141,6,212,169,6,141  
 574 DATA5,212,238,8,192,173,8,192,201,3,208,5,169,0,141,6,192,201,2,240,8  
 575 DATA169,21,141,1,212,76,230,21,169,24,141,1,212,169,17,141,4,212,96,0  
 576 DATA0,0,169,0,141,4,212,141,6,212,169,10,141,5,212,169,30,141,1,212,169  
 577 DATA17,141,4,212,96,0,0,0,173,16,208,41,1,208,19,173,1,208,201,64,208  
 578 DATA12,173,0,208,9,238,1,208,5,169,1,141,51,192,173,33,192,240,7,201,128  
 579 DATA240,3,76,53,22,169,128,141,47,192,238,33,192,173,33,192,41,2,208  
 580 DATA3,76,131,23,162,6,238,16,192,189,1,208,24,125,13,192,157,1,208,173  
 581 DATA16,192,201,8,208,45,138,74,168,169,17,56,249,248,7,153,249,7,169  
 582 DATA7,192,141,20,192,189,8,192,24,125,13,192,141,21,192,157,8,192,32  
 583 DATA153,22,240,9,169,0,56,253,13,192,157,13,192,202,202,208,190,173,16  
 584 DATA192,201,8,208,5,169,0,141,16,192,76,224,22,0,169,40,141,0,193,169  
 585 DATA0,133,141,169,4,133,142,172,21,192,165,141,24,109,0,193,133,141,144  
 586 DATA2,230,142,136,208,241,172,20,192,177,141,201,160,240,11,201,83,240  
 587 DATA7,201,65,240,3,201,116,96,169,32,76,201,22,0,173,30,208,41,1,240  
 588 DATA3,76,113,21,76,139,20,0,173,24,192,208,20,173,27,212,201,252,144  
 589 DATA10,169,1,141,25,192,169,1,141,24,192,76,111,23,234,234,173,27,212  
 590 DATA201,251,144,9,169,0,56,237,25,192,141,25,192,238,9,208,173,25,192  
 591 DATA201,1,208,19,238,8,208,238,8,208,208,32,173,16,208,9,16,141,16,208  
 592 DATA76,61,23,206,8,208,206,8,208,173,8,208,201,254,208,8,173,16,208,41  
 593 DATA239,141,16,208,173,9,208,201,248,240,15,173,16,208,41,16,208,5,173  
 594 DATA8,208,240,3,76,111,23,169,0,141,8,208,169,64,141,9,208,169,0,141  
 595 DATA24,192,173,16,208,41,239,141,16,208,76,111,23,0,0,173,0,220,41,16  
 596 DATA208,13,169,255,141,32,192,169,2,141,40,192,234,234,234,162,4,189  
 597 DATA8,208,24,125,33,192,157,8,208,221,34,192,240,8,221,37,192,240,3,76  
 598 DATA165,23,169,0,56,253,33,192,157,33,192,202,202,208,220,234,234,234  
 599 DATA173,42,192,208,3,76,49,234,169,0,141,11,212,169,33,141,11,212,169  
 600 DATA0,141,42,192,169,32,172,43,192,136,145,148,200,145,148,200,145,148  
 601 DATA165,148,56,233,40,133,148,176,2,198,149,169,32,145,148,136,145,148  
 602 DATA136,145,148,169,250,141,50,192,76,49,234,0,0,0,173,47,192,240,89  
 603 DATA173,45,192,56,233,100,141,45,192,176,3,206,46,192,169,28,133,211  
 604 DATA169,0,133,214,32,108,229,169,5,141,134,2,173,46,192,174,45,192,32  
 605 DATA205,189,169,32,32,210,255,169,32,32,210,255,169,0,141,47,192,173

# programme

```
606 DATA45,192,208,8,173,46,192,208,3,32,113,21,173,46,192,201,4,176,15,169
607 DATA0,141,11,212,169,16,141,8,212,169,17,141,11,212,234,173,50,192,240
608 DATA45,169,6,133,211,169,0,133,214,32,108,229,169,5,141,134,2,173,48
609 DATA192,24,109,50,192,141,48,192,144,3,238,49,192,173,49,192,174,48,192
610 DATA32,205,189,169,0,141,50,192,173,51,192,240,3,104,104,96,96,169,127
611 DATA141,14,220,169,51,133,1,169,0,133,139,133,141,169,8,133,140,169,208
612 DATA133,142,160,0,177,141,145,139,200,208,249,230,140,230,142,165,142
613 DATA201,216,208,239,169,55,133,1,169,129,141,14,220,96,0,0,0,169,0,133
614 DATA139,133,141,169,16,133,140,169,96,133,142,160,0,177,141,145,139,200
615 DATA208,249,230,140,230,142,165,142,201,111,208,239,96,120,169,49,141
616 DATA20,3,169,234,141,21,3,88,96
10000 FORA=0TO2000:NEXT:PRINTPEEK(631);PEEK(632);PEEK(633)
```

## Teil 2

```
1 REM *****
2 REM * JUNGLE (C)'85 *
3 REM * BY *
4 REM * THOMAS GUESMANN *
5 REM *****
10 V=53248:SI=54272:CS=58732:SC=0:DIMFO(3,1,32):PRINT"J"
20 POKEV+21,0:POKEV+32,0:POKEV+33,0
25 POKESI+15,50:POKESI+19,0:POKESI+20,240:POKESI+18,129:POKESI+24,143
30 POKESI+8,16:POKESI+12,9:POKESI+13,0:POKE49200,0:POKE49201,0
50 FORA=0TO3:FORB=V+2TOV+16:READPO(A,0,B-V-2):NEXT
55 FORB=49161TO49193:READPO(A,1,B-49161):NEXT:NEXT
60 FORA=8192TO8192+64*12-1:READB:POKEA,B:NEXT
185 REM
190 REM ZEICHEN DEFINIEREN
195 REM
200 SYS6283
250 FORA=0TO17:READADR:ADR=2048+8*ADR
255 FORB=0TO7:READW:POKEADR+B,W:NEXT:NEXT
265 REM
270 REM SPRITEPOSITIONEN SETZEN
275 REM
280 POKE2040,128:POKE2041,137:POKE2042,136:POKE2043,137:POKE2044,138
285 POKE2045,139:POKE2046,139:POKE181,0:POKE247,0
290 POKEV+37,2:POKEV+38,7:POKEV+39,6:POKEV+40,8:POKEV+41,8:POKEV+42,8
292 POKEV+43,9:POKEV+44,9
295 GOSUB10000:POKEV+22,216
300 POKEV+17,11:FORA=49152TO49198:POKEA,0:NEXT
320 POKEV,36:POKEV+1,216:POKE49154,3:POKE49155,22
330 FORA=V+2TOV+16:POKEA,PO(RU,0,A-V-2):NEXT
335 FORA=49161TO49193:POKEA,PO(RU,1,A-49161):NEXT
340 POKE49197,136:POKE49198,19:POKE49203,0:POKEV+28,111
350 POKEV+30,0:POKEV+34,14:POKEV+35,7:POKEV+24,19:ONRU+1GOTO400,600,800,1000
385 REM
390 REM 1. SPIELFELD
395 REM LIANEN=COMMODORE+H
397 REM
400 PRINT"J"
405 PRINT" "
410 PRINT" "
415 PRINT" "
420 PRINT" "
425 PRINT" "
430 PRINT" "
435 PRINT" "
440 PRINT" "
445 PRINT" "
450 PRINT" "
452 PRINT" "
455 PRINT" "
460 PRINT" "
462 PRINT" "
```



# Super-Service

Lieber Leser,

das Redaktionsteam hat mit außergewöhnlichem Engagement unsere Marktposition nicht nur festigen, sondern um ein weiteres Magazin erweitern können. Nach „COMPUTRONIC“, „HOMECOMPUTER“ und „CPU“, erscheint ab diesem Monat „COMPUTE MIT...“ als viertes Magazin im Tronic-Verlag und ist ab sofort wieder bei Ihrem Zeitschriften-Händler erhältlich.

Neben einem erfolgreichen Start wünschen wir uns natürlich dem Markt gerecht zu werden. Dazu gehören nach unserer Meinung gute Programme und eine Vielzahl von Informationen rund um Ihren Computer. Zusätzlich wollen wir in „COMPUTE MIT...“ unseren schnellen, von vielen Kunden schon geschätzten Service auch in diesem Magazin fortführen und weiter ausbauen.

„COMPUTE MIT...“ haben wir, lieber Leser, speziell auf die Commodore- und Schneider- (hier der CPC 464)-Computer zugeschnitten. Nachdem sich schon im Herbst 1984 auf dem deutschen Computer-Markt abzuzeichnen schien, daß der CPC 464 nach einem nahezu kometenhaften Aufstieg gute Chancen besitzt, dem Giganten Commodore ein ernstzunehmender Konkurrent zu werden, wurde unser Team nahezu gezwungen, diesen Trend aufzunehmen und auch weiterhin zu verfolgen. D. h., für beide populären Homecomputer-Riesen werden wir in jeder Ausgabe von „COMPUTE MIT...“ ein attraktives Angebot bereithalten. Außerdem bietet unsere Redaktion einen **S u p e r - S e r v i c e** an, um Ihnen eventuelle schwierige Abtippereien zu ersparen. Auf dieser Seite finden Sie ein sogenanntes Kombi-ABO. Es beinhaltet ein Abonnement + bespielter Kassette, auf der alle in der jeweiligen Ausgabe veröffentlichten Programme sind. Wir bieten Ihnen damit die Möglichkeit, 12 Magazine + 12 Kassetten jeden Monat mit aktuellen und attraktiven Programmen zu beziehen.

Allen Commodore- und Schneider-Computer-Besitzern wünschen wir für die Zukunft viel Spaß mit „COMPUTE MIT...“ und ein Error-freies Eingeben. – nach der richtigen Eingabe der Daten laufen alle Programme. Dafür verbürgt sich unsere Redaktion – **TRONIC-Verlag**

\*\*\*\*\*

## So ein Kombi-Abo ist nicht nur für scharfe Rechner eine lohnende Sache. Wir liefern Ihnen Ihre „Compute mit ...“ plus kompletten Datenträger (Kassette) frei Haus!

Der Datenträger enthält alle Programme (Commodore und Schneider) der jeweiligen Ausgabe.

**Super-Service**

Name / Vorname .....

**Kombi-Abo**

„Compute mit ...“

Straße / Nr. .... PLZ / Ort .....

Ich wünsche folgendes ABO (Inland):

**Bitte ausschneiden und senden an:**

**Tronic-Verlag GmbH**  
**Postfach 41**  
**3444 Wehretal 1**

12 Exemplare „Compute mit ...“ + 12 bespielte Kassetten  
zum Preis von 97,50 DM (Ausland 110,- DM)

12 Exemplare „Compute mit ...“ ohne bespielte Kassetten  
zum Preis von 30,- DM (Ausland 40,- DM)

gegen

Rechnung       Vorkasse

.....  
Datum, Unterschrift

Abonnement-Kündigungen:  
6 Wochen vor Ablauf des Jahresabonnements. Außerdem kann ich diese Bestellung innerhalb von 10 Tagen beim Verlag widerrufen! Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs.

.....  
Datum, Unterschrift













# Prozedur „Autostart“ in Theorie und Praxis

Unter „Autostart“ versteht man ein Programm, das bei Einschalten des Computers in den Speicher geladen wird und ohne jeden Eingriff seitens des Anwenders automatisch startet

## Commodore 64

Ein typisches Beispiel für „Autostart“ sind die Module, die man in den Expansion-Port einsteckt. Diese sind meistens beim Einschalten des Computers automatisch aktiviert. Die ersten Probleme sind da, wenn das Programm auf Diskette oder Kassette gespeichert ist.

Also, was tun ...?  
Wir wollen zuerst untersuchen, was für Möglichkeiten uns, beim C-64, zur Verfügung stehen.  
Das erste Verfahren, das eigentlich bekannt ist, wäre die Betätigung der Tasten Shift + Run/Stop. Damit würde das erste Programm, das auf

Band gespeichert ist, geladen und gestartet. Aber diese Lösung ist an den Gebrauch der Kassettenrecorder begrenzt.  
Das zweite Verfahren wäre, den Stack zu manipulieren, und zwar folgendermaßen: Der Stack liegt beim C-64 im Bereich \$0100 bis \$01FF, er

### Assembler-listing Nr. 1

```

,02C8 A9 00 LDA #00
,02CA 85 9D STA 9D ; LOAD Routine
,02CC 20 D5 FF JSR FFD5
,02CF 86 2D STX 2D ; speichert Ende des Basic Prg.
,02D1 84 2E STY 2E
,02D3 A9 83 LDA #83
,02D5 8D 02 03 STA 0302 ; Original Warm-Start Vektor,
,02D8 A9 A4 LDA #A4 ; wird zurückgeschrieben
,02DA 8D 03 03 STA 0303
,02DD A2 06 LDX #06
,02DF 86 C6 STX C6 ; kopiert die Befehle
,02E1 BD EC 02 LDA 02EC,X ; Load + Return
,02E4 9D 76 02 STA 0276,X ; Run + Return
,02E7 CA DEX ; in die Tastaturpuffer
,02E8 D0 F7 BNE 02E1
,02EA 6C 02 03 JMP (0302) ; indirekter Sprung zum Warm-Start
,02ED 4C CF 0D JMP 0DCF ; L+(Shift) O + Return
,02F0 52 ★★★
,02F1 D5 0D CMP 0D,X ; R+(Shift) U + Return
,02F3 20 20 20 JSR 2020
,02F6 20 20 20 JSR 2020 ; Platz für den Namen des Programmes
,02F9 20 20 20 JSR 2020
,02FC 20 20 20 JSR 2020
,02FF 20 8B E3 JSR E38B ; Vektor Routine Fehlermeldung
,0302 C8 INY
,0303 02 ★★★ ; Vektor Routine Warm-Start

```

wird zum Ablegen von Rücksprungadressen benutzt. Nach Beendigung einer Routine (z. B. LOAD) holt sich der Prozessor vom Stack zwei Bytes und springt die Adresse an, die sich aus diesen beiden Bytes ergibt.

Also müßte man diese Adresse so ändern, daß der Prozessor nicht an die eigentliche Rücksprungstelle springt, sondern auf eine eigene Routine. Nun, wir wissen nicht genau, wo die Adresse geholt wird. Da bleibt uns nur noch die Möglichkeit, den ganzen Stack mit der von uns gewünschten Rücksprungadresse zu belegen. Dann wird das Maschinenprogramm, auf das unsere Adresse zeigt, direkt an den Stack angehängt. Von diesem Startprogramm wird dann das Hauptprogramm angesprungen.

Der Nachteil bei diesem Verfahren ist, daß man den Speicherbereich von \$0100 bis \$07FF und zusätzlich das Hauptprogramm mitabspeichern muß, und das kostet sehr viel Zeit. Aber was wir machen werden, ist folgendes: Vorerst ein Blick auf die Speicherbelegung des C-64

\$02C8-\$02FF	nicht benutzt
\$0300-\$0301	Vektor Routine Fehlermeldung
\$0302-\$0303	Vektor Routine Warm-Start

Wie man daraus ersehen kann, sind die Speicherplätze von \$02C8 bis \$02FF frei, und das ermöglicht uns, an diese Stelle die Speicherung unserer kleinen Maschinen-Routine (siehe Listing 1) zu legen. Was uns aber noch interessiert, sind die Speicherstellen \$0302, \$0303, die den Vektor für die Warm-Start-Routine enthalten.

Für die, die es nicht wissen sollten, sei gesagt, daß jedesmal, wenn der Computer sich mit READY meldet, er einen Sprung zur Warm-Start gemacht hat. Das heißt, diese Routine wird angesprungen, jedesmal, wenn der Computer eine Operation be-

endet hat, z. B. bei dem Befehl END, oder, und das ist für uns wichtig, nach Beendigung des Load-Vorgangs.

Was wir jetzt tun müssen, ist, den Vektor Warm-Start so zu verändern, daß er auf unsere Maschinen-Routine zeigt, die ab Adresse \$02C8 anfängt. Dies erreicht man, indem man den veränderten Vektor mitabspeichert, wie beim Assembler-Listing 1 zu sehen ist. Zuvor werden die Befehle Load und Run in die Tastaturpuffer kopiert.

Mit dieser kleinen Routine kann man schon, nach dem Load, jedes Basic-Programm automatisch starten. Vorausgesetzt, es liegt ein Programm im Speicher.

Aber das ist nicht unser Ziel.

Wir wollen beide Programme in eines zusammenfassen, deswegen wird Listing 2 gebraucht.

Mit dem ersten Teil wird die Autostart-Routine von \$02C8 bis \$0303 mit der Sekundäradresse 3 (siehe Zeichnung) abgespeichert. Nachdem dieser Vorgang beendet ist, wird durch den noch mal veränderten Vektor der zweite Teil angesprungen, der das Hauptprogramm mit der Sekundäradresse 1 abspeichert.

Anschließend werden die Speicherstellen \$0302, \$0303 in den ursprünglichen Zustand versetzt.

## Zeichnung

03	:	C8	:	02	:	04	:	03	:	20	:	20
\$	\$	\$	\$	\$	\$	\$	\$	\$	\$	\$	\$	\$
0	0	0	0	0	0	0	0	0	0	0	0	0
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	4	4	4	4	4	4	4	F
C	D	E	F	0	1	1	1	1	1	1	1	F

Zusammensetzung der Speicherstelle im Kassettenpuffer nach dem Laden der Autostart-Routine.

Der erste Wert (in \$033C) schließt auf den File-Typ: 01 => Programm; 02 => Sequentiell; 03 => nicht Relokatibel.

Der Zweite und Dritte sind die Anfangsadressen in der Reihenfolge LSB, MSB.

Der Vierte und Fünfte die Endadresse des Programmes.

Ab der Sechsten steht der Filename in ASCII-Code.

## Schutz von Programmen

Sicherlich, die Möglichkeit, ein Programm automatisch zu starten, ist schon an sich eine gute Schutzmaßnahme.

Man kann aber zusätzlich noch vor den JMP(0302), siehe Listing 1, die wichtigen Vektoren ändern.

Diese wären z. B.:

\$0306, \$0307	Vektor List
\$0316, \$0317	Vektor Brk
\$0318, \$0319	Vektor Nmi-Restore
\$0328, \$0329	Vektor Stop
\$0332, \$0333	Vektor Save

## Programmbeschreibung

Das komplette Programm liegt als Basic-Loader vor, wird mit '←A' aktiviert und nach den neuen Namen gefragt. Hat man es eingegeben, so kann es neu abgespeichert werden. Die Länge des Namens darf jetzt höchstens 13 Zeichen enthalten, falls man mehr eingegeben hat, bleiben diese unberücksichtigt.

Das Programm belegt den Bereich ab \$9000, wobei noch zu sagen ist, daß der Basic-Ende-Zeiger auf diese Adresse zeigt und dadurch vor Überschreibung geschützt ist.



## AUTOSTART

### Assembler-listing Nr. 2

```

,90CC A9 01 LDA #01 ; Log. file Nr.
,90CE AA TAX
,90CF A0 03 LDY #03 ; Gerät Nr. => 1 für Kassette
,90D1 20 BA FF JSR FFBA ; Sekundäradresse 3
,90D4 A9 0D LDA #0D ; Setfls (OPEN)
,90D6 A2 F3 LDX #F3 ; Namenslänge
,90D8 A0 02 LDY #02 ; Adresse wo der Name steht
,90DA 20 BD FF JSR FFBD ; Setnam (File-name)
,90DD A9 C8 LDA #C8 ; Anfangsadresse der Autostart-Routine
,90DF 85 FB STA FB ; wird in Zero-Page gespeichert
,90E1 A9 02 LDA #02
,90E3 85 FC STA FC
,90E5 A9 FB LDA #FB
,90E7 A2 04 LDX #04
,90E9 A0 03 LDY #03 ; End Adresse des Autostart
,90EB 20 D8 FF JSR FFD8 ; SAVE Routine
,90EE A9 93 LDA #93 ; löscht den Bildschirm
,90F0 20 D2 FF JSR FFD2
,90F3 A9 9E LDA #9E ; Der Warm-Start Vektor
,90F5 8D 02 03 STA 0302 ; zeigt auf die nächste
,90F8 A9 90 LDA #90 ; Sprungadresse
,90FA 8D 03 03 STA 0303
,90FD 6C 02 03 JMP (0303)

```

---

```

,909E A9 01 LDA #01
,90A0 AA TAX ; OPEN 1,1,1
,90A1 A8 TAY ; => 1 als Sekundäradr. steht für Programm
,90A2 20 BA FF JSR FFBA ; Setfls
,90A5 A9 0D LDA #0D ; Programmname
,90A7 A2 F3 LDX #F3
,90A9 A0 02 LDY #02
,90AB 20 BD FF JSR FFBD ; Setnam
,90AE A5 2B LDA 2B
,90B0 85 FB STA FB ; lädt Anfangsadresse ins Basic-Programm
,90B2 A5 2C LDA 2C ; und speichert in Zero-Page
,90B4 85 FC STA FC
,90B6 A9 FB LDA #FB
,90B8 A6 2D LDX 2D ; lädt Endadresse ins Basic-Programm
,90BA A4 2E LDY 2E
,90BC 20 D8 FF JSR FFD8 ; SAVE Routine
,90BF A9 83 LDA #83
,90C1 8D 02 03 STA 0302 ; Original Warm-Start Inhalt
,90C4 A9 A4 LDA #A4 ; wird wieder gesetzt
,90C6 8D 03 03 STA 0303
,90C9 4C AE A7 JMP A7AE ; zurück zum Interpreter

```



```

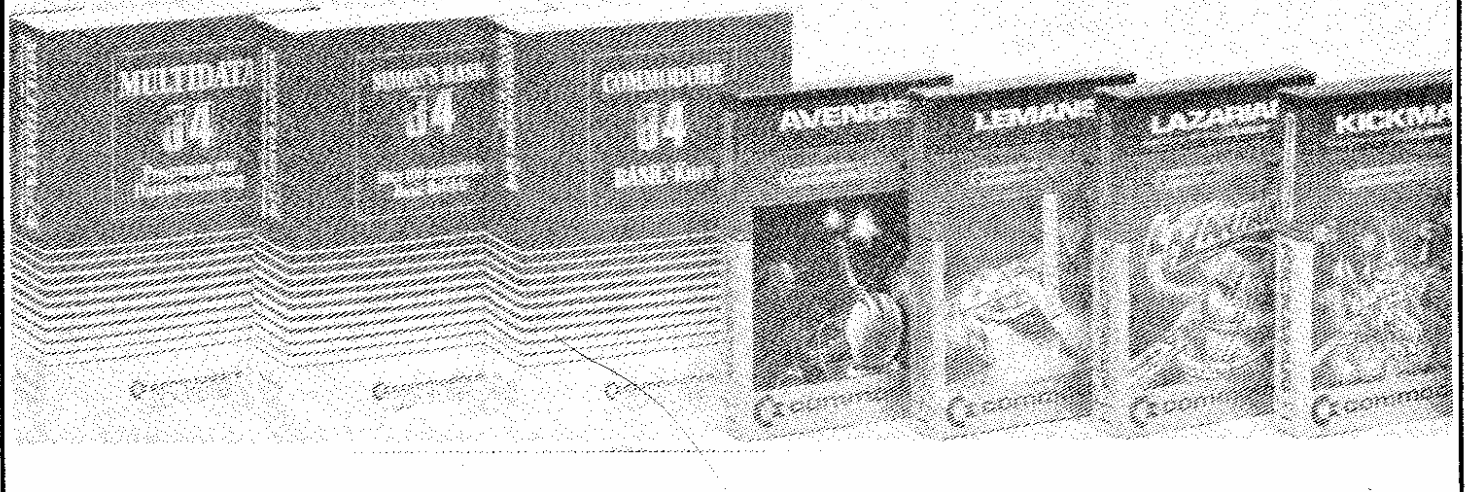
10 REM =====
11 REM ==  A U T O S T A R T  ==
12 REM =====
13 REM ==      FUER C-64      ==
14 REM ==      (C) BY (1985)  ==
15 REM ==      S. VALENTINO   ==
16 REM ==      FRANKFURTERSTR.285 ==
17 REM ==                      ==
18 REM =====
19 REM ==                      ==
20 REM =====
100 PRINT"J":FORT=1TO11:PRINT:NEXT:PRINTTAB(14)"BITTE WARTEN!"
110 FORV=36864TO37332:READQ:POKEY,Q:Q=Q+Q:NEXT
120 IFS<>51274THENPRINT"FEHLER IN DATA'S!":END
130 POKE55,0:POKE56,144:SYS36864
32000 DATA169,241,141,33,208,169,242,141,32,208,160,0,185,63,145,32,210,255,200
32001 DATA192,128,144,245,169,34,141,8,3,169,144,141,9,3,96,32,115,0,240,4,201
32002 DATA95,240,3,76,231,167,32,115,0,201,65,240,10,76,8,175,234,234,234,234
32003 DATA234,234,234,32,115,0,160,0,185,99,144,32,210,255,200,192,43,144,245
32004 DATA160,0,32,87,241,153,43,145,200,201,13,240,4,192,13,208,241,76,142,144
32005 DATA147,13,13,13,13,32,78,69,85,69,32,78,65,77,69,32,68,69,83,32,65,85
32006 DATA84,79,83,84,65,82,84,45,80,82,79,71,82,65,77,77,13,13,32,61,62,160
32007 DATA0,185,0,145,153,200,2,200,192,60,208,245,76,192,145,169,1,170,168,32
32008 DATA186,255,169,13,162,243,160,2,32,189,255,165,43,133,251,165,44,133,252
32009 DATA169,251,166,45,164,46,32,216,255,169,131,141,2,3,169,164,141,3,3,76
32010 DATA174,167,169,1,170,160,3,32,186,255,169,13,162,243,160,2,32,189,255
32011 DATA169,200,133,251,169,2,133,252,169,251,162,4,160,3,32,216,255,169,147
32012 DATA32,210,255,169,158,141,2,3,169,144,141,3,3,108,2,3,169,0,133,157,32
32013 DATA213,255,134,45,132,46,169,131,141,2,3,169,164,141,3,3,162,6,134,198
32014 DATA189,236,2,157,118,2,202,208,247,108,2,3,76,207,13,82,213,13,32,32,32
32015 DATA32,32,32,32,32,32,32,32,32,139,227,200,2,0,0,0,147,13,13,32,32,32
32016 DATA32,32,32,32,32,32,213,192,192,192,192,192,192,192,192,192,192,192,192
32017 DATA192,192,192,192,192,192,192,201,13,32,32,32,32,32,32,32,32,194
32018 DATA32,32,65,85,84,79,83,84,65,82,84,32,58,32,95,65,32,32,194,13,32,32
32019 DATA32,32,32,32,32,32,32,32,194,32,40,67,41,32,83,46,32,86,65,76,69,78
32020 DATA84,73,78,79,32,194,13,32,32,32,32,32,32,32,32,32,202,192,192,192
32021 DATA192,192,192,192,192,192,192,192,192,192,192,192,192,192,192,203,13
32022 DATA13,0,169,13,32,210,255,160,0,169,32,153,43,145,200,192,13,208,248,76
32023 DATA204,144,0

```

## Das umfangreiche Softwareprogramm

Ob Sie professionelle Programme zur Arbeitserleichterung suchen, auf einfachem Wege die Programmiersprache Basic lernen wollen oder sich von phantastischen

Spielen mitreißen lassen wollen – bei Commodore finden sie immer ausgereifte Software, die laufend erweitert wird. Mehr darüber im großen Commodore 64 Software-Katalog.



## „Sequenz 64“

### für Commodore 64

Das Programm „Sequenz 64“ poked nach dem Start mit „RUN“ zunächst die Werte für das Maschinenprogramm. Da es nicht allzu viele Data-Werte sind, wurde auf eine genauere Prüfsummenroutine verzichtet.

Dann erscheint auf dem Bildschirm das Hauptmenue, wobei man folgende sechs Unterprogramme aufrufen kann:

1. Das im Speicher befindliche Musikstück wird abgespielt. Dabei kann man die einzelnen Stimmen mit Hilfe der Funktionstasten ein- und ausschalten. Dieses wird durch einen Interrupt ermöglicht.

2. Die Parameter des SID (Sound Interface Device, Soundchip) werden gesetzt. Hierbei können für jede Stimme festgelegt werden:

a) der Wert der Wellenform, der so eingegeben werden muß, wie u. a. im Commodore Handbuch beschrieben, also:

- 128 = Rauschen
- 64 = Rechteck
- 32 = Sägezahn
- 16 = Dreieck

Es ist auch die Kombination zweier Wellenformen möglich, z. B. erzielt man mit Wellenform = 80 (Rechteck + Dreieck) einen metallischen Klang. Außerdem kann mit der Wellenform das Synchronisations-, das Ringmodulations- und das Testbild gesetzt werden, indem man den jeweiligen Wert addiert:

- Synchronisation : 2
- Ringmodulation : 4
- Test (Osz. aus) : 8

Über die Bedeutung und die Auswirkung dieser Bits kann hier nicht ausführlich gesprochen werden.

b) Die Werte der Hüllkurve (Attack, Decay, Sustain, Release; siehe Commodore-Handbuch), für die je-

weils ein Wert zwischen 0 und 15 angegeben werden muß.

c) Die Pulsbreite, die jedoch nur bei Rechteckschwingungen von Bedeutung ist.

Nach den Werten für die drei Stimmen können auch noch die Werte für die Filter gesetzt werden und zwar folgende Parameter:

a) Der Filtermodus (0 – 15), von dem die einzelnen Bits folgende Bedeutung haben:

- Bit 0 1 = Tiefpaß
- Bit 1 2 = Bandpaß
- Bit 2 4 = Hochpaß
- Bit 3 8 = Stimme 3 aus

b) Filterverteilung, die angibt, welche Stimmen über den Filter geleitet werden. Die Bedeutung der Bits:

- Bit 0 1 = Stimme 1 wird gefiltert
- Bit 1 2 = Stimme 2 wird gefiltert
- Bit 2 4 = Stimme 3 wird gefiltert
- Bit 3 8 = Signal am Audio-In-Eingang wird gefiltert

c) Die Filterfrequenz (0 – 2047) gibt an, ab welcher Frequenz der Filter arbeiten soll.

d) Die Filterresonanz (8 – 15) gibt an, wie stark der Filter mit der gefilterten Stimme „mitschwingt“.

Schließlich kann auch noch die Lautstärke (0 – 15) sowie die Geschwindigkeit des Abspielens (0 – 255) eingegeben werden, wobei 1 die schnellste und 255 die langsamste Geschwindigkeit ist. (0 ist ungefähr genauso langsam wie 255).

3. Hier wird die eigentliche Melodie eingegeben und zwar ist für einen Ton die Eingabe von drei Werten notwendig:

a) Angabe der Oktave des Tones (0 – 7), wobei 0 die tiefste, 7 die höchste Oktave ist.

b) Hier wird die eigentliche Note eingegeben. Zulässig sind 14 ver-

schiedene Zeichen, davon 12 für die Töne der Oktave. Die Zeichen sind im Programm dargestellt.

Außerdem ist ein Pausenzeichen vorgesehen, das wie eine normale Note gezählt wird. Mit dem Zeichen „Z“ kann man in der Eingaberoutine einen Schritt zurückgehen, das heißt, der vorher eingegebene Wert wird angezeigt und kann neu bearbeitet werden. Wenn man vom 1. Ton zurückgehen will, so geht das Programm zum Hauptmenue zurück, unabhängig davon, welche Stimme gerade bearbeitet wurde.

c) Die Dauer eines Tones wird angegeben im Kehrwert des richtigen Notenwertes. Hat man also z. B. einen Notenwert von 1/2, so gibt man 2 ein, denn 2 ist der Kehrwert von 1/2. Das gilt auch für kompliziertere Notenwerte, wie z. B. eine punktierte Viertel, die einen Notenwert von 3/8 hat. Auch hier muß der Kehrwert gebildet werden. (3/8; Kehrwert = 8/3) 8/3 ist gleich 2:66, also muß man eingeben: 2.66666666.

Bei der Eingabe von 0 als Dauer schreibt das Programm in die gerade zu bearbeitende Speicherzelle den Code für Pause und die Dauer 0, woran das Maschinenprogramm erkennt, daß diese Stimme fertig ist. Das Eingabeprogramm geht dann zur Eingabe der Noten für die nächste Stimme über.

4./5. Speichern und Laden.

Dafür sind zwei Maschinenspracheroutinen vorgesehen, die Daten der Melodie aller drei Stimmen und die zuletzt verwendeten SID-Werte auf Cassette schreibt, bzw. von Cassette liest. Es handelt sich bei dem Speicherbereich um die Adressen 38400 – 40959 (hexadezimal : 9600 – 9 fff).

6. Oktav-Änderungen

Hierbei hat man die Möglichkeit, die gesamten Noten einer Stimme um eine Oktave nach oben oder nach unten zu verändern.

Noch etwas zum Maschinenprogramm:

Die Daten liegen jeweils ab der folgenden Adresse:

Stimme 1 : 9600 (hexadezimal)

Stimme 2 : 9900

Stimme 3 : 9c00

Jede Stimme hat also 768 Speicher-

plätze zur Verfügung, das heißt, daß 384 Töne pro Stimme gespielt werden können, weil jeder Ton aus zwei Werten besteht (Tonzahl und Dauer).

Die Tonzahl gibt an, welche Frequenzwerte das Maschinenprogramm aus einer Tabelle liest. Die Dauer wird in einen Zähler geschrie-

ben, der bei jedem Durchlauf heruntergezählt wird. Ist dieser Zähler auf Null, dann wird der nächste Wert geholt, usw.

Der Interrupt setzt je nach Tastendruck das entsprechende Testbit der betreffenden Stimme, bzw. löscht es, so daß der Oszillator der Stimme ausgeschaltet wird.

```

10 POKE53280,5:POKE53281,13:POKE646,0
20 POKE51,0:POKE52,144:POKE53,0:POKE54,144:POKE55,0:POKE56,144:POKE650,255
30 DIMI$(13),PS(21),A2(21)
40 GOSUB 10000
50 PRINT"##### SEQUENZER 64 # "
60 PRINT"##### MELODIE ABSPIELEN"
70 PRINT"##### SII-WERTE AENDERN"
80 PRINT"##### NEUE MELODIE EINGEBEN"
100 PRINT"##### MELODIE ABSPEICHERN"
110 PRINT"##### MELODIE LADEN"
120 PRINT"##### OKTAV-AENDERUNGEN"
130 GETA$:IFA$=""THEN130
140 A=VAL(A$):IFAC10RAD>6THEN130
150 ON A GOTO 200,250,800,1200,1300,1400
160 GOTO130
200 GOSUB 2000
210 SYS36864:GOTO50
250 PRINTCHR$(147)
260 FORT=1T03
270 W(T)=PEEK(40928+(T-1)*7+4)
280 AD(T)=PEEK(40928+(T-1)*7+5)
290 SR(T)=PEEK(40928+(T-1)*7+6)
300 TL(T)=PEEK(40928+(T-1)*7+2)
310 TH(T)=PEEK(40928+(T-1)*7+3)
320 NEXTT
330 RG=PEEK(40928+23):LG=PEEK(40928+24)
340 QL=PEEK(40928+21):QH=PEEK(40928+22)
350 FR=(RGAND240)/16:FV=RGAND15
360 FQ=QH*8+QL:F1=(LGAND240)/16:LA=LGAND15
365 GE=PEEK(37425)
370 FORT=1T03
380 PRINT"#####",STIMME"
385 PRINT"#####WELLENFORM: ";W(T)AND254
390 PRINT"#####ATTACK: ";(AD(T)AND240)/16
400 PRINT"#####WEDAY: ";AD(T)AND15
410 PRINT"#####SUSTAIN: ";(SR(T)AND240)/16
420 PRINT"#####RELEASE: ";SR(T)AND15
430 PRINT"#####PULSBREITE: ";TL(T)+256*TH(T)
440 INPUT"#####";W(T)
445 INPUT"#####";AD(T)
450 INPUT"#####";D(T)
460 INPUT"#####";S(T)
470 INPUT"#####";R(T)
480 INPUT"#####";TA(T)
490 IFAC10<D(T)>15THEN440
492 IFD(T)<D(T)>15THEN440
494 IFS(T)<D(T)>15THEN440
496 IFR(T)<D(T)>15THEN440
498 IFTA(T)<D(T)>4095THEN440
500 AD(T)=A(T)*16+D(T)
510 SR(T)=S(T)*16+R(T)
520 TH(T)=INT(TA(T)/256):TL(T)=TA(T)-TH(T)*256
530 POKE40928+(T-1)*7+5,AD(T)
540 POKE40928+(T-1)*7+6,SR(T)
550 POKE40928+(T-1)*7+2,TL(T)
560 POKE40928+(T-1)*7+3,TH(T)

```

## Commodore 64 Anwenderhandbuch John Heilborn & Ran Talbott

Dieses leicht verständliche, durchgehend illustrierte Anwenderhandbuch vermittelt das nötige Wissen für den Umgang mit dem Commodore 64 und seinen Zusatzgeräten. Dem Anfänger bietet das Buch

- 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.

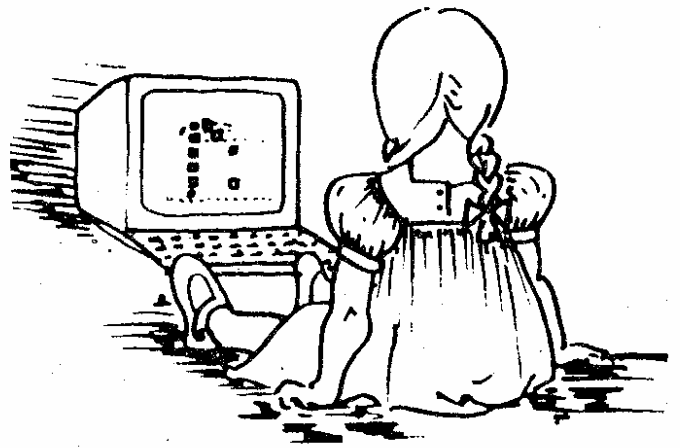
Dem fortgeschrittenen Benutzer dient das Buch als unschätzbare Arbeitsmittel und Nachschlagewerk. Es enthält eine detaillierte Übersicht aller BASIC-Statements und Funktionen, ein ausführliches Verzeichnis der Speicheradressierung zusammen mit einer Erklärung, was jede adressierbare Speicherzelle leistet und wie sie arbeitet, und einen besonderen Abschnitt zur fortgeschrittenen Colorgrafik und Sprite-Grafik.

ISBN 3-89028-016-1, April 1984

446 Seiten, DM 39,80

# programme

```
570 POKE40928+(T-1)*7+4,W(T)
580 NEXT I
590 PRINTCHR$(147)
600 PRINT"FILTERMODUS : ";FI
610 PRINT"FILTERVERTEILUNG: ";FV
620 PRINT"FILTERFREQUENZ : ";FQ
630 PRINT"FILTERRESONANZ : ";FR
640 PRINT"LAUTSTÄRKE : ";LA
650 PRINT"GESCHWINDIGKEIT : ";GE
660 INPUT"#####";FI
670 INPUT"#####";FV
680 INPUT"#####";FQ
690 INPUT"#####";FR
700 INPUT"#####";LA
710 INPUT"#####";GE
720 IFFI<0DRFI>15THEN660
722 IFFV<0DRFV>15THEN660
724 IFFR<0DRFR>15 THEN660
726 IFLA<0DRLA>15THEN660
728 IFGE<0DRGE>255THEN660
730 IFFQ<0DRFQ>2047THEN660
740 LG=LA+FI*16
750 RG=FV+FR*16
760 QL=FQAND7:QH=(FQ-QL)/8
770 POKE40928+21,QL:POKE40928+22,QH
780 POKE40928+23,RG:POKE40928+24,LG
785 POKE37425,GE
790 GOTO50
800 PRINT"FOLGENDE NOTENSYMBOLE SIND ZULÄSSIG:"
810 FORT=0T013
820 PRINT" "T$(T)
830 NEXT T
832 PRINT"ALTE NOTEN LÖSCHEN (J/N) ?"
833 GETA$: IFA$="N"THEN850
835 IFA$<>"J"THEN833
840 FORT=38400T040703:POKET,0:NEXTT
850 PRINT"NEUE EINGABE DER NOTEN"
860 FORI=0T02:T=0
870 PRINT"#####";I+1;". STIMME"
880 GOSUB3000
890 INPUT"#####OKTAVE (0-7)";O
900 INPUT"#####NOTE ";N$
910 N=N$
920 IFT$(N)=N$THEN950
930 N=N+1
940 IFN<14THEN920
950 IFN=12THENO=0:N=100:FL=1
960 IFN=13THENT=F-2:IFT>=0THENGOSUB3000:GOTO880
965 IFT<0THENGOTO50
970 IFN>=14ANDN<100THEN880
980 INPUT"#####DAUER (1/X) ";D
990 IFD=0THEN1070
1000 NW=O*12+N
1010 IFINT(192/D+.5)<0DRINT(192/D+.5)>255THEN880
1020 POKE38400+I*768+T,NW
1030 POKE38401+I*768+T,INT(192/D+.5)
1040 T=T+2
1050 IFT>768THENGOTO1090
1060 GOTO870
1070 POKE38400+I*768+T,100
1080 POKE38401+I*768+T,0
1090 NEXT I
1100 GOTO50
1200 INPUT"##### BEZEICHNUNG (7 ZEICHEN) ";ZE$
1210 IFLLEN(ZE$)<7THENFORT=1T07-LEN(ZE$):ZE$=ZE$+" ":NEXTT
1220 FORT=1T07:ZE=ASC(MID$(ZE$,T,1)):POKE37686+T,ZE:NEXTT
1230 PRINT"#####ILRETURN) DRUECKEN, WENN BAND BEREIT"
1240 GETA$: IFA$<>CHR$(13)THEN1240
1250 PRINT"#####";SY$37648
```





# Da steckt Musik drin

## SIGHT & SOUND

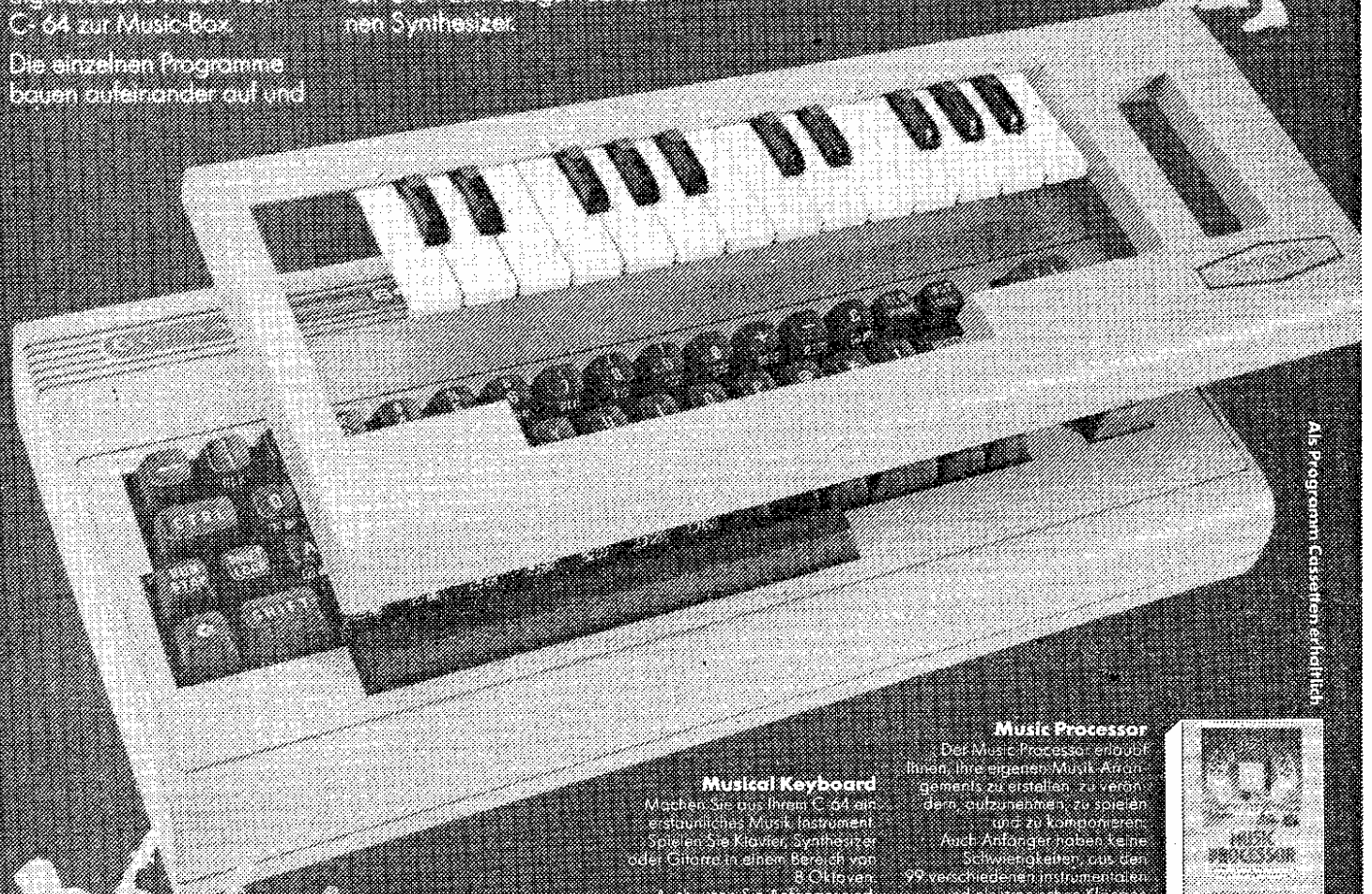
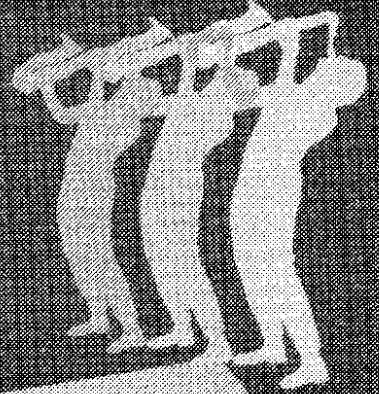
Music Software, Inc.

Das sind Musikprogramme für verschiedenste Anwendungen. Ob für Anfänger oder für Hobbykomponisten, ob zum Lernen oder Arrangieren.

Sight & Sound macht den C-64 zur Music-Box.

Die einzelnen Programme bauen aufeinander auf und

sind untereinander datenkompabil – für Ihre Ideenvielfalt. Die Bedienung kann natürlich über die Tastatur erfolgen. Mit dem abgebildeten Keyboard jedoch wird der C-64 zum ausgewachsenen Synthesizer.



Als Programm-Cassetten erhältlich

### Musical Keyboard

Machen Sie aus Ihrem C-64 ein erstaunliches Musik-Instrument. Spielen Sie Klavier, Synthesizer oder Gitarre in einem Bereich von 8 Oktaven. Auch wenn Sie Anfänger sind, brauchen Sie keine Angst zu haben.

### Kawasaki Synthesizer

Dieses Programm macht aus Ihrem Computer einen programmierbaren Synthesizer und Musik-Processor. Der Künstler: Kontrolle über Klang, Vibration, Tonhöhe und spezielle Effekte. 2. Der Komponist: Vier bedienungsfreundliche Bildschirmdarstellungen und ein Dreispur-Aufnahmeverfahren ermöglichen Ihnen auf dynamischem Weg Musikstücke zu komponieren.

### 3001 Sound Odyssey

3001 vermittelt Ihnen die Grundlage künstlicher Musik, indem Sie lernen, eigene Klangeffekte und Musik zu erzeugen. Mit dem umfangreichen Erläuterungsprogramm können Sie alles über synthetische Musik lernen.

### Music Processor

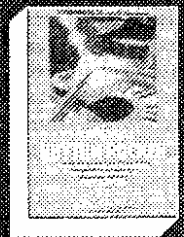
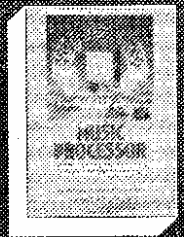
Der Music Processor erlaubt Ihnen, Ihre eigenen Musik-Arrangements zu erstellen, zu verändern, aufzunehmen, zu spielen und zu komponieren. Auch Anfänger haben keine Schwierigkeiten, aus den 99 verschiedenen instrumentalen und elektronischen Klängen eigene Melodien zu entwickeln.

### Kawasaki Rhythm Rocker

Mit diesem Programm können Sie elektronisch Rhythmus erzeugen und diesen gleichzeitig mit hochauflösender Farbgraphik darstellen. Spielen Sie Ihre eigenen Melodien über die barbits im Programm zur Verfügung stehenden Bass- und Rhythmusvorläufer, die schon in bekannten Hits Verwendung fanden.

### On-Stage

Mit diesem Programm können Sie dem computergesteuerten Verstärker Ihrer ganz persönlichen Hi-Touch-Box Kompatibilität, Rhythmus, Melodie und Harmonien. Ihnen gefallt nicht, was Sie hören? Kein Problem – mit dem Joystick können Sie Klang und spezielle Effekte ganz nach Ihren Wünschen ändern.



**ariolasoft**  
Qualität ist unser Programm!





## SEIKOSHA's neue Standard-Matrixdrucker-Familie für Homecomputer

Um zwei Modelle erweitert hat SEIKOSHA (Microscan GmbH, Überseering 31, 2000 Hamburg 60, Tel. 0 40 / 63 20 03-0) die Standard-Matrixdrucker-Serie GP-500:

### GP-500VC:

Der preiswerte 50-Zeichen/Sek.-Drucker für Commodore-Computer C-16, VC-20, C64, C128 und Plus 4,

problemlos über das mitgelieferte Kabel anschließbar. Bei einer Matrix-Konfiguration von 5 x 7 Punkten werden Commodore-Blockgrafik bei bis zu 80 Zeichen/Zeile dargestellt. Hochauflösende Grafik durch Punktadressierung mit bis zu 480

Punkten/Zeile kann eigendefiniert werden. Zwei verschiedene Zeilenvorschübe von 6 Zeilen/Zoll und 9 Zeilen/Zoll bei Vorschubgeschwindigkeiten von 10 Zeilen/Sek. und 15 Zeilen/Sek. sind wählbar, alphanumerische Zeichen sind in Normalgröße oder in doppelter Breite ausdrückbar. Der Drucker druckt auf Standard-EDV-Papier bis zum maximalen Format DIN A4. Der Papiertransport erfolgt über Stachelrad durch Schrittmotor.

### GP-500A:

Der Universal-Drucker für Computer, die über eine Centronics-Schnittstelle verfügen. 50 Zeichen/Sek. unidirektional druckend bei einer Matrix-Konfiguration von 5 x 7 Punkten und einem Zeichenvorrat von 139 Groß-/Kleinbuchstaben, Zahlen und Symbolen.

Alle anderen Daten sind identisch mit GP-500 CPC.

## Farbgrafik-Drucker von SEIKOSHA für Commodore

Zwei neue Modelle hat SEIKOSHA der Farb-Matrixdrucker-Serie GP-700 hinzugefügt:

### GP-700VC:

Über ein mitgeliefertes Kabel direkt anschließbar sind C-16, VC-20, C-64, C128 und Plus 4.

Mit 38 Zeichen/Sek. Ausdruck von Groß- und Kleinbuchstaben, Ziffern, Symbolen und Commodore-Blockgrafik bei bis zu 80 Zeichen/Zeile auf Standard-EDV-Papier oder Einzelblatt-Papier bis zum Format DIN A4. Matrix-Konfiguration 8 x 8 Punkte, hochauflösende Grafik durch Punktadressierung mit bis zu 640 Punkten-

/Zeile, selbst definierbar, Zeilenvorschub  $\frac{1}{6}$  Zoll fest und in Vielfachen von  $\frac{1}{120}$  Zoll softwaremäßig wählbar sind nur einige Besonderheiten dieses bewährten und preisgünstigen Druckers.

### SP-1000VC

mit serieller IEEE-Schnittstelle, über mitgeliefertes Drucker-kabel direkt an Commodore C-16, VC-20, C-64, C-128 und Plus 4 anschließbar. Druck von sämtlichen Commodore-Zeichen einschließlich Blockgrafik sowie eigendefinierte Grafik durch Einzeldelansteuerung.

## Commodore 64-Fans

waren von „Mein erstes Commodore 64 Programm“ – ebenfalls von SYBEX-Gründer Rodnay Zaks – begeistert. Mit **Mein zweites Commodore 64 Programm** von Gary Lippman lernen sie, den zweiten BASIC-Schritt auf dem C 64 zu machen. Ziel des Buches ist es, nach kurzer Zeit selbst nützliche Programme schreiben zu können. Der unterhaltende Stil des Autors und viele lustige Illustrationen machen das Buch für jede Altersgruppe geeignet. **Mein zweites Commodore 64 Programm** (Best.-Nr. 3086) wird ab Februar ausgeliefert und kostet DM 32,-.

Sybox Verlag, Düsseldorf





```
600 IFPR<>8378THENZA=36000:ZE=36090:GOTO62000
700 REM SPRITES
710 REM =====
720 PRINT"!!!!GLEICH GEHT'S LOS !!!":PR=0
730 FORI=0TO62:READT:PR=PR+T:POKE832+I,T:NEXTI
735 FORI=0TO65:READT:PR=PR+T:POKE832+62+I,T:NEXTI
800 IFPR<>3665THENZA=41000:ZE=41120:GOTO62000
850 PR=0
900 READX:IFX=-1THEN1000
920 POKEUI+9*4096+5*256,X:PR=PR+X:UI=UI+1:GOTO900
1000 POKE2044,192:POKE2045,193:UI=0
1010 READX:IFX=-1THEN1100
1020 POKE3*4096+UI,X:PR=PR+X:UI=UI+1:GOTO1010
1100 IFPR<>20775THENZA=52000:ZE=53070:GOTO62000
1200 UI=0:PR=0
1210 READX:IFX=-1THEN1300
1220 POKE4*4096+UI,X:PR=PR+X:UI=UI+1:GOTO1210
1300 UI=0
1310 READX:IFX=-1THEN1380
1320 POKE5*4096+UI,X:PR=PR+X:UI=UI+1:GOTO1310
1380 IFPR<>4781THENZA=53500:ZE=54000:GOTO62000
1400 UI=0:PR=0
1410 READX:IFX=-1THEN1500
1420 POKE4*4096+2*256+UI,X:PR=PR+X:UI=UI+1:GOTO1410
1500 UI=0:ZS=1:ZO=14
1510 READX:IFX=-1THEN1550
1520 POKE5*4096+5*256+UI,X:PR=PR+X:UI=UI+1:GOTO1510
```

weiter nächste Seite

## Schon gewußt?

# Der Homecomputer des Jahres 1984

## Commodore 64 erhält zum zweiten Mal die begehrte Auszeichnung

Frankfurt/Main, den 14. Dezember 1984. – Eine internationale Fachpresse-Jury wählte im November den „Homecomputer des Jahres 1984“. Unter namhaften Konkurrenten erhielt der Commodore 64 zum zweiten Mal nach 1983 die begehrte Auszeichnung. Die Jury setzte sich aus Redakteuren bekannter Fachzeitschriften in England, Frankreich, Holland, Italien, Spanien, den USA und der Bundesrepublik Deutschland zusammen. Die Ehrung wird jeweils zum Jahresende von der Zeitschrift CHIP verliehen (siehe auch CHIP 12/84).



Alwin Stumpf, Geschäftsführer Deutschland-Vertrieb, erhält aus den Händen von Richard Kerler, CHIP, die begehrte Auszeichnung „Homecomputer des Jahres 1984“ für den Commodore 64. (von links: Harald Speyer/Commodore, Richard Kerler/CHIP, Alwin Stumpf/Commodore, Herbert Freese/CHIP)

```

1550 IFPR<4643THENZA=55000:ZE=56000:GOTO62000
1600 PR=0:FORI=0T062:READX:PR=PR+X:POKE194*64+I,X:NEXT:PRINT"X"
1700 IFPR<6180THENZA=57000:ZE=57000:GOTO62000
13400 POKEV+8,100:POKEV+9,77:POKEV+10,154:POKEV+11,78:POKEV+21,51
13440 FORI=0T03:POKEV+I,0:NEXT
13900 POKEV+23,51:POKEV+29,51:POKE53281,10
13910 FORI=0T02S:POKEV+37,1:POKEV+38,0:FORT=0T099:NEXTT:POKEV+37,0:POKEV+38,7
13920 FORT=0T099:NEXTT,1:FORT=0T020:NEXTT:POKEV+37,0:POKEV+38,2:POKE53281,6
14000 SYS5*4096:SYS5*256+5*4096:SYS9*4096+5*256
14050 POKESI,0:POKESI+1,0:Z0=110
14100 FORI=77T00STEP-1:POKEV+9,I:POKEV+11,I:NEXTI
14900 POKEV+23,0:POKEV+29,128:PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
14910 FORI=0T03:PRINT"X";:NEXT
14920 PRINT"X";:NEXT
14930 POKE56295,4:POKE2023,160:POKEV+16,128:POKEV+14,29:POKEV+15,196:ZS=1
15000 POKEV+21,143:SYSAN:POKESI,0:POKESI+1,0
15900 PU=PEEK(7*4096+8)+PEEK(7*4096+2*16)
16000 PRINT"X"■SIE FINGEN ";PU;" UFO'S AB !"
16500 FORT=0T09T3:NEXT
17000 PRINT"X"NOCH MAL (J/N) ? "
17010 GETF#:IFF#="J"THENPRINT"X":POKEV+21,0:GOTO13400
17020 IFF#="N"THEN19000
17030 GOTO17000
19000 POKEV+21,0:PRINT"X":END
31999 REM MASCHINEN-DATA'S
32000 DATA162,195,142,0,112,142,1,112,160,48,140,2,112,162,119,142,4,112,169
32001 DATA212,141,6,112,162,1,142,3,112,140,5,112,141,7,112,7,112,234,142,14
32002 DATA112,234,162,0,142,9,112,234,234,234,162,48,142,16,112,142,17,112,234
32003 DATA234,162,0,142,8,112,142,9,112,142,10,112,142,11,112,142,12,112,76,80
32004 DATA130,234,174,0,220,224,119,240,7,224,123,240,17,76,119,128,174,0,112
32005 DATA224,255,240,18,232,142,0,112,76,119,128,174,0,112,224,0,240,4,202,142
32006 DATA0,112,174,30,208,224,0,240,72,224,3,240,11,224,5,240,27,224,9,240,43
32007 DATA76,198,128,172,8,112,200,140,8,112,165,162,141,2,112,169,1,141,3,112
32008 DATA76,198,128,172,8,112,200,140,8,112,165,162,141,4,112,169,11,141,5,112
32009 DATA76,198,128,172,8,112,200,140,8,112,165,162,141,6,112,169,19,141,7,112
32010 DATA162,0,174,9,112,224,255,240,4,232,76,10,129,172,3,112,200,140,3,112
32011 DATA172,5,112,200,140,5,112,172,7,112,200,140,7,112,174,10,112,188,0,96
32012 DATA232,142,10,112,162,15,142,24,212,162,1,142,5,212,162,17,142,6,212,142
32013 DATA4,212,140,1,212,76,255,130,142,9,112,162,0,174,0,112,142,0,208,174
32014 DATA1,112,142,1,208,174,2,112,142,2,208,174,3,112,142,3,208,174,4,112,142
32015 DATA4,208,174,5,112,142,5,208,174,6,112,142,6,208,174,7,112,142,7,208,162
32016 DATA7,142,39,208,162,7,142,40,208,142,41,208,142,42,208,162,0,160,0,174
32017 DATA11,112,172,12,112,232,224,10,208,251,174,11,112,200,192,10,208,243
32018 DATA162,7,142,39,208,162,7,142,40,208,142,41,208,142,42,208,174,3,112,224
32019 DATA208,240,17,174,5,112,224,208,240,10,174,7,112,224,209,240,3,76,143,-1
32020 DATA129,96,174,9,112,224,16,240,11,224,144,240,7,224,209,240,3,76,164,129
32021 DATA76,211,128,162,33,142,4,212,174,8,112,224,255,240,3,76,76,128,162,0
32022 DATA142,8,112,174,32,112,232,142,32,112,76,76,128,76,177,128,76,213,128
32023 DATA0,159,117,255,21,159,117,234,21,159,117,255,21,159,117,255,5,159,117
32024 DATA255,21,159,117,98,21,138,117,234,21,159,117,255,0,159,117,255,0,159
32025 DATA117,234,1,159,117,245,10,159,96,255,21,159,117,255,21,159,96,117,104
32026 DATA234,24,159,234,234,0,154,96,234,0,138,96,234,0,130,96,232,0,138,96
32027 DATA234,21,154,96,234,0,138,117,234,0,0,225,117,0,0,96,234,0,138,112,234
32028 DATA0,138,96,234,0,138,96,234,0,0,96,96,29,8,96,96,0,138,234,234,0,8,234
32029 DATA234,0,138,96,234,138,138,96,234,0,138,96,234,0,138,174,30,208,224,9
32030 DATA240,11,224,5,240,7,224,3,240,3,76,128,130,160,0,200,192,247,240,6,140
32031 DATA33,208,76,100,130,76,128,130,0,138,117,234,21,138,96,234,0,138,117
32032 DATA234,0,159,174,8,112,224,255,240,3,76,80,128,162,0,142,8,112,174,32
32033 DATA112,232,142,32,112,76,80,128,255,21,159,247,255,0,159,117,234,159,155
32034 DATA117,255,21,159,117,255,21,159,117,255,16,159,117,255,21,159,117,255
32035 DATA21,159,117,255,21,159,117,255,21,159,117,255,21,159,117,255,138,138
32036 DATA96,255,21,159,117,255,21,138,117,255,21,159,117,255,21,159,117,255
32037 DATA21,159,117,255,21,130,117,234,21,154,117,255,21,159,96,255,21,159,112
32038 DATA255,21,138,117,255,21,149,234,255,16,159,117,255,21,162,0,142,9,174
32039 DATA17,112,142,9,112,174,16,112,224,2,240,7,232,142,16,112,76,13,129,160
32040 DATA0,140,16,112,174,17,112,232,142,17,112,142,9,112,76,13,129,192,68,240
32041 DATA12,200,140,14,112,160,0,140,15,112,76,13,129,162,0,172,13,112,142,14
32042 DATA112,200,140,9,112,140,13,112,76,13,129,234
35000 DATA-1

```





# 6502-Assembler-Kurs

## Teil 1

Die Hardware des Commodore 64 ist in dieser Preisklasse wohl kaum übertroffen. Leider ist das Basic dieses Rechners kaum geeignet diese Vorzüge auszuspielen. Es wurde wohl mehr Wert auf die Kompatibilität zu den alten Rechnern CBM und PET gelegt. Da der PET einer der ersten Microcomputer war, ist das Basic auch entsprechend veraltet.

Um nun diesen Mangel auszugleichen gibt es zwei Möglichkeiten:

1. eine Basic-Erweiterung zuzulegen  
Vorteil: leistungsfähige Befehle einfach programmierbar  
Nachteil: nicht mehr kompatibel
2. in Assembler zu programmieren  
Vorteil: schnelle Verarbeitungsgeschwindigkeit, geringer Speicherplatzbedarf  
Nachteil: schwierig zu programmieren

Dieser Kurs ist für diejenigen gedacht, die an die Grenzen des Commodore-Basic gestoßen sind und sich für die Assemblerprogrammierung entschlossen haben.

Ziel dieses Kurses ist es, dem Anfänger die Grundkenntnisse des 6502-Assemblers nahezubringen. Unter-

stützt wird dieser Kurs von kleinen Beispielprogrammen, welche auf dem Commodore 64 lauffähig sind.

### GRUNDLAGEN Assembler ??????

Assembler ist im Gegensatz zu Basic eine Programmiersprache, die direkt auf den jeweiligen Microprozessor zugeschnitten ist. Die eigentlichen Befehle bestehen nur aus Zahlenfolgen (Objektcode), die vom Microprozessor interpretiert und ausgeführt werden. Da der Objektcode direkt auf den Microprozessor zugeschnitten ist, spricht man auch von dem Maschinencode. Da das Auswendiglernen von langen Zahlenfolgen wohl nicht sehr attraktiv ist, wurde vom Hersteller jedem Befehl eine kurze Bezeichnung zugeordnet, dem sogenannten Mnemonik. Um ein mit Mnemoniks geschriebenes Programm (Assemblerprogramm) in den Maschinencode umzusetzen, benötigen wir den Assembler. Bevor wir jedoch zur eigentlichen Assembler-Programmierung übergehen, wollen wir uns einige Grundkenntnisse verschaffen.



### Verschiedene Zahlensysteme:

In der Digital-Elektronik sowie in Computern werden Informationen in Form von Bitfolgen verarbeitet. Ein Bit kann zwei Zustände annehmen: logisch 0

(keine Spannung; low Pegel)

logisch 1

(5 Volt Spannung; high Pegel)

Die meisten Microcomputer, auch der C 64, arbeiten mit einem sogenannten 8 Bit Prozessor. Dies bedeutet, daß 8 Bit gleichzeitig bearbeitet werden können. Der Speicher eines solchen Rechners ist so kompliziert, daß 8 Bit zu einem BYTE zusammengefaßt werden und nur eine Speicherstelle belegen. Der Commodore 64 kann etwa 60 kilo Byte verarbeiten, das bedeutet, 60 000 Bytes mal 8 Bit, also 480 000 Bits (Spannungszustände).

Um das Binärsystem (Dualsystem) richtig zu verstehen, müssen wir uns noch einmal mit dem Dezimalsystem beschäftigen. Jedem Leser dürfte dieses bekannt sein, da er wohl tagtäglich damit konfrontiert wird.

Das Dezimalsystem besitzt die Basis 10. Dies bedeutet, daß 10 verschiedene Ziffern zur Verfügung stehen (0, 1, 2, 3, 4, 5, 6, 7, 8, 9).

Nehmen wir einmal die Zahl 234

Diese Zahl besteht aus 3 Ziffern. Die Zahl besitzt 2 Hunderter, 3 Zehner und 4 Einer.

$$234 = 2 \cdot 100 + 3 \cdot 10 + 4 \cdot 1$$

#### Das Binärsystem:

Wie schon erwähnt, kennt man im Binär-System nur zwei Zustände (Eins/Null), also besitzt dieses die Basis 2.

Nehmen wir einmal die Zahl

10110111

$$10110111 = 1 \cdot 128 + 0 \cdot 64 + 1 \cdot 32 + 1 \cdot 16 + 0 \cdot 8 + 1 \cdot 4 + 1 \cdot 2 + 1 \cdot 1$$

Das Dualsystem ist sehr einfach aufgebaut. Große Zahlen besitzen allerdings sehr viele Stellen, was die Lesbarkeit dieser Zahlen nicht gerade fördert.

Folgende Tabelle erleichtert das Umrechnen von Binärzahlen in das dezimale Zahlensystem:

Bit	7	6	5	4	3	2	1	0
Multiplikator	128	64	32	16	08	04	02	01
Beispielzahl	1	1	1	0	1	1	1	1

Um zu errechnen, welchen Wert die Binärzahl besitzt, trägt man diese Bitweise in die Tabelle ein und multipliziert Bit für Bit mit dem entsprechenden Multiplikator.

Unser Beispiel errechnet sich wie folgt:

Bit 7	128	
6	64	
5	32	
4	0	
3	8	
2	x 4	
1	2	
0	1	

Summe

239 entspricht 11101111

Wir wollen nun einmal versuchen, eine dezimale Zahl in eine Binärzahl umzuwandeln. Als Beispiel nehmen wir die Zahl 123. Um nun festzustellen, welchen Binärwert diese Zahl besitzt, müssen wir diese durch die bekannten Multiplikatoren dividieren.

Fangen wir an:

- 123 / 128 (bit 7) = 0 Rest 123
- 123 / 64 (bit 6) = 1 Rest 59
- 59 / 32 (bit 5) = 1 Rest 27
- 27 / 16 (bit 4) = 1 Rest 11
- 11 / 8 (bit 3) = 1 Rest 3
- 3 / 4 (bit 2) = 0 Rest 3
- 3 / 2 (bit 1) = 1 Rest 1
- 1 / 1 (bit 0) = 1 Rest 0

Die entsprechende Binärzahl lautet 01111011

Der 6502 Prozessor besitzt einen Datenbus von 8 Bit und einen Adreßbus von 16 Bit. Versuchen Sie einmal zu errechnen, welchen Dezimalwert der Daten- und der Adreßbus annehmen kann.

### Das Sedezimale Zahlensystem (Hexadezimalsystem):

Neben dem Binärsystem ist auch dieses sehr wichtig für den Programmierer. Der Vorteil der „Hex-Zahlen“ ist, daß diese nur zwei Ziffern benötigen um eine 8 Bit Zahl darzustellen. Das Hexadezimalsystem besitzt als Basis die Zahl 16. Daher benötigt man zur Darstellung einer Zahl, 16 verschiedene Ziffern. Da die 10 Ziffern des Dezimalsystems nicht ausreichen, wurde das Alphabet herangezogen.

Dezimal	Binär	Hexadezimal
0	00000000	00
1	00000001	01
2	00000010	02
3	00000011	03
4	00000100	04
5	00000101	05
6	00000110	06
7	00000111	07
8	00001000	08
9	00001001	09
10	00001010	0A
11	00001011	0B
12	00001100	0C
13	00001101	0D
14	00001110	0E
15	00001111	0F
16	00010000	10

Eine vierstellige Hexzahl wird wie folgt berechnet:

$$4DB6 = 4*4096 + 13*256 + 11*16 + 6*1 = 19894$$

Mit dieser Tabelle lassen sich Hex-Zahlen sehr leicht in Dezimalzahlen umwandeln:

Stelle	4	3	2	1
Multiplikator	4096	0256	0016	0001
Beispielzahl	A	B	1	2

Unser Beispiel errechnet sich wie folgt:

Stelle 4	A*4096	=	10*4096
Stelle 3	B*256	=	11*256
Stelle 2	1*16	=	1*16
Stelle 1	2*1	=	2*1

Summe = 43794 = 1010101100010010

Etwas schwieriger ist die Umwandlung einer Dezimalzahl in eine Hexadezimalzahl. Wir wollen dies einmal mit der Zahl 1000 probieren:

- 1000/4096 (Stelle 4) = 0 Rest 1000
- 1000/ 256 (Stelle 3) = 3 Rest 232
- 232/ 16 (Stelle 2) = 14 Rest 8
- 8/ 1 (Stelle 1) = 8 Rest 0

Das Ergebnis wäre:

$$0*4096 + 3*256 + 14*16 + 8*1 = 03E8 \text{ Hex}$$

Hexadezimalzahlen lassen sich wesentlich besser eingeben als Binär- oder Dezimalzahlen, da sie kürzer in der Schreibweise sind und lange nicht so ermüdend bei der Überprüfung. Fehler sind wesentlich leichter in einer Folge von Hexadezimalziffern zu finden als in einer dezimal ausgedruckten Zahlenfolge.

Da beim Programmieren oft mehrere Zahlensysteme benutzt werden, ist es erforderlich, diese durch ein vorangestelltes Zeichen zu kennzeichnen. Wir benutzen hierfür die folgenden Symbole:

·/. Binärzahl z. B. ·/.110011111 oder ·/.110

\$ Hexzahl z. B. \$03EA oder \$AF

Wird der Zahl kein Zeichen vorangestellt, so handelt es sich um eine Dezimalzahl.

### Literatur

- 6502 Microcomputer, Programmierung, Peter Heuer, Hofacker Verlag
  - 6502 Programmieren in Assembler, Lance A. Leventhal, Te-Wi Verlag
  - 64 Intern, Angershausen, Becker, Englisch, Gerits, Data Becker Buch
- Frank Brall

\* \* \*

## Fortsetzung im nächsten Heft

# OLYMPIA

## für VC-20 Grundversion

Listing nächste Seite:

Die Weitsprung-Olympiade 1985 findet mit Ihrer Beteiligung statt. Tausende von Zuschauern haben sich eingefunden, um zu sehen, wie Sie evtl. einen neuen Weitsprung-Weltrekord aufstellen. Die Steuerung des Weitsprung-Sportlers geschieht über die Tastatur. Mit der RETURN-Taste können Sie auf einer langen Geraden Anlauf nehmen und in Höhe des Absprungbalkens müssen Sie mit Hilfe der SPACE-Taste abspringen. Wichtig ist dabei, daß Sie ein Übertreten und ein zu frühes Abspringen vermeiden. Je schneller Sie vor dem Sprung

anlaufen, um so weiter springen Sie. Sollte Ihnen ein neuer Rekord gelungen sein, erfolgt nach dem Sprung eine Siegerehrung. Also bereiten Sie sich gründlich auf Ihren Rekordversuch vor und beachten Sie olympische Sportregeln.







```

2406 G1$=" " : G2$=" "
2410 G3$=RIGHT$(G1$,32)
2412 G4$=" " : G5$="CONGRATULATION"
2416 G6$=RIGHT$(G4$,32)
2420 FORE=1T050:PRINTG1$:PRINTG2$:PRINTG3$
2422 FORE2=1T050:NEXTE2:PRINTG4$:PRINTG5$:PRINTG6$:FORE2=1T050:NEXTE2,E
2430 POKE36869,240:PRINT"DEINE SPRUNGWEITE: "HS
2432 PRINT"DAMIT HAST DU DEN REKORDSPRUNG VON: ":PRINTHS$:PRINT"UEBERBOTEN!"
2434 INPUT"DEIN NAME":HS$:GOTO308

```

## FLOWERS für VC-20 Grundversion

**Ziel des Spieles ist es, möglichst viele Blumen zu pflücken, ohne von einer Biene gestochen zu werden. Am Anfang erscheinen Bienen über verschiedenen Blumen. Diese Blumen dürfen anschließend nicht gepflückt werden.**

Das Spielfeld ist in zwei Teile aufgeteilt. Dabei muß man fünf Blumen je Reihe einsammeln, um in die nächste Reihe zu gelangen. Nach fünf gesammelten Blumen ist ein Brummtönen zu hören. Hat man die zweite Reihe geschafft, kommt man in eine neue Phase, bei der die Bienen am Anfang

immer kürzer auf dem Bildschirm erscheinen.

Um jetzt in eine neue Reihe zu kommen muß man ganz nach rechts gehen. Aber achten Sie darauf, daß Sie danach nicht mehr zurückgehen können.

Das Spiel selbst wird mit einem Joystick gesteuert (Feuertaste = Blumen, die direkt über Ihnen sind, können gepflückt werden). Man hat aber auch die Möglichkeit, mit der Tastatur (Z = links/rechts, SPACE = Blume pflücken) die Steuerung vorzunehmen.

### Das Programm besteht aus drei Teilen:

#### 1. Teil: Sonderzeichen

```

11 FORE=7424T07431:POKEE,0:NEXTE:POKE56,20:POKE36879,25
12 FORH=7168T07407:READA:POKEH,A:NEXTH:FORG=7448T07503:READA:POKEG,A:NEXTG
13 PRINT"LADE NUM TEIL 2":NEW
14 DATA0,0,28,60,60,126,254,254,0,0,0,0,0,24,60
16 DATA1,1,7,15,31,31,127,255,128,193,195,198,239,239,255,255
18 DATA131,135,143,159,255,255,255,255,195,231,255,255,255,255,255
20 DATA128,128,192,240,248,248,254,255,255,255,255,255,255,255,1,3,7,135,
,143,223
22 DATA 255,255,254,254,255,255,255,255,255,255,128,192,192,224,252,254,254,255
24 DATA0,0,0,0,0,0,255,24,60,126,126,60,24,60,255,28
26 DATA62,127,65,85,65,77,255,0,0,214,84,124,124,124,56
28 DATA16,16,20,92,124,112,16,16,148,249,255,255,255,255,255,255
30 DATA30,159,255,255,255,255,255,255,28,20,62,245,245,62,20,28
32 DATA0,60,60,255,94,126,62,28,24,90,90,90,90,90,66,60,54,54,54,54,38,238,238
34 DATA0,60,60,255,122,126,124,56,108,108,108,108,108,100,119,119
36 DATA0,3,3,15,5,7,3,1,1,5,5,5,5,5,4,3,3,7,143,254,252,248,112,32,0,192,192,240,
,224,224
38 DATA224,192,128,160,160,160,160,160,32,192,192,192,224,96,60,60,12,12
40 DATA0,3,3,15,7,7,7,3,3,3,7,6,62,60,48,48,0,192,192,240,160,224,192,128
42 DATA192,224,241,127,63,31,14,4
44 DATA0,6,31,63,31,11,1,0,0,198,255,255,255,255,157,0,0,64,224,248,252,248,32,0

```

# programme

## 2. Teil: Anleitung

```
99 POKE36879,25:POKE36878,15
100 A$=" DU WILLST BLUMEN VON EINEM ACKER KLAUEN.ABER PASS AUF,DASS DU KEINE BLU
ME MIT "
102 A$=A$+"BIENE NIMMST.SONST WIRST DU GESTOCHEN.DU MUSST MINDESTENS 5 BLUMEN JE
REIHE "
104 A$=A$+"NEHMEN,UM IN DIE NAECHSTE REIHE(BILD)ZU KOMMEN.VIEL SPASS MIT FLOWERS
!"
108 PRINT" ";D$=" "+A$
110 PRINT"LEFT$(A$,22)" "
112 FORW=1TO50:NEXTW:A$=RIGHT$(A$,(LEN(A$)-1)):IFLEN(A$)=0THENA$=D$
113 GETES$:IF-(PEEK(37151)AND32)=0)=10RES$=" "THEN120
114 GOTO110
120 PRINT" ";:POKE36869,255:CLR
122 F$=" >>> FLOWERS <<< " :M$=" " :H$=" " :L$=" "
124 J$=" " :FORW=1TO20
126 PRINT"LEFT$(F$,W)"M$:FORQ=1TO75:NEXT:PRINT"LEFT$(F$,W)"L$
127 PRINT"LEFT$(F$,W)"H$:FORQ=1TO75:NEXT:PRINT"LEFT$(F$,W)"J$
128 POKE36877,200:POKE36877,0:NEXT
130 F$=" WRITTEN BY: " :R$="T.SCHLAEGER;M.RITTER"
132 FORW=1TO20
134 PRINT"LEFT$(F$,W)"M$:FORQ=1TO75:NEXT:PRINT"LEFT$(F$,W)"L$
136 PRINT"LEFT$(F$,W)"H$:FORQ=1TO75:NEXT:PRINT"LEFT$(F$,W)"J$
```

# bücher

## VC 20

### Anwenderhandbuch John Heilborn & Ran Talbott

Das Anwenderhandbuch vermittelt alles nötige Wissen für den Umgang mit dem VC 20 und seinen Zusatzgeräten. Die VC 20-BASIC-Programmiersprache, der ganze Bereich der Colorgrafik und der Tonerzeugung und sogar Technik und Design eigener elektronischer Unterhaltungsspiele werden genau beschrieben. Das Buch eignet sich sowohl für den Anfänger zur schnellen Einführung, als auch für den erfahrenen Anwender als Nachschlagewerk. Computer Praxis (5/84) schreibt darüber: „... Uns hat dieses Buch ... durch sein immenses Informationsangebot und die erstklassige Aufmachung überzeugt.“

ISBN 3-89028-004-8. 1983  
388 Seiten, DM 32,-

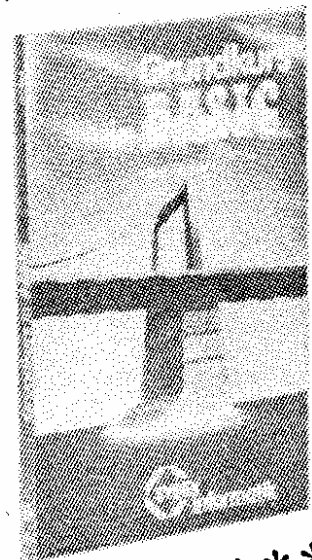
## Ulrich Ströbel Grundkurs in BASIC

ca. 260 Seiten / m. Abb. /  
Best.-Nr. 3058  
ISBN 3-88745-058-2 (1985)  
ca. DM 27,50 / sFr 25,70 /  
S 217,-

Diese Einführung in die meistgenutzte Programmiersprache BASIC richtet sich an Lehrer und Schüler in Informatik-AGs aller Schulformen; aber auch Hobby-Programmierer werden es mit Gewinn lesen. In kleinen Schritten werden die wichtigsten Fachbegriffe und Programmierbefehle dargestellt. Detailliert und umfassend lernt der Anfänger, schon mit wenigen Begriffen recht eindrucksvolle Programme in BASIC zu erstellen. Eine Fülle praktischer Beispiele und Übungsaufgaben begleitet den Leser auf seinem Weg zum fortgeschrittenen BASIC-Programmierer.

Sybex Verlag, Düsseldorf

\*\*\*\*\*



\*\*\*\*\*



**Wenn Sie keine Lust  
haben, alle  
Programme  
einzutippen, ...  
... wir helfen  
Ihnen!**



Sie erhalten von uns innerhalb **1 Woche** den kompletten Datenträger mit allen Programmen dieser Ausgabe für **Commodore** (beinhaltet VC-64 + VC-20 + C16) und für **Schneider (CPC-464)**

**Preis je Kassette DM 16,-**  
**Preis je Diskette DM 20,-**  
(Diskette nur Commodore)

**Bitte nebenstehenden Bestellcoupon ausfüllen und einsenden an:  
Tronic-Verlag, Postfach 41, 3444 Wehretal 1**

**Bitte beachten Sie:**  
Sie ersparen sich zusätzliche Kosten (bis zu DM 5,-), wenn Sie per Vorkasse (bar, Verrechnungsscheck) bestellen. Ausland: ● Bestellungen nur gegen Vorkasse!

**Bestellungen per Telefon:**  
Wenn es schnell gehen soll ... rufen Sie uns an. Wir nehmen Ihre Bestellung gern entgegen.



**Tel.-Nr.: 0 56 51 / 4 06 93 oder 4 06 43**



**Aus diesem Heft:**

**Commodore Kasette**  
**Bestell-Nr. COM C 1/1 16,- DM**  
**Bestell-Nr. COM D 1/1 20,- DM**

**Schneider Kasette**  
**Bestell-Nr. COM S 1/1 16,- DM**

**Commodore 64**

Mauern/Widerstand, Kasette 8,- DM, Diskette 15,- DM, Bestell-Nr. C 41

Space-Comets/Erdspalte/Sprite-Data, Kasette 15,- DM, Diskette 23,50 DM, Bestell-Nr. C 51

Autostart/Bestellschein/Roadpainter, Kasette 16,50 DM, Diskette 23,50 DM, Bestell-Nr. C 61

Hardcopy/Space-Fighter/Data-Generator, Kasette 15,50 DM, Diskette 19,50 DM, Bestell-Nr. C 71

Monster-Attack/Block-Painter/Epson-Drucker, Kasette 16,50 DM, Diskette 23,50 DM, Bestell-Nr. C 81

Projekt/Datenbank, Kasette 16,- DM, Diskette 23,50 DM, Bestell-Nr. C 91

Spiders/The Basic, Kasette 16,50 DM, Diskette 23,50 DM, Bestell-Nr. C 101

Multi-Key/S-Tool 64/Interrupt-Programme, Kasette 16,- DM, Diskette 21,- DM, Bestell-Nr. UC 2/85

Spritehilfe/Diskloader/Directory, Diskette 21,- DM, Bestell-Nr. UC 3/85

Tape-Directory/Asmon/Data-Generator, Kasette 15,- DM, Diskette 21,- DM, Bestell-Nr. UC 3/85

Fast-Load, Kasette 10,- DM, Bestell-Nr. 4/85

Diskmonitor/Zeichensatz, Diskette 20,- DM, Bestell-Nr. UC 5/85

Wüstenrallye/Jet-Pac/Black Moore Castle, Kasette 17,50 DM, Diskette 23,50 DM, Bestell-Nr. C 32

High Noon/Skeet/Grafik-Designer, Kasette 17,50 DM, Diskette 23,50 DM, Bestell-Nr. C 121

Painter/Star-Baddle/Editor, Kasette 17,50 DM, Diskette 23,50 DM, Bestell-Nr. C 22

**Schneider CPC-464**

Autorennen, Kasette 11,- DM, Bestell-Nr. US 3/85

Universal-Datari V1, Kasette 9,- DM, Bestell-Nr. US 4/85

Computerschrift/Symbol-Swap, Kasette 10,- DM, Bestell-Nr. US 5/85

**VC-20**

Bestellschein/Glücky, Kasette 8,50 DM, Diskette 15,- DM, Bestell-Nr. V 61

Multigraph/All Rammer, Kasette 11,- DM, Diskette 15,50 DM, Bestell-Nr. V 71

Zyklus/Meteorit, Kasette 11,- DM, Diskette 19,50 DM, Bestell-Nr. V 81

Garten/Schloß Gruselstein, Kasette 14,- DM, Diskette 19,50 DM, Bestell-Nr. V 91

Fressman/Outlaw, Kasette 14,- DM, Diskette 19,50 DM, Bestell-Nr. V 101

Prost/Bufalo Bill, Kasette 14,- DM, Diskette 19,50 DM, Bestell-Nr. V 121

Joy Man/Powerpack, Kasette 14,- DM, Diskette 19,50 DM, Bestell-Nr. V 22

Der rasende Malocher/Frankie goe to Pharao, Kasette 14,- DM, Diskette 19,50 DM, Bestell-Nr. V 32

Grafik-Painter, Kasette 10,- DM, Bestell-Nr. UV 5/85

**Bestellcoupon**

**"Compute mit" Software-Service \* "Compute mit" Software-Service \* "Compute mit" Software-Service**

Die Zustellung erfolgt: gegen **Vorkasse**   
 Ausland: nur gegen Vorkasse

oder Inland per **Nachnahme**   
 + Versandkosten

innerhalb von 1 Woche

Entnehmen Sie bitte aus unserer Preisliste die notwendigen Angaben für Ihre Bestellung:

Bitte liefern Sie mir:

Kasette für

Bestell-Nr. \_\_\_\_\_

Anzahl

Diskette für

Bestell-Nr. \_\_\_\_\_

Anzahl

zum Preis von gesamt

\_\_\_\_\_ DM

Name/Vorname: \_\_\_\_\_

Straße, Nr.: \_\_\_\_\_

PLZ/Ort: \_\_\_\_\_

Datum, Unterschrift \_\_\_\_\_

# Schneider Personal Computer „CPC464“

Die Zeit des Experimentierens ist jetzt endgültig vorbei. Mit dem Schneider Colour Personal Computer „CPC464“ verfügen Sie über ausgereifte Computertechnik in seiner reinsten Form. Das beste Beispiel für höchste Technologie in Farbe, kom-

promiöse Perfektion in Bedienungskomfort und Ausstattung. Durch sein attraktives Preis-/Leistungsverhältnis ist der „CPC464“ anderen Computern dieser Klasse weit überlegen, denn der „CPC464“ ist der einzige mit einer Schreibma-

schinentastatur und 64 KB RAM (Arbeitsspeicher) und 32 KB ROM (Systemspeicher) und einem eingebauten Cassettenrecorder (Data-corder) und einem Bildschirm-System und einem sehr schnellen Standard-Basic.

## Schneider „CPC464“ – der fortschrittliche Computer für die Familie

Der „CPC464“ ist sicherlich sehr viel mehr als nur ein Gerät für Telespiele. Mit hochauflösender Graphik, 80 Zeichen/Zeile Bildschirmanzeige zusammen mit zweckentsprechenden Bildschirmsystemen, ist der „CPC464“ der preisgünstigste Computer mit all den Eigenschaften zum „Einsteigen und Anfahren“. Die große Auswahl attraktiver Programme für die ganze Familie, vom Computerspiel bis hin zu kaufmännischen Lösungen, tragen zu seiner Vielseitigkeit bei.

Geübte Benutzer werden sehr schnell die Erweiterungsmöglichkeiten zu schätzen wissen:

- die eingebaute parallele Drucker-Schnittstelle,
  - das wahlweise anzuschließende Diskettenlaufwerk, das sowohl CP/M\* (damit können mehr als 3000 bewährte Programme verwendet werden) als auch die Programmiersprache LOGO (bekannteste Sprache für Lehr- und Lernprogramme in den USA) beinhaltet.
  - die nahezu unbegrenzten Möglichkeiten des Z80-Daten-Bus, mit dem erweiterten ROM-Anschluß.
- Computer-Neulinge brauchen keine Angst davor zu haben, den „CPC464“ zu benutzen: Der Computer wird ausgeliefert mit einer Einfüh-

rungscassette, die Sie mit dem Computer und seinen Möglichkeiten vertraut macht. Das Benutzerhandbuch führt Anfänger Schritt für Schritt hin zum problemlosen Gebrauch und gibt detaillierte Auskünfte über die Hard- und Software-Eigenschaften des „CPC464“.

Ein bedienungsfreundlicher Joystick (JY-1) ist zusätzlich für den „CPC464“ erhältlich. Der Griff liegt gut in der Hand und enthält einen schnell reagierenden Feuerknopf. Die Standsicherheit gewährleisten seine Saugfüße. Das Kabel zum Anschluß an die Tastatur ist 150 cm lang. Über eine eingebaute Buchse ist der Anschluß eines zweiten Joystick möglich. Die optionale Stromversorgung MP-1 ermöglicht den Anschluß des „CPC464“ an jedes moderne Farbfernsehgerät. Besonders nützlich für alle, die den grünen monochromen Bildschirm gewählt haben und die Graphik in Farbe darstellen wollen. Das MP-1 ist handlich und paßt im Stil zum „CPC464“.

Das „CPC464“-System bietet nahezu unbeschränkte Anschlußmöglichkeiten:

- das Floppy-Disk-System für schnellen Zugriff auf große Datenmengen
- Verstärker und Lautsprecher für beste Stereowiedergabe

- Drucker zum Schreiben von Briefen
- Ausdruck von graphischen Darstellungen, Programmlisten, Rechnungen usw.
- Datenkommunikation mit anderen Computern über Modems
- ROM-Erweiterung für zusätzliche Hard- und Software und weitere Anwendungsmöglichkeiten.

### Die Vorteile des „CPC464“ liegen in der Kombination von leistungsfähiger Hardware und benutzerfreundlicher Software

**Z80 CPU** – der weltweit meistverwendete Mikroprozessor in der Homecomputerklasse, mit dem ein Großteil der vorhandenen Software genutzt werden kann, bietet vor allem die Möglichkeit, das Betriebssystem CP/M\* zu benutzen.

**64 K RAM** – der „CPC464“ hat standardmäßig 64 K RAM, von denen über 42 K tatsächlich dem Benutzer zur Verfügung stehen. Dies wird erreicht durch die ROM Overlay-Technik. Der zur Verfügung stehende Speicherplatz ermöglicht es, auch größere Programme zu entwickeln.

**Graphik** – der „CPC464“ unterstützt 3 Darstellungsarten, 80spaltigen Text, eine Palette von 27 Farben und eine Auflösung bis zu 640 x 200 Bildpunkten.

**Tastatur** – eine Schreibmaschinentastatur mit 74 farblich abgesetzten Tasten und besonders großer „ENTER“ (Eingabe)-Taste. Die Cursor-Kontrolltasten sind funktionell angeordnet, und ein eigener Ziffernblock ermöglicht bequeme und schnelle Datenerfassung.

**Eingebauter Daten-Recorder** – Cassettenspeicher steht mit dem eingebauten Daten-Recorder zur Verfügung. Damit können Sie Ihre Programme einlesen und abspeichern, ohne sich mit den leidigen Problemen der Kompatibilität, des Anschließens und des richtigen Pegels abgeben zu müssen.

Programme können entweder mit 1K Baud oder 2K Baud (Schnell-Laden) geschrieben werden. Die Lesegeschwindigkeit wird automatisch softwaremäßig eingestellt.

**BASIC** – entspricht dem Industriestandard.

Sie werden sehen, daß es schneller und vielseitiger ist, als Sie es von BASIC erwarten, mit vielen Erweiterungen für Graphik und Musik.

**Erweiterter Zeichensatz** – voller 8-bit Zeichensatz mit Graphik und Symbolen, der über die CHR\$(n)-Funktion, also über die Tastatur, zugänglich ist.

**Funktionstasten** – bis zu 32 Tasten können beim „CPC464“ mit bis zu 32 Zeichen pro Taste belegt werden.

**Fenster** – bis zu acht Fenster für Text und 1 Fenster für Graphik können definiert werden.

**Ton** – der „CPC464“ bietet Musik aus 3 Kanälen (Stimmen) über 8 Oktaven. Für jeden dieser Kanäle können unabhängig Tonhöhe und Lautstärke gewählt werden. Sie stehen als linker, mittlerer und rechter Kanal an der Stereo-Ausgangsbuchse zur Verfügung. Der eingebaute Lautsprecher gibt die Musik in Mono wieder. Die Lautstärke ist dabei regelbar.

**Druckeranschluß** – ein Centronicskompatibler Drucker-Anschluß nach Industriestandard ist vorhanden. Die „Busy“-Signalleitung wird verwendet für „Handshake“ (gegenseitige Übertragung).

**Systemerweiterung** – Viele Hardware-Schnittstellen sind vorhanden, die zusätzliche Software-Erweiterungen ermöglichen.

**ROM-Erweiterung** – jedes ROM belegt die oberen 16 K des Arbeitsspeichers. Über die Firmware können bis zu 240 ROM-Erweiterungen zu je 16 K angesprochen werden.

### Der lesefreundliche Bildschirm

Der grüne monochrome Bildschirm ist ideal für Aufgaben aus dem kaufmännischen Bereich, für Textverarbeitung, Finanzplanung, Kassenführung und Programmentwicklung mit dem „CPC464“.

Text und Zahlen sind scharf und klar. Vergleichen Sie, wieviel Sie mit einem Blick auf die 80spaltige Anzeige des „GT 64“ erfassen können und

wie wenig bei einer 40spaltigen Anzeige üblicher Home Computer. Betrachten Sie dies einmal anhand einer Programmliste.

Genau wie beim „CTM 640“ ist beim „GT 64“ die Stromversorgung für das Gesamtsystem eingebaut. Zusammen mit den hervorragenden Eigenschaften des „CPC464“ haben Sie damit ein System, das Ihnen außergewöhnlich hochwertige Computertechnik und vielseitige Anwendungsmöglichkeiten bietet, wie sie vergleichsweise nur in wesentlich höheren Preisklassen angeboten werden.

### Sinnvoll und praktisch

Wenn Sie den „CPC464“ mit dem „GT 64“ benutzen, können Sie mit Hilfe der optionalen „MP-1“-Stromversorgung auch Ihr normales Farbfernsehgerät anschließen. Alle Farben des „CPC464“ können dann ohne jegliche Änderungen dargestellt werden.

## Für Spitzenleistungen in jedem Aufgabenbereich:

### Der Schneider „CPC464“ – der vollausgestattete Computer ...

- Komplettes ausgestattetes System einschließlich RGB Farbbildschirm oder grünem Monochrom-Schirm
- Eingebauter „Schnell-Lade“ Daten-Recorder
- 64 K RAM Arbeitsspeicher
- „Echte“ Schreibmaschinentastatur mit 74 Tasten, abgesetztem Ziffernblock, frei definierbaren Funktionstasten und einstellbarem Daueranschlag
- 3 Anzeigearten, 27 Farben, frei definierbare Blinkfrequenz
- Hohe Bildschirmauflösung für 80spaltigen Text, 640 x 200 adressierbare Bildpunkte für Graphik
- und bis zu acht Textfenster
- Joystickanschluß für zwei Geräte
- Centronicskompatibler Anschluß für Parallel-Drucker
- Ein kompletter Erweiterungsanschluß für ROM, serielle Schnittstellen, Disketten-Laufwerke und Modem
- 3stimmiger Stereo-Ausgang über 8 Oktaven mit Hüllkurven für Lautstärke und Klang
- Umfassendes Programmangebot für kaufmännische Bereiche, Lehr- und Lernprogramme und die beliebtesten Computerspiele

## Karl-Hermann Rollke Grundkurs in Pascal

224 Seiten / Band 1 /  
Best.-Nr. 3046  
ISBN 3-88745-046-9 (1984)  
DM 24,80 / sFr 23,- / S 193,-

Der sichere Einstieg in Pascal für den Informatik-Unterricht in der Sekundarstufe II, für Informatik-Kurse an der Volkshochschule, für Kurse und als Sekundärliteratur an der Hochschule, für Hobby-Programmierer und zum Selbststudium – von einem Lehrer aus der Praxis geschrieben. In einer leicht verständlichen Sprache werden Sie mit einer vielgenutzten Programmiersprache vertraut gemacht. Sie machen Ihre ersten Schritte im UCSD-Pascal-System, lernen den Aufbau eines Pascal-Programms, reservierte Wörter, Syntaxdiagramme, Prozeduren, Funktionen, Rekursionen und Sortieralgorithmen kennen. Ihre neuen Kenntnisse können Sie direkt an vielen Übungen und Beispielprogrammen erproben, die auf einem Apple-Rechner getestet wurden und leicht auf andere Systeme zu übertragen sind.



## Karl-Hermann Rollke Grundkurs in Pascal

224 Seiten / mit Abb. / Band 2 /  
Best.-Nr. 3061  
ISBN 3-88745-061-2 (1985)  
DM 24,80 / sFr 23,- / S 193,-

Lehrer und Schüler, Studenten und Autodidakten runden mit diesem Band ihren Einstieg in die leistungsfähige Programmiersprache Pascal ab. Der zweite Teil des Grundkurses behandelt als ein Hauptthema die Datenstrukturen: dazu gehören Verbünde, Mengen, Dateien und Zeiger.

Weitere Schwerpunkte: Das Bibliothekskonzept in UCSD-Pascal; Einführung in die Programmierung auf Maschinenebene; Vorschläge für Projekte. Viele Übungen und Beispiele, die auf einem Apple getestet wurden, helfen Ihnen, Ihr Wissen direkt zu erproben und anzuwenden.

Sybex Verlag, Düsseldorf

# COBRA

## Programm für Schneider CPC-464

Ziel des Spiels ist es, eine möglichst große Punktezahl zu erreichen. Dies geschieht dadurch, daß man seine Schlange jede Menge Mäuse verzehren läßt. Gesteuert wird die Schlange mit einem Joystick oder den Cursortasten.

Die Schlange darf natürlich nicht in eine Mauer oder in sich selbst beißen, denn das wäre ihr Ende. Ebenso darf sie keinen Igel fressen – der würde ihr im Hals stecken bleiben.

Während des Spiels werden angezeigt:

Am oberen Rand Score, Bonus und Highscore, im unteren Rand die verbliebenen Leben.

Für jede verzehrte Maus gibt es 10 Punkte. Nach jeweils 350 Punkten gibt es einen Bonus, nach jeweils 500 Punkten ein neues Leben und einen weiteren Bonus. Nach jeweils 1000 Punkten beginnt eine neue Stage.

Die Farbe des Spielfelds ändert sich und es werden 2 zusätzliche Hindernismauern aufgestellt. Außerdem gibt es einen weiteren Bonus. Insgesamt gibt es 4 solcher Stages. Aber es ist

schon schwierig, über 3000 Punkte zu erreichen und noch ein Leben übrig zu haben. Das erfordert viel Geschick und Übung.

Der letzte Bonus ist ein Zeitbonus, den es während der beiden letzten Leben gibt. Hierbei erhält man alle 10 Sekunden 1-15 Punkte (RND-Funktion).

Übrigens wird jeder Bonus durch die RND-Funktion ermittelt.

Nach Spielende wird der Bonus zum Score hinzugezählt. Die Hauptschwierigkeit des Spiels besteht darin, daß die Schlange jedesmal, wenn



sie eine Maus verspeist hat, ein Stück wächst. Da nicht alle Mäuse den gleichen Nährwert haben, wächst die Schlange auch nicht immer gleich viel. Die Zahl der neuen Körpersegmente liegt zwischen 1 und 7 (RND).

## Speicherplatzbedarf

8 394 Bytes (Listing)  
40 891 Bytes (nach Durchlauf  
[DIM ...])

Hier noch ein kleiner Tip:  
Vielleicht sollte man sich für eine Art der Steuerung entscheiden (nur Joystick oder nur Tastatur). Man kann dann einen Teil der Abfrage streichen. Die Schlange wird dadurch etwas schneller. Will man z. B. nur mit Joystick steuern, dann können in den Zeilen 1210 und 1220 die INKEY-Abfragen gestrichen werden (OR INKEY[n] = 0).  
Die Abfrage muß dann lauten:  
1210 IF jy = 4 THEN ...

## Die wichtigsten Variablen:

co = Farbe Schlange  
ink3 = Farbe Mäuse, Igel, Mauern, Schrift  
bo = Zähler Bonus  
bx = Zähler neues Leben  
sc = Score  
hsc = Highscore  
zc = Bonuspunkte  
le = Anzahl Leben  
le(pl) = Anzahl Schlangensegmente  
by = Zeichencode [CHR\$(by)]  
hin\$ = Hindernismauern  
cob\$ = Schrift „COBRA“

Die Eingabe der Listings kann erleichtert werden, indem die automatische Zeilennummerierung eingesetzt wird. Hierzu muß nur der Befehl „AUTO“ direkt eingegeben werden.

Die Zeilen sind fortlaufend in Zehnerschritten durchnummeriert.  
Als weitere Erleichterung kann folgendes Programm verwendet werden:

```
10 KEY DEF 17,1,40
20 KEY DEF 19,1,41
30 KEY DEF 26,1,35
40 KEY DEF 68,1,36
50 KEY DEF 22,1,63
60 KEY DEF 16,1,34
```

Das Programm wird mit „RUN“ gestartet und kann nach Durchlauf mit „DELETE 10-60“ gelöscht werden. Jetzt können Zeichen, die normalerweise mit SHIFT eingegeben werden müssen, ohne SHIFT eingegeben werden. Nachfolgend eine Tabelle der neuen Belegung:

Taste:	Belegung
[	(
]	)
■?	#
TAB	\$
\	? (PRINT)
CLR	"

```
10 REM *****
20 REM *
30 REM * C O B R A *
40 REM *
50 REM * (c)1985 by A.Hess *
60 REM *
70 REM *****
80 GOTO 110
90 IF le<=2 THEN zc=zc+INT(RND*15)+1
100 RETURN
110 EVERY 500 GOSUB 90
120 hin$=STRING$(5,248):stage=1
130 DEFINT a-z:MODE 1:ink3=24:INK 0,0:INK 1,9:INK 2,15:INK 3,ink3:BORDER 0
140 SYMBOL AFTER 47
150 SYMBOL 48,126,66,66,98,98,98,98,126
160 SYMBOL 49,24,24,24,56,56,56,56,56
170 SYMBOL 50,126,66,2,126,96,96,98,126
180 SYMBOL 51,126,2,2,126,6,6,6,126
190 SYMBOL 52,64,64,96,104,104,126,12,12
200 SYMBOL 53,126,64,64,126,6,102,102,126
210 SYMBOL 54,126,64,64,126,98,98,98,126
220 SYMBOL 55,62,34,2,6,6,6,6,6
230 SYMBOL 56,126,66,66,126,98,98,98,126
240 SYMBOL 57,126,66,66,126,6,6,6,126
250 SYMBOL 130,126,70,70,64,94,94,66,126
260 SYMBOL 131,99,119,95,75,67,67,91,91
270 SYMBOL 132,126,66,66,70,70,70,70,126
280 SYMBOL 133,97,97,97,97,99,54,28,8
290 SYMBOL 134,124,68,68,126,102,102,102,102
300 SYMBOL 135,66,66,66,127,99,99,99,99
310 SYMBOL 136,32,32,32,48,48,48,50,62
320 SYMBOL 137,124,68,68,124,70,70,70,126
330 SYMBOL 241,0,60,110,118,110,118,60,0
340 SYMBOL 242,60,60,90,219,153,255,126,24
350 SYMBOL 243,56,108,99,255,255,99,108,56
```

# programme

```
360 SYMBOL 244,24,126,255,153,219,90,60,60
370 SYMBOL 245,28,54,198,255,255,198,54,28
380 SYMBOL 246,1,2,4,50,49,63,95,255
390 SYMBOL 247,0,20,93,254,127,255,126,85
400 SYMBOL 248,255,161,161,255,137,137,137,255
410 SYMBOL 249,126,66,64,126,2,98,98,126
420 SYMBOL 251,62,34,34,62,98,98,98,98
430 SYMBOL 253,124,64,64,124,96,96,96,124
440 SYMBOL 254,124,76,76,64,64,68,68,124
450 ENT 2,5,0,1,5,-1,1,1,-1,1,2,-2,1,2,-10,1
460 ENT 3,40,-3,2
470 ENT 1,25,5,4,15,-7,2
480 FOR i=1 TO 4:READ xr(i),yr(i):NEXT i
490 '
500 PAPER 0:CLS:PEN 1:PRINT CHR$(254);CHR$(132);CHR$(137);CHR$(134);CHR$(251):PE
N 3
510 FOR x=0 TO 39
520 FOR y=-1 TO 8
530 LOCATE x+1,y+5:IF TEST(x*2,399-y*2)=1 THEN PRINT CHR$(248); ELSE PRINT " ";
540 NEXT y,x
550 by=241:PEN 3
560 LOCATE 7,15:PRINT"Versuchen Sie moeglichst":LOCATE 7,16:PRINT"viele Mae
use zu fressen!": LOCATE 7,17:PRINT"Gehen Sie d
abei den IgeIn":LOCATE 7,18:PRINT"aus dem Weg!"
570 LOCATE 7,20:PRINT"Starten Sie das Spiel mit"
580 LOCATE 7,21:PRINT"dem";:PEN 1:PRINT" Feuerknopf ";:PEN 3
590 LOCATE 7,22:PRINT"oder mit der 'COPY'-Taste !"
600 jr=2:x=1:y=1:coj=2:k=0:bo=0:stage=1
610 x1=x:y1=y:by=241:co=coj:GOSUB 1270:coj=coj+1:IF coj=3 THEN coj=1
620 x=x+xr(jr):y=y+yr(jr):IF x>k AND y>k AND x<41-k AND y<26-k THEN 650
630 x=x-xr(jr):y=y-yr(jr):jr=jr+1:IF jr>4 THEN jr=jr-4
640 IF jr=2 THEN k=k+1:y=y+1:IF k=13 THEN 490
650 x1=x:y1=y:by=241+jr:co=3:GOSUB 1270
660 IF f1=1 THEN f1=0:INK 2,9:INK 1,15 ELSE f1=1:INK 2,15:INK 1,9
670 IF JOY(0)<16 AND INKEY(9)<>>0 THEN FOR n=1 TO 100:NEXT:GOTO 610
680 '
690 DIM x(1),y(1)
700 as=1:le=5:bx=0:bo=0:zc=0:ink3=24
710 sc=0
720 bx=0
730 IF le=0 THEN sc=sc+zc:le=0:GOSUB 1430:GOSUB 1360:ERASE x,y:bo=0:zc=0:GOTO 6
90
740 ERASE x,y:DIM x(1,4000),y(1,4000):PAPER 0:PEN 3:GOSUB 1590:CLS
750 xp(1)=9:yp(1)=10:jr(1)=2:l(1)=20:p(1)=1+1(1)
760 xp(0)=30:yp(0)=20:jr(0)=4:l(0)=20:p(0)=1(0)+1
770 coj=1:co=3
780 FOR x1=1 TO 40
790 y1=2:by=248:GOSUB 1270:y1=25:GOSUB 1270
800 NEXT x1
810 IF sc>=1000 THEN GOSUB 1750
820 IF sc>=2000 THEN GOSUB 1790
830 IF sc>=3000 THEN GOSUB 1820
840 IF sc>=4000 THEN GOSUB 1850
850 FOR y1=2 TO 25
860 x1=1:GOSUB 1270:x1=40:GOSUB 1270
870 NEXT y1
880 FOR i=1 TO 10
890 GOSUB 1300:IF i/2=INT(i/2) THEN by=247 ELSE by=246
900 GOSUB 1270
910 NEXT i
920 GOSUB 1360
930 MOVE 0,383:DRAW 639,383,0:DRAW 639,381,0:DRAW 0,381,0:tf=0
940 '
950 FOR pl=0 TO as-1
960 SOUND 2,240,10,13,0,2,1
970 p=p(pl):x=xp(pl):y=yp(pl):jr=jr(pl):l=l(pl):x=x+xr(jr):y=y+yr(jr)
980 x2=x*16-15:y2=414-y*16:c1=TEST(x2+12,y2-6):c2=TEST(x2+8,y2-5)
990 IF c1=1 OR c1=2 THEN 1070
1000 IF c2=3 THEN 1070
```

```

1010 IF c1<>3 THEN 1140
1020 SOUND 2,100,50,15,,3
1030 l(p1)=l(p1)+INT(RND*7)+1
1040 GOSUB 1300:by=246:co=3:GOSUB 1270
1050 GOSUB 1300:by=247:GOSUB 1270
1060 sc=sc+10:bo=bo+1:bx=bx+1:GOSUB 1360:GOTO 1140
1070
1080 le=le-1:tf=1:by=241+jr:x1=x:y1=y:co=2
1090 FOR i=1 TO 5
1100 SOUND 2,90+i*i*10,42,15,,1
1110 GOSUB 1270:co=co+1:IF co=4 THEN co=2
1120 IF(SQ(2) AND 128)=128 THEN 1120
1130 NEXT i
1140 x1=x(p1,p-1):IF x1<>0 THEN y1=y(p1,p-1):co=coj:by=240-(p1=0):GOSUB 1270
1150 x1=x:y1=y:co=3:by=241+jr:GOSUB 1270
1160 x(p1,p)=x:y(p1,p)=y:xp(p1)=x:yp(p1)=y
1170 IF p-1>0 THEN x1=x(p1,p-1):IF x1<>0 THEN y1=y(p1,p-1):co=0:by=241:GOSUB 1270
1180 p(p1)=p(p1)+1:e#=INKEY#
1190 FOR p11=0 TO as-1
1200 jy=JOY(0)
1210 IF jy=4 OR INKEY(8)=0 THEN jr(p11)=4 ELSE IF jy=8 OR INKEY(1)=0 THEN jr(p11)=2
1220 IF jy=1 OR INKEY(0)=0 THEN jr(p11)=1 ELSE IF jy=2 OR INKEY(2)=0 THEN jr(p11)=3
1230 NEXT p11,p1
1240 IF coj=1 THEN INK 1,9:INK 2,15:coj=2 ELSE INK 1,15:INK 2,9:coj=1
1250 IF tf=1 THEN 730
1260 GOTO 940
1270
1280 INK 3,ink3:LOCATE x1,y1:PEN co:PRINT CHR$(by);
1290 RETURN
1300
1310 x1=INT(RND(2)*40+1):y1=INT(RND(3)*24+2)
1320 x2=x1*16-15:y2=414-y1*16
1330 c1=TEST(x2+12,y2-6):c2=TEST(x2+6,y2-6)
1340 IF c1<>0 OR c2<>0 THEN GOSUB 1310
1350 RETURN
1360 IF bx=35 THEN FOR n=1 TO 10:SOUND 5,50*n,6,15:NEXT:zc=zc+INT(RND*30)+10:bx=0
1370 IF bo=50 THEN le=le+1:zc=zc+INT(RND*80)+10:bo=0:GOSUB 1670
1380 IF sc>hsc THEN hsc=sc
1390 LOCATE 1,1:PRINT CHR$(249);CHR$(254);":":LOCATE 12,1:PRINT CHR$(137);CHR$(132);":":LOCATE 24,1:PRINT CHR$(135);CHR$(249);CHR$(254);":":LOCATE 16,25:PRINT CHR$(144);CHR$(136);CHR$(253);":":STRING$(le,244);CHR$(144);
1400 LOCATE 4,1:PRINT USING"#####";sc:LOCATE 15,1:PRINT USING"#####";zc:LOCATE 28,1:PRINT USING"#####";hsc
1410 LOCATE 36,1:PRINT CHR$(254);CHR$(132);CHR$(137);CHR$(134);CHR$(251):PEN 3
1420 RETURN
1430 GOSUB 1360
1440 PAPER 2:INK 3,26,0:SPEED INK 10,4:PEN 3
1450 LOCATE 10,11:PRINT SPACE$(15);:LOCATE 10,12:PRINT SPACE$(15);
1460 LOCATE 10,13:PRINT SPACE$(15);:LOCATE 10,12:t$=" * "+CHR$(130)+CHR$(251)+CHR$(131)+CHR$(253)+" "+CHR$(132)+CHR$(133)+CHR$(253)+CHR$(134)+" * "
1470 FOR i=1 TO LEN(t$)
1480 PRINT MID$(t$,i,1);:SOUND 2,400,10,15,,1
1490 IF (SQ(2) AND 128)=128 THEN 1490
1500 NEXT i
1510 INK 3,24:PAPER 0:GOSUB 1380:PAPER 0:LOCATE 3,25:PRINT" Noch ein Spiel (J/N)
?"
1520 i#=INKEY#:IF i#="J" OR i#="j" THEN CLS:GOTO 700
1530 IF i#="N" OR i#="n" THEN 1880
1540 GOTO 1520
1550
1560 PAPER 0
1570 ink3=24
1580 RETURN

```

# programme

```
1590
1600 PEN 3:PAPER 0
1610 LOCATE 1,25:PRINT SPACE$(40);
1620 LOCATE 1,25:PEN 3
1630 PRINT" Druucken Sie den Feuerknopf oder 'COPY'";
1640 IF JOY(0)<16 AND INKEY(9)<>0 THEN 1640
1650 RETURN
1660 DATA 0,-1,1,0,0,1,-1,0
1670 FOR n=1 TO 10
1680 FOR t=20 TO 300 STEP 50
1690 SOUND 5,t,1,15
1700 NEXT t
1710 FOR t=300 TO 20 STEP -50
1720 SOUND 5,t,1,15
1730 NEXT t,n
1740 RETURN
1750 PEN 3:LOCATE 2,9:PRINT hin$:LOCATE 35,11:PRINT hin$:zc=zc+INT(RND*30)+1
1760 ink3=15
1770 PEN 1
1780 RETURN
1790 GOSUB 1750:PEN 3:LOCATE 6,6:PRINT hin$:LOCATE 18,15:PRINT hin$:PEN 1:zc=zc+
INT(RND*50)+1
1800 ink3=2
1810 RETURN
1820 GOSUB 1790:PEN 3:LOCATE 5,20:PRINT hin$:LOCATE 24,22:PRINT hin$:PEN 1:zc=zc
+INT(RND*70)+1
1830 ink3=6
1840 RETURN
1850 GOSUB 1820:PEN 3:LOCATE 8,22:PRINT hin$:LOCATE 17,4:PRINT hin$:PEN 1:zc=zc+
INT(RND*100)+1
1860 ink3=18
1870 RETURN
1880 PAPER 0:INK 3,24:INK 2,6,24:SPEED INK 10,4:CLS
1890 PEN 3:LOCATE 1,4:PRINT"Das Spiel kann mit ";PEN 2:PRINT" 'GOTO 700'";PEN
3:PRINT" wieder":PRINT:PRINT"gestartet werden, o
hne dass der High-":PRINT:PRINT"score verloren geht !"
1900 LOCATE 1,23:END
```

## Krümi:

Schneider CPC-464

### Bei Krümi handelt es sich um eine Variation des bekannten Pacman-Spiels

Krümi ist ein äußerst hungriges Männchen, das dauernd gelbe und rote Kekse fressen muß. Fährt es durch ein leeres Feld, so muß es von seiner Notration, den „Vespern“ zehren. Frißt Krümi einen roten Keks, blinkt in der Anzeige ein Feld und aus dem Lautsprecher ertönt das „Jagdzeichen“. Solange das Feld blinkt und das Jagdzeichen ertönt,

kann Krümi auch die Geister fressen, die normalerweise eine tödliche Gefahr für ihn darstellen.

Hat Krümi ein Labyrinth leergefressen oder über 500 Punkte erreicht, so erscheint ein neues Labyrinth. Doch Vorsicht! In diesem Labyrinth sind auch hellblaue Felder, die die Geister passieren können, Krümi jedoch nicht.

Gesteuert wird Krümi mit den Cursorstasten oder mit Joystick.

#### Punkteverteilung:

Keks	: 2 Punkte
Geist gefressen	: 80 Punkte
	+ 40 Vesper
leeres Labyrinth	: 300 Punkte
	+ Vesper
verlorenes Leben	: - 50 Punkte
leeres Feld	: - 1 Vesper

```

*           *           *           *           *
1 ' *** von           : Markus Buehler
2 ' *** tel           : 0781/33778
3 ' *** Adresse       : im Erb 29       7600 Offenburg/Fessenbach
4 SYMBOL 245,60,126,231,230,252,254,127,62
5 SYMBOL 246,60,126,231,231,252,255,127,62
6 SYMBOL 247,60,126,231,103,63,127,254,124
7 SYMBOL 248,60,126,231,231,63,255,254,124
8 g=34:FOR i=249 TO 250:SYMBOL i,g,119,255,207,207,255,126,60:g=54:NEXT
9 g=34:FOR i=251 TO 252:SYMBOL i,60,126,255,207,207,255,119,g:g=54:NEXT
10 SYMBOL 253,60,90,60,24,60,60,126,255:geist#=CHR$(253)
11 DIM kr$(1,4),x(2),y(2):DEFINT a-z
12 kr$(0,1)=CHR$(251): kr$(1,1)=CHR$(252)
13 kr$(0,2)=CHR$(249): kr$(1,2)=CHR$(250)
14 kr$(0,3)=CHR$(247): kr$(1,3)=CHR$(248)
15 kr$(0,4)=CHR$(245) : kr$(1,4)=CHR$(246)
16 ENT -1,1,20,12,1,-20,12,1,5,0,5:ENT -2,10,-5,3,1,10,0,5:ENT 3,1,20,1,5,1,-20,
1,5:ENV 1,1,0,5,1,15,15
17 ' *** Titelbild ***
18 BORDER 0:PAPER 0:INK 0,0:MODE 1:INK 2,0 :INK 3,0:INK 13,9,11:SPEED INK 8,8:z=
2
19 PEN 1:PRINT TAB(10)" Ein Moment bitte !":PEN 2
20 LOCATE 1,25 : PRINT "Krumi"
21 PEN 3 :FOR i=0 TO 8 : FOR k=0 TO 40
22 IF TEST(2*k,2*(8-i))=2 THEN z=5-z:PEN z:LOCATE k+1,i+7:PRINT CHR$(245): ELSE
23
24 LOCATE 18,8:PRINT CHR$(247)CHR$(245) SPC(2)CHR$(247)CHR$(245)
25 LOCATE 1,25:PRINT " " :LOCATE 1,1:PRINT STRING$(40,32):;INK 3,1,7:INK 2,7,
1
26 LOCATE 16,1:PRINT CHR$(164):" 1985 by Markus Buehler"
27 INK 1,24 : PEN 1 : LOCATE 8,23 : PRINT "Schwierigkeitsgrad (1-3) : ";
28 s9=VAL(INKEY$):IF s9<1 OR s9>3 THEN 28 ELSE PRINT s9:leben=6-s9:s9=3*s9
29 LOCATE 12,25 : PRINT "TASTE DRUECKEN":CALL &BB10
30 MODE 0:INK 0,0:INK 1,24:INK 2,1:INK 3,6:SPEED INK 10,10
31 ' *** Spielfeld aufbauen ***
32 MODE 0:BORDER 12:WINDOW 1,20,1,20 :INK 1,24:INK 0,1:INK 6,2:PAPER 5
33 GOSUB 142
34 WINDOW #1,1,20,21,25 : PAPER#1,9 : PEN #1,5 : CLS#1
35 PRINT #1: PRINT#1,"Kekse : "USING"####";Punkte
36 PRINT #1:PRINT#1,"Vesper: "USING"####";ve
37 INK 11,12,1:PEN#1,11:PRINT #1,"Rekord: "USING"####";re:PEN #1,5
38 PEN 3 : LOCATE 10,11 : PRINT CHR$(247)
39 WINDOW #3,16,20,22,22:PAPER #3,9:INK 7,3:PEN #3,7:PRINT#3,STRING$(leben,247);
40 WINDOW #2,14,14,22,23:PAPER #2,9:PEN2=9:CLS#2:INK 15,6,11
41 ' *** Spielbeginn ***
42 JAGD=JAGD-1:IF JAGD=0 THEN PEN2=9:PAPER#2,PEN2:CLS#2
43 IF jaed>0 THEN SOUND 2,50,10,14,,3
44 g=3-g:vk0=vk:yk0=yk
45 ' *** STEUERUNG ***
46 p0=p:p#=INKEY$:IF p#="" THEN 47 ELSE p=ASC(p#)
47 IF p=11 OR p=240 THEN ri=1 : ko=-1:rk=2:GOTO 52
48 IF p=10 OR p=241 THEN ri=2 : ko=1:rk=1:GOTO 52
49 IF p=8 OR p=242 THEN ri=4 : ko=-1:rk=3:GOTO 55
50 IF p=9 OR p=243 THEN ri=8 : ko=1:rk=4:GOTO 55
51 p=p0:GOTO 46
52 yk=yk+ko
53 IF TEST(xk*32-16,(25-yk)*16+8)=0 OR TEST(xk*32-16,(25-yk)*16+8)=6 OR yk>20 OR
yk<1 THEN yk=yk-ko:Punkte=Punkte-3:GOSUB 83:SOUND
4,500,10,15:GOTO 66
54 GOTO 57
55 xk=xk+ko
56 IF TEST(xk*32-16,(25-yk)*16+8)=0 OR TEST(xk*32-16,(25-yk)*16+8)=6 OR xk>20 OR
xk<1 THEN xk=xk-ko:PUNKTE=PUNKTE-3:GOSUB 83:SOUND 4
,500,10,15:GOTO 66

```

\* \* \* \* \*

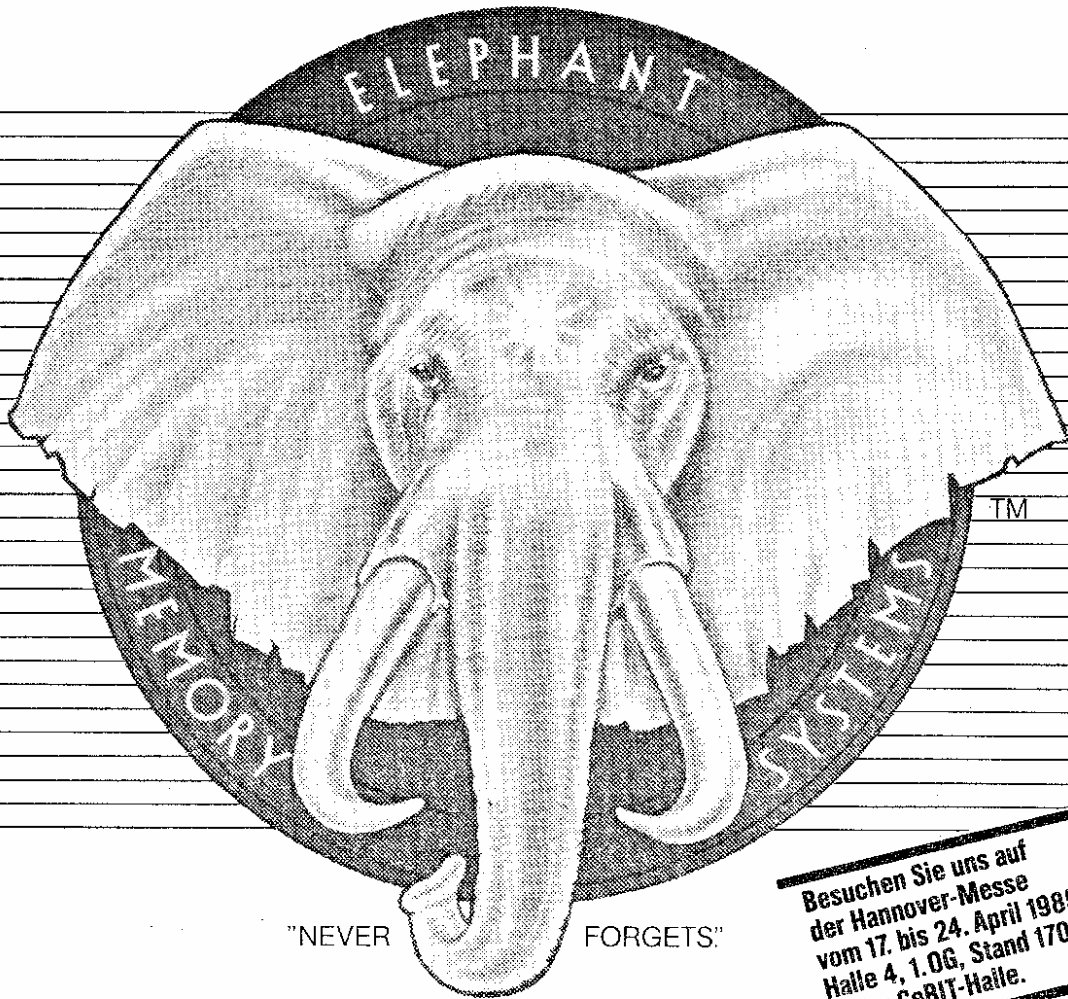


# programme

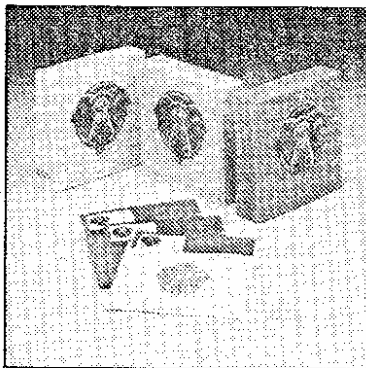
```
57 IF z=0 THEN z=1: 9f=1 ELSE z=0:9f=30
58 te=TEST(xk*32-16,(25-yk)*16+8)
59 IF te=1 OR (geist=1 AND te=3) THEN pu=pu+1:GOSUB 83
60 IF te=3 AND geist=0 THEN pu=pu+1:GOSUB 128
61 IF te=5 AND ve=0 THEN LOCATE#1,1,3:PEN #1,13:PRINT#1,"VesPer: 0":FOR i=1 TO 10:SOUND 1,100,20,0,1:NEXT:WHILE SQ(1)=132:WEND:SO
UND 1,100,30,0:GOTO 130
62 IF te=5 THEN GOSUB 129
63 PEN 3:LOCATE xk,yk:PRINT kr$(z,rk):LOCATE xk0,yk0:PRINT " ";
64 SOUND 1,0,10,13,,,9f
65 IF x(1)=xk AND y(1)=yk THEN 99 ELSE IF x(2)=xk AND y(2)=yk THEN 99
66 IF x(1)=xk AND y(1)=yk THEN 99 ELSE IF x(2)=xk AND y(2)=yk THEN 99
67 ' *** Geister beweegen ***
68 IF geist=1 THEN g=1
69 LOCATE x(g),y(g):PEN Pen9(g):IF Pen9(g)=6 THEN PRINT CHR$(143);ELSE PRINT CHR
$(144);
70 PEN 1:x0(g)=x(g):y0(g)=y(g)
71 IF g<10*RND THEN 89
72 IF y(g)>yk THEN y(g)=y(g)-1
73 IF y(g)<yk THEN y(g)=y(g)+1
74 IF x(g)<xk THEN x(g)=x(g)+1
75 IF x(g)>xk THEN x(g)=x(g)-1
76 IF TEST(x(g)*32-16,(25-y(g))*16+8)=0 THEN 96
77 g1=2/g:IF ABS(x(g)-x(g1))<2 AND ABS(y(g)-y(g1))<2 THEN 89
78 IF x(g)<1 THEN x(g)=1 ELSE IF x(g)>20 THEN x(g)=20 ELSE IF y(g)<1 THEN y(g)=
1 ELSE IF y(g)>20 THEN y(g)=20
79 Pen9(g)=TEST(x(g)*32-16,(25-y(g))*16+8):IF Pen9(g)=4 THEN Pen9(g)=5
80 IF x(g)=xk AND y(g)=yk THEN 99
81 PEN 4:LOCATE x(g),y(g):PRINT geist$:
82 GOTO 41
83 ' *** Anzeige ***
84 Punkte=Punkte+2:IF Punkte<0 THEN Punkte=0
85 LOCATE #1,8,2:PRINT #1,USING "####":Punkte
86 IF pu>=257 THEN GOSUB 151
87 RETURN

88 ' *** neue FLaetze suchen+ueberPruefen ***
89 x(g)=x0(g):y(g)=y0(g)
90 rn=INT(RND*4)+1
91 ON rn GOTO 92,93,94,95,90
92 y(g)=y(g)+1:GOTO 76
93 y(g)=y(g)-1:GOTO 76
94 x(g)=x(g)-1:GOTO 76
95 x(g)=x(g)+1:GOTO 76
96 IF TEST(x0(g)*32-16,(25-y(g))*16+8)<>0 THEN x(g)=x0(g):GOTO 76
97 IF TEST(x(g)*32-16,(25-y0(g))*16+8)<>0 THEN y(g)=y0(g):GOTO 76
98 GOTO 89
99 ' *** Zusammenstoss ***
100 IF PEN2=9 OR geist=1 THEN 101 ELSE 114
101 LEBEN = LEBEN-1
102 SOUND 2,70,245,15,,1:WHILE SQ(2)=132
103 LOCATE xk,yk:PEN 4:PRINT geist$:FOR i=0 TO 100:NEXT:LOCATE xk,yk:PEN 3:PRINT
kr$(z,rk):CALL &BD19:FOR i=0 TO 100:NEXT:WEND
104 LOCATE #3,leben+1,1:PRINT#3," ";
105 IF LEBEN=0 THEN 130
106 PUNKTE=PUNKTE-50
107 IF PUNKTE<0 THEN PUNKTE=0
108 LOCATE #1,8,2:PRINT #1,USING"####":PUNKTE
109 L=1:GOSUB 140:GOSUB 139:X(1)=2:Y(1)=2:LOCATE 2,2:PRINT GEIST$:
110 IF geist=1 THEN 112
111 L=2:GOSUB 139:X(2)=19:Y(2)=19:LOCATE 19,19:PRINT GEIST$:
112 PEN 3:LOCATE xk,yk:PRINT CHR$(245);
113 GOTO 41
114 L=1:GOSUB 139:L=2:GOSUB 139:LOCATE xk,yk:PEN 3:PRINT kr$(z,rk)
115 ' *** Geist fressen
116 x(2)=400:y(2)=250:IF RND>0 THEN x(1)=2:y(1)=2 ELSE x(1)=19:y(1)=19
117 Punkte=Punkte+79:ve=ve+41
118 INK 3,5,21:SOUND 2,500,250,15,,2:WHILE SQ(2)=132:WEND:INK 3,6
119 L=1:GOSUB 140:GOSUB 83:GOSUB 129
```

# DOPPELT GEWINNEN MIT "ELEPHANT"!



Besuchen Sie uns auf  
der Hannover-Messe  
vom 17. bis 24. April 1985,  
Halle 4, 1. OG, Stand 1701/1801  
in der CeBIT-Halle.



Jetzt gibt es neben dem 1. Gewinn, der überragenden Qualität der "ELEPHANT"-Floppy-Disk, noch einen triftigen Grund mehr, schnell Ihren "ELEPHANT"-Händler aufzusuchen: ein Puzzle-Spiel, das Ihnen einen schönen Extra-Gewinn einbringen kann. Was es zu gewinnen gibt, steht auf dem fertigen Puzzle!  
Also nichts wie hin zu Ihrem "ELEPHANT"-Spezialisten.

**Dennison**

## ELEPHANT™ NEVER FORGETS.

**MARCOM Computerzubehör GmbH**

Podbielskistraße 321, 3000 Hannover 51, Tel. (05 11) 64 74 20

Frankreich: Soroclass, 8, Rue Montgolfier - 93115, Rosny-Sous-Bois, Tel.: 16 (1) 855-73-70

Großbritannien: Dennison Mfg. Co. Ltd., Colonial Way, Walford WD2 4JY, Tel.: 0923 41244, Telex: 923321

Weiteres Ausland: Dennison International Company, 4006 Erkrath 1, Matthias-Claudius-Straße 9, Telex: 858 6600



# HAUNTED CASTLE

## Schneider CPC-464

Sie sind ein Abenteurer und laufen durch ein Spukschloß, um Schätze einzusammeln. Aber das ist nicht so einfach, da die Schätze durch geisterhafte Wächter und ein fliegendes Beil bewacht werden. Zudem rückt die rechte Wand des Schlosses immer näher auf Sie zu.

Eine Berührung mit der Wand oder einem Wächter sowie dem Beil wäre tödlich.

Jeder Raum besteht aus zwei Zimmern. Im rechten befindet sich der Schatz, der Wächter, die bewegliche Wand und zu Beginn des Spiels der Abenteurer.

Im linken sind das Beil und eine Tür, um in den nächsten Raum zu gelangen.

Zu Beginn des Spiels besitzen Sie 4 Leben (die Ersatzleben werden unten angezeigt). Hat die bewegliche Wand

den gesamten rechten Raum ausgefüllt, verlieren Sie ein Leben.

Von den Wächtern können Sie nur im rechten Zimmer verfolgt werden. Es gibt pro Raum auch nur einen Wächter.

Manchmal erscheint anstatt eines Schatzes ein Schutzschild. Wenn es Ihnen gelingt, ihn aufzunehmen, können die Wächter und das Beil Ihnen nichts anhaben. Die Wirkung des Schildes erlischt, sobald Sie den Raum verlassen.

Als weitere Schwierigkeit haben die Wächter die Möglichkeit, über die Schätze in den beiden Räumen zu laufen und diese durch Berührung unsichtbar werden zu lassen.

Eine gute Grafik und ein schneller Spielablauf mit verschiedenen Schwierigkeitsstufen sorgen für eine große Abwechslung und Spannung.



```

10 REM "Haunted Castle"
20 MODE 0
30 REM TITELBILD
40 GOSUB 2370
50 qw$="*****"
60 PEN 1:LOCATE 1,2:PRINT qw$:LOCATE 1,24:PRINT qw$
70 MOVE 50,250
80 DRAW 50,300,15
90 MOVE 80,250:DRAW 80,300
100 MOVE 51,275
110 DRAW 79,275
120 BORDER 12:PEN 15:LOCATE 4,9:PRINT"HAUNTED"
130 x=305:y=180:r=25
140 FOR g=75 TO 280:DEG
150 PLOT x+r*COS(g),y+r*SIN(g),15
160 NEXT g
170 LOCATE 11,15:PRINT"ASTLE"
180 LOCATE 2,20:PEN 9:PRINT"@ by":LOCATE 2,22:PRINT "Rainer Krotz"
190 FOR tt=1 TO 10000:NEXT tt
200 hi=1000
210 sc=0
220 sch=20
230 ss9=0.01
240 l=3
250 ENT 1,200,5,1
260 INK 6,8
270 MODE 1
280 WINDOW #1,1,40,21,25:PAPER #1,6:CLS #1
290 WINDOW #2,1,40,1,20
300 REM BILDSCHIRMAUFBAU
310 CLS #2:PEN 7:ux=INT(RND(1)*10)+2:uy=INT(RND(1)*9+10)
320 LOCATE ux-1,1:PRINT CHR$(233):FOR uw=ux TO 40

```

# programme

```
330 LOCATE ux,uy:PRINT CHR$(233)
340 LOCATE ux,1:PRINT CHR$(233)
350 NEXT uy
360 FOR lw=2 TO uy
370 LOCATE ux-1,lw:PRINT CHR$(233):NEXT lw
380 LOCATE ux-1,uy:PRINT CHR$(233)
390 fx=40
400 FOR fy=2 TO lw-2
410 LOCATE fx,fy:PRINT CHR$(207)
420 NEXT fy
430 sch=sch-1
440 IF sch<3 THEN sch=20
450 tx=ux+3
460 PEN 1:LOCATE tx,3:PRINT CHR$(232)
470 PEN 7
480 fg=0
490 sz=INT(RND(1)*8+4)
500 swx=tx+sz
510 FOR tw=2 TO uy-1
520 LOCATE swx,tw:PRINT CHR$(138):NEXT tw
530 st=uy-INT(RND(1)*15):IF st<2 THEN st=2
540 IF st>=uy THEN st=uy-1
550 LOCATE swx,st:PRINT " "
560 PEN 1
570 LOCATE 2,23:PRINT "SCORE: ";sc
580 bx=swx-1:by=2
590 b#=CHR$(215)
600 LOCATE 17,23:PRINT "HIGHSCORE: ";hi
610 LOCATE 36,23:PRINT CHR$(249);1
620 PEN 1
630 wbt=1
640 ss9=ss9+0.01
650 LOCATE 13,19:PRINT"HAUNTED CASTLE"
660 sst=1:s9=0
670 REM SCHATZ
680 e=INT(RND(1)*30+1):ON e GOSUB 1570,1570,1570,1570,1570,1290,1290,1290,1450,1
450,1450,1540,1540,1540,1260,1260,1320,1410,1410,129
0,1350,1350,1510,1510,1480,1480,1480,1380,1380,1380
690 sx=swx+INT(RND(1)*3+1)
700 sy=INT(RND(1)*17+2)
710 IF sy>=uy THEN sy=uy-1
720 LOCATE sx,sy:PRINT s#
730 REM WAECHTER
740 g=INT(RND(1)*42+1):ON g GOSUB 1600,1600,1600,1670,1670,1640,1640,1640,1700,1
790,1790,1790,1880,1880,1940,1940,1600,1640,2030,203
0,2060,2060,2090,2090,1730,1730,2120,2120,1970,1970,1970,2000,2000,1760,1760,176
0,1820,1820,1850,1850,1910,1910,1910
750 gx=swx+4
760 gy=INT(RND(1)*16+2)
770 IF gy>=uy THEN gy=uy-1
780 LOCATE gx,gy:PRINT g#
790 bw=bw+ss9
800 a#=CHR$(251)
810 a1#=CHR$(251)
820 a2#=CHR$(250)
830 x=fx-2
840 y=3
850 INK 14,11
860 LOCATE x,y:PRINT a#
870 wbt=wbt+1:IF wbt=sch THEN GOSUB 2150
880 LOCATE gx,gy:PRINT " "
890 IF gx>x THEN gx=gx-bw
900 IF gx<x THEN gx=gx+bw
910 IF gy<y THEN gy=gy+bw
920 IF gy>y THEN gy=gy-bw
930 IF gx<=swx+0.5 THEN gx=gx+1.5
940 PEN 14
950 LOCATE gx,gy:PRINT g#
960 PEN 1
```

## Variablendefinierung:

- sc Punkte
- sch Schnelligkeit der Wand
- ssg Schnelligkeit der Wächter
- sx, sy Position Schatz
- gx, gy Position Wächter
- bw Position Beil
- x, y Position Abtueurer



```

970 IF ax<=x+1 AND ax>=x-1 THEN 990
980 GOTO 1000
990 IF ay<=y+1 AND ay>=y-1 AND fa=0 THEN 2250
1000 LOCATE bx,by:PRINT " "
1010 by=by+1
1020 IF by=uy THEN by=2
1030 LOCATE bx,by:PRINT b$
1040 IF x=bx AND y=by AND fa=0 THEN 2250
1050 REM JOYSTICKABFRAGE UND BEWEGUNG
1060 IF JOY(0)=4 THEN LOCATE x,y:PRINT " :x=x-1:a$a1$:GOTO 1100
1070 IF JOY(0)=8 THEN LOCATE x,y:PRINT " :x=x+1:a$a2$:GOTO 1100
1080 IF JOY(0)=1 THEN LOCATE x,y:PRINT " :y=y-1:GOTO 1100
1090 IF JOY(0)=2 THEN LOCATE x,y:PRINT " :y=y+1:GOTO 1100
1100 IF x=sux AND y(<)st THEN 2250
1110 IF x=fx THEN 2250
1120 IF y=uy THEN 2250
1130 IF y=1 THEN 2250
1140 IF x=ux-1 THEN 2250
1150 IF x=tx AND y=3 THEN 1220
1160 IF x=sx AND y=sy AND sst=1 THEN 1180
1170 GOTO 1200
1180 sc=sc+sz:SOUND 1,319,20:SOUND 1,284,30:SOUND 1,213,40:ssst=0:LOCATE 8,23:PRI
NT sc
1190 IF s9=1 THEN fa=1
1200 IF x=bx AND y=by AND fa=0 THEN 2250
1210 GOTO 860
1220 sc=sc+1000
1230 SOUND 2,478,30:SOUND 2,319,40:SOUND 2,426,30
1240 GOTO 310
1250 REM SCHATZART UND WERT
1260 s$=CHR$(202)
1270 sz=5000
1280 RETURN
1290 s$=CHR$(203)
1300 sz=2000
1310 RETURN
1320 s$=CHR$(201)
1330 sz=10000
1340 RETURN
1350 s$=CHR$(208)
1360 sz=5000
1370 RETURN
1380 s$=CHR$(209)
1390 sz=2000
1400 RETURN
1410 s$=CHR$(204)
1420 sz=1000
1430 s9=1
1440 RETURN
1450 s$=CHR$(205)
1460 sz=3000
1470 RETURN
1480 s$=CHR$(225)
1490 sz=3000
1500 RETURN
1510 s$=CHR$(226)
1520 sz=4000
1530 RETURN
1540 s$=CHR$(206)
1550 sz=4000
1560 RETURN
1570 s$=CHR$(217)
1580 sz=1000
1590 RETURN
1600 g$=CHR$(211)
1610 bw=0.1
1620 RETURN
1630 REM GEISTERART UND SCHNELLIGKEIT
1640 g$=CHR$(212)
1650 bw=0.2
1660 RETURN
1670 g$=CHR$(210)
1680 bw=0.3
1690 RETURN
1700 g$=CHR$(213)
1710 bw=0.4
1720 RETURN
1730 g$=CHR$(244)
1740 bw=0.2
1750 RETURN
1760 g$=CHR$(245)
1770 bw=0.3
1780 RETURN
1790 g$=CHR$(214)
1800 bw=0.3
1810 RETURN
1820 g$=CHR$(246)
1830 bw=0.1
1840 RETURN
1850 g$=CHR$(247)
1860 bw=0.4
1870 RETURN
1880 g$=CHR$(215)
1890 bw=0.2
1900 RETURN
1910 g$=CHR$(248)
1920 bw=0.2
1930 RETURN
1940 g$=CHR$(218)
1950 bw=0.2
1960 RETURN
1970 g$=CHR$(242)
1980 bw=0.3
1990 RETURN
2000 g$=CHR$(243)
2010 bw=0.2

```

# programme

```
2020 RETURN
2030 g#=CHR$(219)
2040 bw=0.1
2050 RETURN
2060 g#=CHR$(220)
2070 bw=0.3
2080 RETURN
2090 g#=CHR$(240)
2100 bw=0.4
2110 RETURN
2120 g#=CHR$(241)
2130 bw=0.3
2140 RETURN
2150 fx=fx-1
2160 PEN 7
2170 SOUND 3,1012,30
2180 FOR fy=2 TO 10-2
2190 LOCATE fx,fy:PRINT CHR$(207)
2200 NEXT fy
2210 IF fx=swx+1 THEN 2250
2220 wht=1
2230 PEN 1
2240 RETURN
2250 LOCATE x,y:PRINT"+"
2260 SOUND 5,478,210,7,0,1
2270 FOR i=1 TO 1000:NEXT i
2280 l=l-1:IF l>=0 THEN 310
2290 PEN 1:LOCATE 15,10:PRINT"%% GAME OVER %"
2300 IF sc<hi THEN 2330
2310 hi=sc:LOCATE 27,23:PRINT hi
2320 SOUND 5,478,40:SOUND 5,379,40:SOUND 5,478,40:SOUND 5,284,40:SOUND 5,478,40:
SOUND 5,239,80
2330 LOCATE 15,17:PRINT"PRESS FIRE TO PLAY !!!"
2340 IF JOY(0)=16 THEN 210
2350 GOTO 2340
2360 REM ZEICHEN
2370 SYMBOL AFTER 200
2380 SYMBOL 201,0,0,16,40,84,170,0,0
2390 SYMBOL 202,0,0,0,146,146,84,124,124
2400 SYMBOL 203,0,56,16,40,68,40,16
2410 SYMBOL 204,0,124,108,68,108,124,56,16
2420 SYMBOL 205,16,40,68,68,40,16,56,16
2430 SYMBOL 206,214,84,56,16,16,16,16,16
2440 SYMBOL 208,48,24,12,38,83,255,255,255
2450 SYMBOL 209,254,124,56,16,16,16,56
2460 SYMBOL 225,0,68,146,186,84,40,214,124
2470 SYMBOL 226,56,84,186,238,186,84,56
2480 SYMBOL 217,16,16,56,108,108,56,16,16
2490 SYMBOL 210,20,8,62,85,28,20,20,54
2500 SYMBOL 211,8,26,75,210,74,106,42,62
2510 SYMBOL 212,28,62,42,42,62,126,254,170
2520 SYMBOL 213,124,254,146,146,238,84,40
2530 SYMBOL 214,66,36,24,24,219,126,60
2540 SYMBOL 215,2,6,30,14,10,8,8,8
2550 SYMBOL 216,4,18,16,172,73,84,173,255
2560 SYMBOL 218,0,66,36,153,126,24,38,64
2570 SYMBOL 219,16,56,42,58,124,124,56,108
2580 SYMBOL 220,65,65,34,62,42,54,34,28
2590 SYMBOL 232,127,65,65,81,65,65,65,65
2600 SYMBOL 240,7,29,116,208,128,129,149,255
2610 SYMBOL 241,0,20,127,93,93,20,20,54
2620 SYMBOL 242,0,17,142,132,124,74,145
2630 SYMBOL 243,34,28,20,28,127,62,8,20
2640 SYMBOL 244,8,28,42,119,34,28,20,54
2650 SYMBOL 245,96,144,88,24,60,90,129,102
2660 SYMBOL 246,56,56,16,56,84,146,162,148
2670 SYMBOL 247,16,57,17,125,187,57,69,199
2680 SYMBOL 248,255,153,255,195,189,255,36,102
2690 RETURN
```



# ROM-Calls ohne Einsprungsbedingungen

Stellen Sie sich vor, Sie haben mit viel Mühe einige Tasten Ihres CPC neu definiert und wollen das nun wieder rückgängig machen. Eine Möglichkeit wäre natürlich, die vorherige Prozedur in umgekehrter Reihenfolge zu wiederholen, aber rationeller ist CALL &BB00. Er setzt alle Tasten zurück und auch die alten Repeat-Werte werden wieder eingesetzt.

Aber wie ist es, wenn Sie nur die Funktionstasten zurücksetzen wollen? Auch mit oberem Call hätten Sie dann noch viel Arbeit. Hier hilft CALL &BB03. Er leert auch den Tastaturbuffer. Das ist extrem nützlich, da die Tastaturträge abgebaut wird und Inputs besser eingesetzt werden können.

Beide Calls sperren auch die Escape-Taste. Das ist ein schöner Programmschutz, während des Programmierens jedoch eminent gefährlich. Man sollte mit dem Sperren von Escape und damit der Breakfunktion sehr vorsichtig sein. Wollen Sie nur Break sperren, hilft CALL &BB48.

Was aber ist zu tun, will man obigen Call in seinem Programm verwenden, nicht aber in ständiger Angst vor einer Endlosschleife leben, während man noch entwickelt? Hier hilft CALL &BB45. ESC wird wieder aktiviert und Sie können beruhigt weiterarbeiten.

Es ist erfreulich, ESC ausschalten zu können, doch kann man das Programm nach wie vor mit ESC+SHIFT+CTRL „rauswerfen“. Aber auch das kann man unterbinden. POKE &BDEE, 201 wirkt. 201 ist die Zahl, die in Assembler – Return – bedeutet. Die Abfrage, ob die drei Tasten zusammen gedrückt wurden, wird unterlassen.

Um diesen Zustand wieder aufzuheben, gibt man folgenden Befehl ein: POKE &BDEE, 195. Die Abfrage findet wieder statt.

Besonders häufig kommt es vor, daß der Anwender eine Taste drücken soll, um im Programm fortzufahren. Was ist zu tun? Das Basic schafft diverse Möglichkeiten, am sinnvollsten erscheint jedoch ein kurzer Call.

## Schneider- Computer- Software

Die qualitativ hochwertige Software nutzt die Vorteile, die sich aus den hervorragenden technischen Merkmalen des „CPC464“ ergeben. Und die „Speed-Load“-Eigenschaft gestattet es, auch umfangreiche Programme sehr schnell zu laden. Schneider Software bietet eine breite Palette von Spezialprogrammen sowie eine Reihe von Lernprogrammen, Programmen aus dem kaufmännischen Bereich und natürlich die besten der bekannten Computerspiele. Die Lernprogramme bringen schon den Jüngsten viel Spaß, machen das Lernen interessanter und führen so schnell zum gewünschten Erfolg. Die Computerprogramme sind anspruchsvoll, unterhaltsam und voller Farben. Die ausgezeichneten Klang- und Graphikmöglichkeiten des „CPC464“ kommen dabei voll zur Geltung. Die kaufmännischen Programme nutzen die 80-Zeichen/Zeile des professionellen Bildschirms und die mächtige Kapazität, Daten zu verarbeiten und zu verwalten.

Sowohl CALL &BB18 als auch CALL &BB06 helfen und nehmen auch ein Zeichen aus dem Tastaturbuffer.

Besonders verwirrend sind die extrem guten Farbeffekte des Schneiders. Sie zu nutzen ist gelegentlich so schwierig, daß man sich unversehens in blauer Schrift auf blauem Grund findet. Nun kann man langwierig im Dunklen tappen und Inks modifizieren oder man kennt folgenden Call, der die Anfangswerte setzt: CALL &BC02. Dieser Call setzt auch die Blinkgeschwindigkeit auf den Ausgangswert.

Normalerweise kann der Datacorder nur über Lade- und Speicherbefehle aktiviert werden. Nun haben Sie die Chance, es zu ändern. CALL &BC6E schaltet ihn an und CALL &BC71 schaltet ihn wieder aus.

Die Soundmöglichkeiten des CPC 464 sind enorm. Wenn Sie jedoch ein Lied, das der Computer gerade spielt, unterbrechen, spielt er die Töne, die noch in der Warteschlange sind, zu Ende. Das kann ärgerlich sein, wenn man im Programm zu einem bestimmten Zeitpunkt den Sound ändern will. Setzen Sie einfach CALL &BCA7 und es herrscht Ruhe.

Es kann natürlich auch vorkommen, daß Sie nur eine Ruhepause brauchen, danach aber mit derselben Melodie fortgefahren werden soll. Setzt man nun den ebengenannten Call, kommt es zu einer häßlichen Lücke im Melodieverlauf. Benutzt man jedoch CALL &BCB6 zur Pausenerzeugung, so kann man nach der Unterbrechung lückenlos mit CALL &BCB9 fortfahren. Viel Spaß und effizientes Programmieren wünschen Ihnen nun Marcus Schneider und Stephan Schulz.

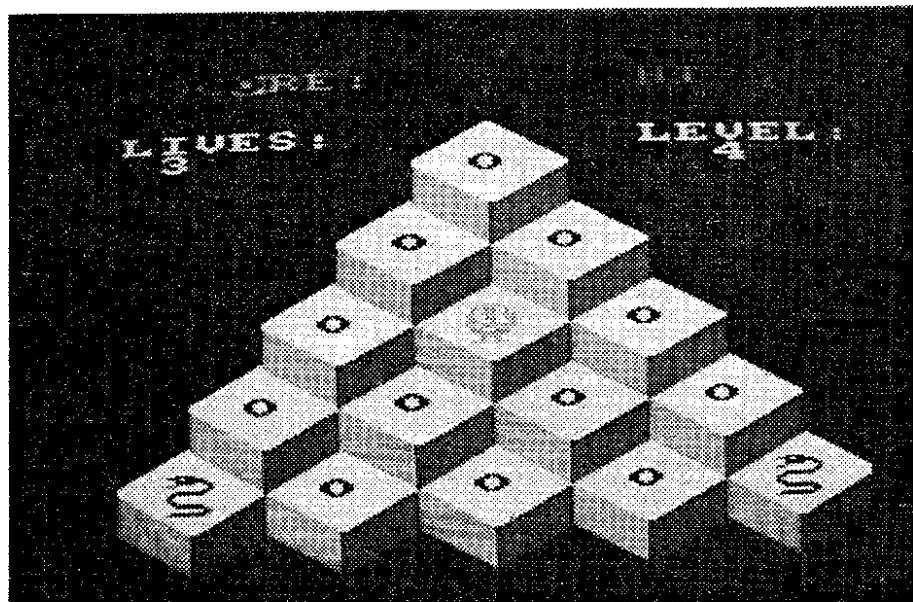
# Q-BERT

Schneider CPC-464

**Ziel dieses Spieles ist es, Q-Bert über die Würfelpyramide zu steuern und dabei die Eier der Schlangen einzusammeln.**

Dabei muß Q-Bert jedoch darauf achten, daß er nicht mit den Schlangen zusammentrifft, die die Verfolgung von der Spielfigur übernommen haben. Das Programm besteht aus verschiedenen Spiel-Levels. In Level 1-2 gibt es nur eine Schlange die Sie verfolgt, ab Level 3 sind es bereits zwei. Als weitere Schwierigkeit legen in diesem Level die Schlangen auch hin und wieder ein Ei dazu oder nehmen eines wieder weg. Wer es trotzdem schafft alle Schwierigkeitsstufen zu überwinden, der darf sich vielleicht in die „HALL OF FAME“ eintragen.

Die Steuerung erfolgt über die Tasten Y, U, H und J. Sie haben aber auch die Möglichkeit einen Kempston Joystick zu benutzen.



## Numerische Variable:

nn = Schleifenzähler beim Eingeben der MC-Routine (Zeile 120)  
 vv = Gelesener Wert aus DATA-Zeile 110  
 t = Zeitschleifenzähler  
 a,b,c,d,e und f = Variablen im Zusammenhang mit Bildschirmgrafik  
 fl = Zahl der Eier, die eingesammelt werden müssen  
 lev = Level  
 liv = Zahl der Leben  
 i = Joystick-abfrage (Wert)  
 pox = X-Koordinate v. Q-BERT  
 poy = Y-Koordinate v. Q-BERT  
 bon = Bonus  
 sc = Score  
 s = Index v. versch. Arrays  
 u = Sound-Frequenz (Zeile 565)  
 hall = Index v. versch. Arrays

```

tim = Wiederholungsschleifenzähler
tin = Farbwert in Zeile 765
y1 = ink 4
y2 = ink 5
y3 = ink 6
y4 = ink 8
y5 = ink 7
z = Index v. Array (Zeile 915)
le = Länge des eingegebenen Namens (Zeile 1010)
v = Index v. Array (Zeile 1035)
ti = Zeitschleifenzähler

```

## Stringvariable:

```

l$ = inkey$
nam$ = eingegebener Name
inpu$ = inkey$

```

## Numerische Arrays:

```

hsc(6) = 6 höchste Scores
o(n) = Testvariable, 1, wenn auf aktueller Position ein Ei liegt
sx(n) = X-Koordinate v. Schlange N
sy(n) = Y-Koordinate v. Schlange N

```



# hardware

## GP-500CPC: Ein speziell für den Computer Schneider CPC-464 entwickelter 50-Zeichen/Sek.-Drucker

Der GP-500CPC ist problemlos über das mitgelieferte Kabel anschließbar. Bei einer Matrix-Konfiguration von 5 x 7 Punkten werden Commodore-Blockgrafik bei bis zu 80 Zeichen-/Zeile dargestellt. Hochauflösende Grafik durch Punktadressierung mit bis zu 480 Punkten/Zeile kann eigen-definiert werden. Zwei verschiedene Zeilenvorschübe von 6 Zeilen/Zoll und 9 Zeilen/Zoll bei Vorschubgeschwindigkeiten von 10 Zeilen/Sek. und 15 Zeilen/Sek. sind wählbar, alphanumerische Zeichen sind in Normalgröße oder in doppelter Breite ausdrückbar. 8 verschiedene, länder-spezifische Zeichensätze sowie Wa-genrücklauf und Zeilenvorschub sind

durch leicht zugängliche DIP-Schal-ter von außen wählbar. Der Drucker druckt auf Standard-EDV-Papier bis zum maximalen Format DIN A4. Der Papiertransport erfolgt über Stachel-rad durch Schrittmotor.

## GP-700CPC

Bei weitgehend technischen Daten ist dieser Drucker über ein zum Liefer-umfang gehörendes Verbindungska-bel direkt an den Computer Schnei-der CPC-464 anschließbar.

## Fans des Schneider CPC 464

mögen ein wenig enttäuscht gewesen sein, daß das SYBEX-Buch **Schnei-der CPC 464: Mein erstes BASIC Programm** noch nicht auf dem Weihnachts-Gabentisch zu finden war. Wir wollten halt unserer Verlags-Philosophie treu bleiben: Produkte herauszubringen, die wirklich voll ausgereift sind und den Anwender nicht im Stich lassen, wenn es darauf ankommt – selbst, wenn es mal ein bißchen länger als geplant dauert. Die Weiterentwicklung von **Rodnay Zaks' „Mein erstes BASIC Pro-gramm“** für den leistungsfähigen Colour Personal Computer von Schneider ist ab Februar für DM 32,- erhältlich.



# programme

cr(n) = Testvariable, 1, wenn Zusammenstoß v. Q-BERT mit Schlange  
N erfolgt ist

String-Arrays:

hi\$(6,16) = Namen der 6 besten Spieler

```
10 REM *****
15 REM *
20 REM * Q - B E R T *
25 REM *
30 REM * (C) FEB '85 *
35 REM *
40 REM * BY *
45 REM *
50 REM * O.KIESSLING *
55 REM *
60 REM * MAX LEESER STR. 6 *
65 REM *
70 REM *
75 REM *
80 REM *****
85 PRINT CHR$(23);"0";:BORDER 0:PAPER 0:INK 0,0:INK 4,4:INK 5,15:INK 6,24:MODE 0
90 '
95 ' H I - S C O R E L O E S C H E N
100 '
105 CLEAR:SYMBOL AFTER 190:GOSUB 1045:DIM HSC(6):DIM HI$(6,16):DIM O(2):DIM SX(2)
:DIM SY(2):DIM CR(2):KEY 130,"":KEY 139,"MODE 2:PE
N 1:LIST"+CHR$(13):GOSUB 880
110 DATA 33,0,192,17,0,64,1,0,64,237,176,201,33,0,64,17,0,192,1,0,64,237,176,201
115 RESTORE 110
120 FOR NN=33000 TO 33023:READ VV:POKE NN,VV:NEXT:GOSUB 175:CALL 33000
125 '
130 ' T I T L E
135 '
140 MODE 0:BORDER 0:PAPER 0:INK 0,0:INK 1,15:INK 2,0:INK 3,15:INK 4,1:INK 5,2:IN
K 6,11:INK 7,7:INK 8,24:INK 9,2,24:INK 10,16:INK 11,
22
145 LOCATE 8,5:PEN 9:PRINT "Q-BERT":LOCATE 5,10:PEN 10:PRINT "(C) FEB 1985":PEN
11:LOCATE 8,15:PRINT "O.M.K.":PEN 1:FOR T=0 TO 4000:
NEXT T:L$=""
150 L$=INKEY$:IF L$="" THEN 150
155 SC=0:LIV=7:O(1)=1:O(2)=1:BOB=3000:LEV=1:CLS:CALL 33012:GOTO 240
160 '
165 ' B I L D A U F B A U
170 '
175 CLS
180 FOR A=4 TO 0 STEP -1:FOR B=0 TO A*2 STEP 2
185 FOR C=0 TO 32 STEP 2
190 PLOT 64*(B+(4-A))+D,320-(64*A+C)
195 DRAWR 64,-32,4:DRAWR 64,32,5
200 NEXT
205 E=0:F=1:FOR D=0 TO 128:PLOT 64*(B+(4-A))+D,320-(64*A+C)+36-(E/2),6:DRAWR 0,E
,6:E=E+F:IF E>64 THEN F=-F
210 NEXT:MOVE 64*(B+(4-A))+D-80,320-(64*A+C)+32-(E/2)+8:PRINT CHR$(23);"1";:TAG:
PRINT "0";:TAGOFF:PRINT CHR$(23);"0";
215 NEXT:NEXT:PEN 7:LOCATE 1,1:PRINT " SCORE:" :LOCATE 15,1:PRINT "HI-SC:" :LOCATE
1,4:PEN 6:PRINT "LIVES:" :LOCATE 15,4:PRINT "LEVEL:"
220 RETURN
225 '
230 ' G R U N D P O S I T I O N E N
235 '
240 PAPER 0:PRINT CHR$(23);"0";:CR(2)=0:O(1)=1:O(2)=0+(1 AND LEV>2):POX=10:POY=1
3: SX(1)=2: SX(2)=0: SY(1)=21: SY(2)=0: FL=15
```



```

245 IF LEV>2 THEN SX(2)=18:SY(2)=21
250 PAPER 6:PEN 7:LOCATE 10,13:PRINT CHR$(191):CHR$(192)
255 LOCATE 10,14:PRINT CHR$(193):CHR$(194):PAPER 0:PEN 6:LOCATE 1,5:PRINT LIV:LO
DATE 16,5:PRINT LEV:PAPER 6
260 PEN 8:LOCATE 2,21:PRINT CHR$(195):CHR$(196):LOCATE 2,22:PRINT CHR$(197):CHR$
(198)
265 IF LEV>2 THEN LOCATE 18,21:PRINT CHR$(195):CHR$(196):LOCATE 18,22:PRINT CHR$
(197):CHR$(198):PAPER 0:PEN 1
270 '
275 ' H A U P T T E I L
280 '
285 EI:I=JOY(0):IF I=0 AND INKEY$="" THEN 460
290 I=I+(5 AND INKEY(43)=0)+(6 AND INKEY(44)=0)+(9 AND INKEY(42)=0)+(10 AND INKE
Y(45)=0)
295 '
300 ' O - B E R T L O E S C H E N
305 '
310 PAPER 6:LOCATE POX,POY:PRINT CHR$(128):CHR$(128):LOCATE POX,POY+1:PRINT CHR$
(128):CHR$(128)
315 PAPER 0:BON=BON-(100 AND BON>99):SOUND 4,100,10
320 PEN 5:LOCATE 1,2:PRINT SC:LOCATE 14,2:PRINT HSC(1)
325 '
330 ' J O Y - S T I C K A B F R A G E
335 '
340 ' O - B E R T B E W E G E N
345 '
350 IF I=5 OR I=6 THEN POX=POX-2
355 IF I=9 OR I=10 THEN POX=POX+2
360 IF I=5 OR I=9 THEN POY=POY-4
365 IF I=6 OR I=10 THEN POY=POY+4
370 IF SX(1)=POX AND SY(1)=POY THEN 625
375 '
380 ' O - B E R T V O N D E R
385 '
390 ' P Y R A M I D E
395 '
400 ' G E S P R U N G E N ? ? ?
405 '
410 IF POX>19 OR POX<2 OR POY>22 OR TEST(-32+32*POX,400-POY*16)=0 THEN 625
415 '
420 ' S C O R E E R H O E H E N
425 '
430 IF TEST(-16+32*POX,400-POY*16)=0 OR TEST(-16+32*POX,400-POY*16)=4 THEN SC=SC
+25*LEV+(10 AND LEV=0):FL=FL-1
435 IF FL=2 THEN 805
440 '
445 ' O - B E R T D R U C K E N
450 '
455 PAPER 6:PEN 7:LOCATE POX,POY:PRINT CHR$(191):CHR$(192):LOCATE POX,POY+1:PRIN
T CHR$(193):CHR$(194):PAPER 0:PEN 1
460 PAPER 6:PEN 7:LOCATE 2,1:PAPER 0:PRINT "SC":PEN 0:PAPER 6
465 '
470 ' S C H L A N G E L O E S C H E N
475 '
480 FOR S=1 TO 1+(1 AND LEV>2)
485 LOCATE SX(S),SY(S):PRINT CHR$(128):CHR$(128):LOCATE SX(S),SY(S)+1:PRINT CHR$
(128):CHR$(128)
490 IF O(S)=0 THEN 500
495 PRINT CHR$(23):"1":TAG:MOVE -16+32*SX(S),407-SY(S)*16:PEN 0:PAPER 0:PRINT "
0":TAGOFF:PRINT CHR$(23):"0":GOTO 500
500 PAPER 0:PEN 1:O(S)=0
505 '
510 ' S C H L A N G E N B E W E G E N
515 '
520 SOUND 4,200,10
525 IF SX(S)=2 THEN SX(S)=4:SY(S)=17:GOTO 565
530 IF SX(S)=18 THEN SX(S)=16:SY(S)=17:GOTO 565
535 IF SY(S)=5 THEN SY(S)=9:SY(S)=SX(S)+2-(4 AND RND(1)>0.5):GOTO 565

```

# programme

```
540 IF SY(S)=21 THEN SY(S)=17: SX(S)=SX(S)+2-(4 AND RND(1)>0.5): GOTO 565
545 IF TEST((SX(S)*32)-90,400-SY(S)*16+15)=0 THEN SX(S)=SX(S)+2: SY(S)=SY(S)+4-(8
  AND RND(1)>0.5): GOTO 565
550 IF TEST((SX(S)*32)+160,400-SY(S)*16+15)=0 THEN SX(S)=SX(S)-2: SY(S)=SY(S)+4-(
  8 AND RND(1)>0.5): GOTO 565
555 SX(S)=SX(S)-2+(4 AND RND(1)>0.5): SY(S)=SY(S)+4-(8 AND RND(1)>0.5)
560 REM IF SX(S)<2 OR SX(S)>18 OR SY(S)>21 OR TEST((SX(S)*32)-32,400-SY(S)*16+15
  )=0 THEN SX(S)=10: SY(S)=5
565 IF TEST(-16+32*SX(S),400-SY(S)*16)=0 OR TEST(-16+32*SX(S),400-SY(S)*16)=4 TH
  EN FOR U=50 TO 100 STEP 5: SOUND 1,U,1: NEXT: O(S)=1
570 '
575 ' S C H L A N G E N   D R U C K E N
580 '
585 PAPER 6: PEN 8: LOCATE SX(S),SY(S): PRINT CHR$(195): CHR$(196): LOCATE SX(S),SY(S
  )+1: PRINT CHR$(197): CHR$(198)
590 IF (SX(S)=POX AND SY(S)=POY) THEN CR(S)=1 ELSE CR(S)=0
595 NEXT
600 IF SX(1)=SX(2) AND SY(1)=SY(2) THEN O(1)=1: O(2)=1
605 IF CR(1)=0 AND CR(2)=0 THEN 285
610 '
615 ' L E B E N   A B Z I E H E N
620 '
625 DI: LIV=LIV-1: FOR T=0 TO 2500: NEXT T
630 '
635 ' P Y R A M I D E   M I T   0 ' S
640 '
645 ' F U E L L E N
650 '
655 PAPER 6: FOR A=5 TO 1 STEP -1: FOR B=2 TO A*4 STEP 4
660 LOCATE B+2*(5-A),25-4*(6-A): PRINT CHR$(128): CHR$(128)
665 LOCATE B+2*(5-A),26-4*(6-A): PRINT CHR$(128): CHR$(128)
670 LOCATE 1,9: PRINT CHR$(23): "1": MOVE (B+2*(5-A))*32-16,(400-(25-4*(6-A))*16)+
  8: TAG: PEN 0: PAPER 6: PRINT "0": TAGOFF: PRINT CHR$(23)
  : "0":
675 NEXT B: NEXT A: PAPER 0: FL=15
680 '
685 ' W E N N   L I V = 0 ,   D A N N
690 '
695 ' S P I E L   B E E N D E N
700 '
705 IF LIV>0 THEN 240
710 PAPER 0: CLS: LOCATE 6,12: PEN 9: PRINT "GAME OVER": FOR N=1 TO 6000: NEXT: IF SC<
  =HSC(5) THEN 740 ELSE 950
715 '
720 ' H A L L   O F   F A M E
725 '
730 ' Z E I G E N
735 '
740 PAPER 0: MODE 1
745 PEN 2: LOCATE 9,1: PRINT "H A L L   O F   F A M E"
750 FOR HALL=1 TO 5: LOCATE 8,2+2*HALL: PRINT HSC(HALL): LOCATE 17,2+2*HALL: PRINT H
  I$(HALL,16): NEXT
755 TIM=0
760 FOR TIM=0 TO 4
765 FOR TIN=1 TO 26: INK 2,TIN: FOR TI=0 TO 200: NEXT: NEXT
770 NEXT
775 GOTO 140
780 '
785 ' E I N E   R U N D E   W E I T E R
790 '
795 ' +   E X T R A L E B E N
800 '
805 '
810 LEV=LEV+1
815 IF LEV=2 THEN Y1=3: Y2=6: Y3=16: Y4=24: Y5=4: GOTO 845
820 IF LEV=3 THEN Y1=9: Y2=18: Y3=22: Y4=4
825 IF LEV=4 THEN Y1=6: Y2=2: Y3=24: Y4=4
```

```

830 IF LEV=5 THEN Y1=13:Y2=25:Y3=26:Y4=4
835 IF LEV=6 THEN Y1=1:Y2=2:Y3=11:Y4=24:LEV=0:LIV=LIV+1
840 Y5=7
845 INK 4:Y1:INK 5:Y2:INK 6:Y3:INK 7:Y5:INK 8:Y4:SC=SC+(BON AND BON>0):LOCATE 1,
1:PRINT " ":GOTO 655
850 BON=4000
855 '
860 ' HALL OF FAME
865 '
870 ' INITIALISIEREN
875 '
880 DATA 17175,"OLIVER"      "
885 DATA 9450,"INGO"       "
890 DATA 6725,"HORST"     "
895 DATA 4900,"GERALD"    "
900 DATA 1975,"NORBERT"   "
905 DATA 0,000,"         "
910 RESTORE 880
915 FOR Z=1 TO 6:READ HSC(Z):READ HI$(Z,16):NEXT
920 RETURN
925 '
930 ' IN HALL OF FAME
935 '
940 ' EINTRAGEN
945 '
950 CLS
955 PEN 6
960 PRINT " SIE HABEN EINES":PRINT
965 PRINT " DER HOECHSTEN":PRINT
970 PRINT " ERGEBNISSE ERZIELT":PRINT:PRINT
975 PRINT " BITTE TRAGEN SIE":PRINT
980 PRINT " IHREN NAMEN EIN:" :PRINT:PRINT
985 NAM$=""
990 INPUT$="":INPUT$=INKEY$:IF INPUT$="" THEN 990
995 IF INPUT$=CHR$(8) OR INPUT$=CHR$(9) OR INPUT$=CHR$(11) OR INPUT$=CHR$(10) THEN G
OTO 990
1000 IF LEN(INPUT$)=0 THEN 990
1005 IF ASC(INPUT$)=13 THEN NAM$=NAM$+SPACE$(16-LEN(NAM$)):GOTO 1025
1010 LE=LEN(NAM$)-(1 AND LEN(NAM$)>0):IF INPUT$=CHR$(127) THEN NAM$=LEFT$(NAM$,LE
):GOTO 1020
1015 NAM$=NAM$+INPUT$:IF LEN(NAM$)>16 THEN NAM$=LEFT$(NAM$,16):GOTO 990
1020 LOCATE 2,13:PRINT NAM$:SPACE$(2):GOTO 990
1025 LOCATE 2,13:PRINT NAM$
1030 IF LEN(NAM$)<16 THEN NAM$=NAM$+SPACE$(16-LEN(NAM$))
1035 FOR V=5 TO 1 STEP -1:IF SC>HSC(V) THEN HSC(V+1)=HSC(V):HI$(V+1,16)=HI$(V,16
):HSC(V)=SC:HI$(V,16)=NAM$:NEXT
1040 GOTO 740
1045 '
1050 ' USER-DEFINED
1055 '
1060 ' GRAPHICS
1065 '
1070 '
1075 '
1080 ' O - BERT ( CHR# 191-194 )
1085 '
1090 SYMBOL 191,3,7,13,9,27,25,25,28
1095 SYMBOL 192,192,224,144,144,184,152,152,136
1100 SYMBOL 193,27,27,12,15,7,3,2,14
1105 SYMBOL 194,248,184,112,240,224,192,64,112
1110 '
1115 ' S C H L A N G E N ( CHR# 195-198 )
1120 '
1125 SYMBOL 195,7,29,7,7,24,0,7,15
1130 SYMBOL 196,128,224,240,152,8,24,240,224
1135 SYMBOL 197,24,16,24,15,7,0,15,31
1140 SYMBOL 198,0,0,0,224,240,24,240,224
1145 RETURN

```

## Ein neuerlicher Beweis für die hohe Qualität der Dennison-Diskette „Elephant-Memory-Systems“ TM

Als eines der wichtigsten Prüfungskriterien für die Qualität einer Diskette gilt unter Fachleuten die Antwort auf die Frage, wie oft sich ein Diskettentyp formatieren läßt. Als gutes Ergebnis wird hierbei angesehen, wenn ein bestimmter Diskettentyp eine fünf- bis siebenmalige Formatierung ermöglicht, ohne die geringste Ausfallerscheinung zu zeigen.

In einer aufwendigen Versuchsreihe,

die ein angesehenes Software-Haus für seine Wahl des geeignetsten Diskettentyps zu Beginn des Jahres 1985 durchführte, überraschte die EMS-Diskette neben einer anderen, nicht in der Bundesrepublik erhältlichen, die Tester mit der sensationellen Eigenschaft, sich bis zu dreißigmal problem- und fehlerlos formatieren zu lassen!

Mit diesem Ergebnis stellt die „Ele-

phant-Memory-Systems“-Diskette wieder einmal ihre herausragenden positiven Eigenschaften objektiv und im Vergleich mit sogar preislich höher angesiedelten unter Beweis. Der Alleinvertrieb dieser hochwertigen Disketten in der Bundesrepublik Deutschland und Westberlin erfolgt über die MARCOM Computerzubehör GmbH in Hannover, Tel. 05 11 / 64 74 20.

## ★ Kleinanzeigen ★ Kleinanzeigen ★ Kleinanzeigen ★

### Biete an Software

**CPC-464 !!!** 12 gute Spielprogramme auf Kassette, keine Raubkopien, DM 20,- an Postscheck Hamburg 404576-207 oder Schein In Brief an Hans-J. Behrend, Neustadt 3, 3550 Marburg.

**VC 20** Suche Basic-Literatur und Programme für VC 20. Helmut Huter, Obertor 54, A 6628 Vils/Österreich.

### Biete an Hardware

**Tausche VC-64** Spitzenprogramme. Tauschlisten an: Jürgen Hödl, Muggenthaler Str. 10, 8391 Tittling.

**CPC-464 !!!** Verlängerungs-Kabel zwischen Monitor und Keyboard mit passenden Steckern nur DM 20,- an Postscheck Hamburg 32572-201 oder Schein an H. Behrendt, Marrensberg 2, 2390 Flensburg.

**Suche C 64**, gut erhalten, zahle bis 400,- DM oder Tausch gegen VC-20 + 6 KK + Superprogramme, A. Dinsch, Berliner Str. 17, 5980 Werdohl.

### Verschiedenes

● **C 64 Hallo Faulpelze C 64**  
● Wir tippen Ihre C-64-Programme für 2 Pfg. die Zeile ab. Schicken Sie Prg. an: Matthias Meyer, Martin-Luther-Str. 4, 2732 Sittensen. Tel. 0 42 82 / 21 30

## Auftrag für Gelegenheitsanzeigen in „Compute mit ...“

An „Compute mit ...“  
Tronic-Verlag  
Postfach  
3444 Wehretal 1

Name und Adresse \_\_\_\_\_

Abo.-Nr. \_\_\_\_\_

Unterschrift \_\_\_\_\_ Ich zahle sofort nach Rechnungserhalt. Datum \_\_\_\_\_

Unter der Rubrik „Kleinanzeigen“ veröffentlichen wir Gelegenheitsanzeigen für Verkaufsangebote, Kauf- und Tauschgesuche, Kontaktaufnahme bzw. Erfahrungsaustausch usw.

#### Aktuell:

Unsere Preise für „Kleinanzeigen“:  
Jede Kleinanzeige nur 10,- DM

Jede gewerbliche  
Gelegenheitsanzeige nur 15,- DM  
(dürfen nicht unter Chiffre erscheinen)

Ich wünsche folgenden Text zu veröffentlichen:

Grid for text entry consisting of 10 rows and 40 columns.

**Kleinanzeigen  
jetzt noch  
preiswerter**

suche  Software  
biete an  Software

suche  Hardware  
biete an  Hardware

Zutreffendes ankreuzen  
 Tausch  Versch.  
 Kontakte  Chiffre



# Der Farbbildschirm „CTM 640“

Die Erfahrung von Schneider in der Produktion von Farb-TV-Geräten hat zu einem kompakten, zuverlässigen Farbbildschirm für den „CPC464“-Computer geführt.

Nur wenn der Bildschirm so nahe als möglich mit der Elektronik verbunden ist, die die Bildschirmanzeige steuert, wird die bestmögliche Leistung der hohen Bildschirmauflösung erreicht. Das gilt ebenfalls für die Graphik und Farbmöglichkeit des „CPC464“.

Mit dem „CTM 640“ wird jede einzelne Farbe auf dem Bildschirm direkt vom Computer gesteuert. Dadurch entfallen alle unnötigen Zwischenschritte, die ein unscharfes Bild erzeugen könnten. Sie bekommen immer die besten Bilder und brauchen sich nicht wie beim Fernseher um die Feineinstellung zu kümmern. Und es

gibt keinen Streit, wenn jemand fernsehen will, ein anderer aber zur gleichen Zeit den Computer braucht.

Der „CTM 640“ ist mit zuverlässigen „Solid-State“-Bausteinen ausgestattet. Die „schnelle Aufwärmung“, helle scharfe Anzeige und neueste Bildschirm-Technologie mit in-line Elektronenstrahl-Regelung sorgen für hohe Bildqualität.

Mit dem Tragegriff transportieren Sie das System leicht und sicher. Die Stromversorgung im Bildschirmgehäuse reduziert die Anzahl der Kabelverbindungen auf ein einziges Netzkabel und ein Verbindungskabel zwischen Tastatur und Bildschirm. Damit ist auch der Daten-Recorder angeschlossen.

Vergleichen Sie das mit anderen Produkten!

Das folgende BASIC-Demonstrationsprogramm verdeutlicht insbesondere die Farbgraphikmöglichkeiten des SCHNEIDER CPC 464:

```

10 MODE 0: INK 0,2 x INK
   1,24: PAPER 0
20 PEN 1: FOR B = 0 TO 26:
   BORDER B
30 LOCATE 3,12: PRINT
   "Rahmenfarbe: ";B
40 SOUND 4,(40-B)
50 FOR T = 1 TO 500: NEXT
   T: NEXT B: CLS
60 FOR P = 0 TO 15: PAPER
   P: PEN 5: PRINT
   "Hintergrundfarbe: ";P: PRINT
70 FOR T = 1 TO 1000: NEXT
   T: CLS: NEXT P
80 GOTO 10

```

## BASIC-Programme zur graphischen Darstellung einer linearen Funktion:

Im Vergleich finden Sie hier ein kurzes Basicprogramm zu graphischen Darstellung einer linearen Funktion für den Schneider CPC 464 und den Commodore 64. Hier können sie selbst testen und vergleichen.

### SCHNEIDER CPC 464

```

10 A = 50
20 B = .7
30 PLOT 10,390
40 DRAW 10,10
50 DRAW 630,10
60 FOR X = 10 TO 630
70 Y = A + B * X!
80 IF Y < 0 OR Y > 389 THEN 100
90 PLOT X,Y
100 NEXT X
110 END

```

### Commodore 64

```

10 A = 50
20 B = .7
30 GOSUB 1000
40 FOR X = 0 TO 200
50 Y = 130
60 GOSUB 2000
70 NEXT X
80 FOR Y = 20 TO 150
90 X=0
100 GOSUB 2000
110 NEXT Y
120 FOR X = 0 TO 200
130 Y = 130-A-B * X
140 IF Y < 0 OR Y > 199 THEN 160
150 GOSUB 2000
160 NEXT X
170 END

```

```

1000 V = 53248
1010 POKE V+17,59
1020 POKE V+24,24
1030 FOR I = 1024 TO 2023
1040 POKE I,14
1050 NEXT I
1060 FOR I = 8192 TO 16383
1070 POKE I,0
1080 NEXT I
1090 RETURN
2000 XK = 8 * INT(X/8)
2010 YK = 320 * INT(Y/8) +
   INT((Y/8-INT(Y/8)) * 8)
2020 EX = 2↑(7-INT((X/8-
   INT(X/8)) * 8))
2030 S = 8192+XK+YK
2040 POKE S,PEEK(S) OR EX
2050 RETURN

```



# Ihr Erfolgsprogramm – aus dem Tronic-Verlag



**CPU**  
monatlich  
Format A4  
**5,50 DM**



**Compute mit**  
monatlich  
Format A4  
**2,80 DM**



**Homecomputer**  
zweimonatlich  
Format A4  
**6,- DM**



**Computronic**  
zweimonatlich  
Format A4  
**6,50 DM**

**Spiele-Kassetten**  
jetzt auch über den  
Zeitschriftenhandel



**Spiele-Kassetten:**  
Commodore 64  
VC-20, Atari,  
TI-99, ZX-Spectrum,  
Laser 2001,  
Schneider CPC 464  
**Je 19,90 DM**

## Spitzenprodukte

für den wachsenden Markt  
der Heimcomputer-Anwender

**Tronic-Verlag GmbH · Postfach 41 · 3444 Wehretal 1**  
**Telefon 0 56 51 - 4 06 43 / 4 06 93**