

monatlich

Compute mit

№ diese **COMMODORE & SCHNEIDER**

3/86

VC-64, VC-20, C-16/116, CPC-464

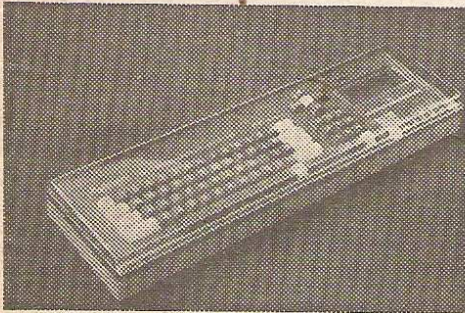
Unabhängiges Magazin für Anwender von Commodore- und Schneider-Computern

2,80 DM
24 ÖS
2,80 sFr

Aktuell +++ Aktuell +++ Aktuell +++ Aktuell

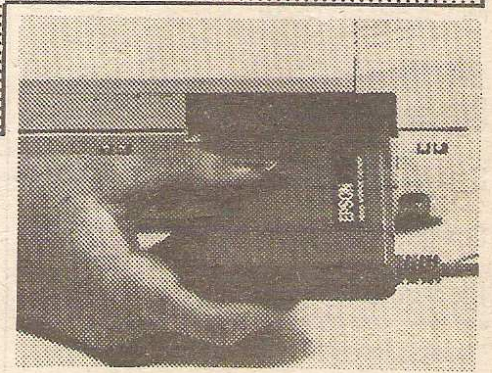
Druckertische und Staubschutzhauben aus Hartplastik

Seit kurzer Zeit hat die Firma Media Plast Druckertische und Staubschutzhauben aus Hartplastik mit in Ihr Lieferprogramm aufgenommen. Die Staubschutzhauben werden nicht nur für die Tastaturen unterschiedlichster Systeme sondern auch für andere Hardware-Bereiche (Floppystationen u. ä.) angeboten. Die Druckertische sind aufgrund Ihrer Bauweise äußerst platzsparend. Mehr darüber können Sie auf der Seite 4 lesen.



Neue Druckersteckmodule von EPSON

Die Fa. EPSON bietet einen neuen einheitlichen Grundmodell-Drucker an, der durch Einstecken eines Moduls zum speziellen Drucker ausgerüstet werden kann. Mehr über diesen neuen Standard finden Sie auf Seite 4 dieser Ausgabe.



Neue C-64 Serie

Nach der erfolgreichen Serie „Tastatur-Verwaltung des C-64“ beginnen wir in diesem Heft mit der Serie „Funktionstasten“. Der Autor dieser ebenfalls 2-teiligen Serie – PK. Bhattacharya – versucht hier das Thema Funktionstasten des Commodore 64 für alle User verständlich und praxisnah zu erklären.

Jetzt auch alle VC-20 Programm-Listings mit Checksummen!

Ab dieser Ausgabe werden auch alle VC-20 Anwender in den Vorteil der Checksummen kommen. Damit wird auch bei der Eingabe dieser Listings anhand der Checksumme genau angezeigt, ob ein Eingabefehler in den einzelnen Zeilen vorliegt oder nicht.

**Kleinanzeigen – Software-Service – Tips & Tricks – Leser-+ Meckerecke
Serie – Report – Werkstatt – Bücher – Software-Abo – Assembler-Kurs
Minor-Jumper – Hero – Starchip – Robin at Ravenwreck – Smily – u.v.m.**

inhalt Software

Commodore

Minor-Jumper (C-16/116)	6
Hero (C-16/116)	12
Checksummer (C-64)	18
Befehlsenerweiterung (C-64)	19
Starchip (C-64)	26
Mini-Car-Race (C-64)	31
Checksummer (VC-20)	36
Robin at Ravenwreck (VC-20)	36
Mini-Karate (VC-20)	44

Schneider

Smily	47
Screpri	57
Die verrückte Fabrik	62

Werkstatt

Data-Maker (C-16/116)	10
Text-Scroller und Text-Editor (C-64)	22
Autonumber-Routine (VC-20)	42
Disc-Help (CPC464/664)	60

Tips & Tricks

für den C-16/116	9
für den C-64	23
für den VC-20	41
für den Schneider	59

Serie

Funktionstasten des Commodore 64	24
----------------------------------	----

Bewerbung als Programmator	15
----------------------------	----

Kleinanzeigen	16
---------------	----

Software-Service	31
------------------	----

Bücher

Textverarbeitung auf dem Commodore 64	34
Denksport auf dem Commodore 64	34
Modem und Hacker Handbuch	35

Assembler-Kurs

Teil 10	45
---------	----

Report

Messebericht	3
Plastik für den Computer	4
LX-90 Drucker von EPSON	4
Anti-Statik-Matten	5
PBOX für den Arbeitsplatzdrucker	5

Sind Aufwendungen für Home-Computer Werbungskosten	55
--	----

Leser + Meckerecke	56
--------------------	----

„Compute mit“

liegt ab

17. März

wieder bei Ihrem
Zeitschriftenhändler
für Sie bereit.

Impressum

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

Redaktion:

Axel Credé (verantwortlich)
Chefredakteur: Hartmut Wendt
Siegfried Görk, Manfred Kleimann, Bernd Zimmermann
Frank Brall, Ottfried Schmidt

Gesamtherstellung:

Druckhaus Dierichs Kassel, Frankfurter Str. 168, 3500 Kassel

Vertrieb:

Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz
Verlagsunion
Friedrich-Bergius-Straße 20
6200 Wiesbaden
Telefon 0 61 21 / 26 60

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

Erscheinungsweise:

Erstverkaufstag von „Compute mit“ jeweils Mitte des Monats

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 30,- DM im Jahr (12 Ausgaben)
Ausland: Europa 40,- 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.

Anzeigenpreise:

Bitte Mediaunterlagen anfordern.

Anzeigenverwaltung:

Tronic-Verlag GmbH,
Landstraße 29, 3444 Wehretal
Telefon 0 56 51 / 4 06 43 oder 4 06 93
Telefax 0 56 51 / 4 06 85

Anzeigenleitung:

Inland: H. Wendt
Ausland: M. Kleimann

Die große Schau der kleinen Computer

Ein Redaktionsteam von „Compute mit“ war für Sie auf der diesjährigen Mikrocomputermesse '86, die in der Zeit vom 29. Januar bis 2. Februar auf dem Messegelände Frankfurt statt fand. Wir haben uns umgesehen und berichten über die wesentlichsten und interessantesten Bereiche dieser Ausstellung.

Großes Programm an Sonderveranstaltungen

Die Initiatoren der Mikrocomputermesse '86 haben den Schwerpunkt dieser internationalen Fachmesse auf fünf Anwendergruppen gelegt: Handel, Handwerk, Industrie, Gesundheits- und Bildungswesen.

Im Rahmen von täglichen Sonderveranstaltungen wurden Interessierte aus diesen fünf Bereichen in Fachseminaren und Referaten auf den Umgang mit Mikrocomputern vorbereitet und informiert.

Man wollte auf diese Weise versuchen noch mehr Interessierte für das große Gebiet der Datenverarbeitung mit Mikrocomputern zu gewinnen. Der PC sollte nach Meinung der Fachleute in allen Büros seinen Platz haben, da mit dem Stand der heutigen Technik in fast allen Bereichen des Arbeitsprozesses wesentliche Erleichterungen und Vereinfachungen erzielt werden. Der Veranstaltungsplan dieser Messe sah daher seine Hauptaufgabe in der Aufklärung über den sinnvollen Einsatz eines Mikrocomputers.

Keine wesentlichen Neuerscheinungen

Wer diese Messe in der Erwartung besuchte, wesentliche Neuerscheinungen im Bereich der Homecomputerszene zu finden, wurde arg enttäuscht. Zwar gab es einige Neuvorstellungen im Hard- und Softwarebereich, aber der absolute „Mehsehrit“ fehlte gänzlich. Eine Umfrage an den einzelnen Messeständen ergab die

wohl interessantesten Neuigkeiten bei der Fa. Vortex. Hier wurde ein Farbfernseh-Modulator vorgestellt, der allen Schneiderbesitzern erlaubt, ihren Computer an den hauseigenen Farbfernseher anzuschließen und somit auf den Grünmonitor zu verzichten. Die bisher notwendige externe Stromversorgung des Monitors wurde bei diesem Modulator gleich integriert. In der nächsten Zeit plant man bei Vortex alle Schneider-Computer mit einem MS-DOS Modul zu versorgen. Dadurch erhalten auch die Schneider-Besitzer endlich den Zugang zur IBM- (und kompatiblen) Software.

Als weitere Neuerscheinung – hier werden alle Schneider-Floppy-Besitzer angesprochen – wird Ende Februar die F 1 - X (5.25" Diskettenstation) und die M 1 - X (3.5" Diskettenstation) angeboten. Durch Austausch eines EPROM's im Original-Schneider-Controller hat man die Möglichkeit, diese Diskettenstationen auf die bisher erhältlichen F 1 - S und F 1 - D Stationen umzurüsten. Bei einem Verkaufspreis von ca. 700,--DM, dürfte diese eine sehr interessante Alternative zu den Original Diskettenstationen darstellen. Nähere Einzelheiten zu diesem Thema werden wir demnächst veröffentlichen.

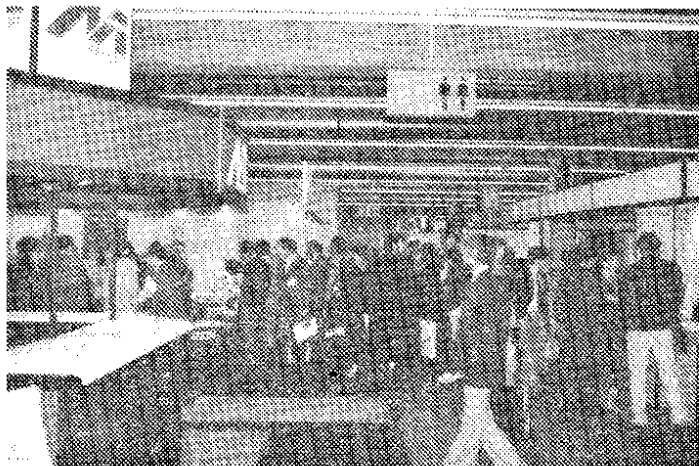
Die größte Ausstellungsfläche für Homecomputer stand in einer Sonderhalle zur Verfügung. Hier hatten alle Besucher

die Gelegenheit Hard- und Software zu testen, wovon auch rege Gebrauch gemacht wurde.

Gleichzeitig war in dieser Halle der WDR-Computerclub und ein Informationsstand des Hessischen Rundfunks zum Thema Computer aufgebaut. Im Rahmen einer Sonderschau hatten einige Aussteller das Thema „Computer und Musik“ dargestellt. Hier wurden die Möglichkeiten gezeigt, wie man mit dem Computer Heimorgeln einsetzen kann, oder wie man mit Hilfe des Joysticks zum Drummer wird. Diese Demonstrationen fanden bei dem Publikum starkes Interesse.

Gelungene Veranstaltung

Durch die Vielzahl der angesprochenen Fachbereiche und durch das große Angebot an Ausstellern kann man von einer gelungenen und vielversprechenden Veranstaltung sprechen. Vielleicht sollten aber noch mehr Hard- und Softwarehersteller die Gelegenheit nutzen, ihre Produkte vorzustellen. Die Gelegenheit dazu bietet sich bei der Microcomputer '87.



PLASTIK FÜR DEN COMPUTER

Staubschutzhauben aus Hartplastik und Druckertische gehören seit einiger Zeit zum Angebot der Firma Media Plast aus Dortmund.

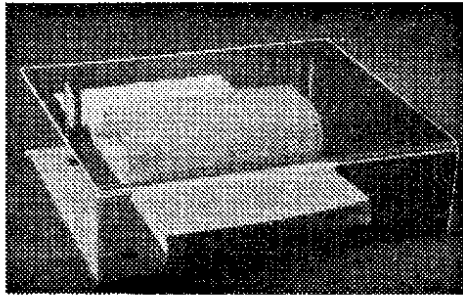
Mit der Problematik verstaubter Tastaturen und Peripheriegeräte konfrontiert, entwickelte die Firma Media Plast eine ganze Serie von Abdeckhauben für die unterschiedlichsten Computertypen und Peripheriegeräte.

Die Abdeckhauben sind aus Hartplastik und im Moment lieferbar für alle drei Schneider CPC-Typen sowie die Schneider 3 Zoll-Laufwerke. Auch für den Commodore-Renner C 64 ist die Plastikabdeckhaube erhältlich. Besonders die Anwender der Commodore-Floppy 1541 werden sich freuen, daß es auch für dieses empfindliche Peripheriegerät entsprechende Abdeckhauben gibt. Weitere gängige Computertypen werden ständig in das Programm aufgenommen.

Durch die neuen paßgenauen Abdeckhauben aus Hartplastik werden Funktionsstörungen durch Verschmutzung nahezu ausgeschlossen. Der paßgenaue Sitz der Abdeckhauben wird durch modernste technische Herstellverfahren mit äußerst präzisen Werkzeugen erreicht.

Aus demselben Material, aus dem die Abdeckhauben hergestellt werden, werden

auch Druckertische für Drucker aller Typen von Media Plast hergestellt und angeboten.



Sehr funktionelle und transparente Aufbauten lassen sich mit diesem neuen Konzept der Druckertische ohne Probleme und ohne großen Platzbedarf erzielen.

Die Druckertische sind so konzeptioniert, daß das nötige Druckerpapier unter dem Drucker ausreichend Platz findet. Mit diesem neuen Druckertisch-Konzept werden Papierstaus weitestgehend ausgeschlossen, und trotz alledem bleibt die gesamte Anlage leicht überschaubar und durchsichtig. Die Druckertische werden in den verschiedensten Ausführungen mit unterschiedlichen Höhen und Breiten geliefert.

Nähere Informationen:
Media Plast
Lübecker Str. 32
4600 Dortmund

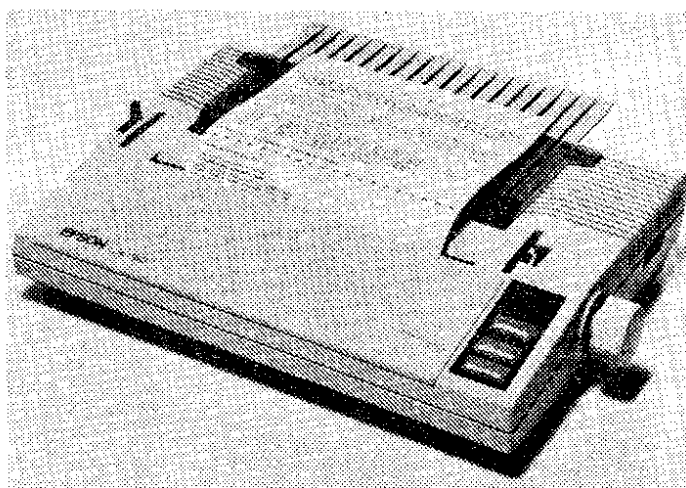
EPSON geht neue Wege im Bereich der Standardisierung von Druckern für Personal- und Homecomputer. Nachdem bereits der neue – EPSON Standard Code for Printers (ESC/P) – im Personalcomputerbereich Wirklichkeit geworden ist, versucht man nun auch im Homecomputerbereich das „Wirrwar“ an Schnittstellen und unterschiedlicher Drucker-Hardware zu entwirren. Auf die Dauer ist ein solches Anpassungsdurcheinander weder den Herstellern von Software noch dem Käufer zuzumuten.

Der Expanded Standard Code for Printers sieht eine einheitliche Belegung der Zeichensätze vor. Er umfaßt den Zeichensatz nach dem PC-Industriestandard, den ASCII-Zeichensatz und nationale Zeichensätze mit Umlauten und Sonderzeichen. Im Homecomputerbereich sind Atari, Commodore, Schneider, Apple, MSX usw. führende Hersteller, die aber leider nicht dem Industriestandard entsprechen. Jeder Hersteller bietet seine eigenen Drucker an, die in der Regel dem Low-cost Niveau der Computer angepaßt sind. Oft müssen bei Qualität und Leistung Abstriche gemacht werden, weil der Drucker die Möglichkeiten des Computers nicht ausnutzt.

Deshalb stellt EPSON mit dem LX-90 Drucker ein neues Konzept vor. Hierbei wird nicht für jeden Computer ein spezieller Drucker entwickelt, sondern man geht von einem Grundmodell aus, das keine Betriebssoftware enthält. Diese wird in neu entwickelte Steckmodule ausgelagert und ist genau auf den betreffenden Computer abgestimmt. Im Steckmodul befindet sich dann auch das komplette Interface. Durch das Einstecken der unterschiedlichen Module wird mit einem Handgriff aus dem einheitlichen Grundmodell ein spezieller Drucker für Commodore, Atari, Apple, MSX oder Schneider. Das ist eine verbraucherfreundliche Lösung, solange sich bei Heimcomputern noch ein einheitlicher Standard herausgebildet hat.

Damit bleibt der LX-90 ein echtes Low-cost Angebot und der Käufer erhält einen Drucker, der für professionelle Beanspru-

LX-90 nach allen Seiten kompatibel



Das neueste Druckermodell LX-90 paßt sich dank einschiebbarer Module den Heimcomputer mit ihren unterschiedlichen Betriebssystemen exakt an. Er bietet professionelle Qualität zum niedrigen Preis.

chung ausgelegt ist und mühelos harten Dauereinsatz verkraftet. Mechanik und Elektronik sind für zusätzliche Leistungen wie Schönschrift und automatischen Einzelblatteinzug ausgelegt, so daß der Druk-

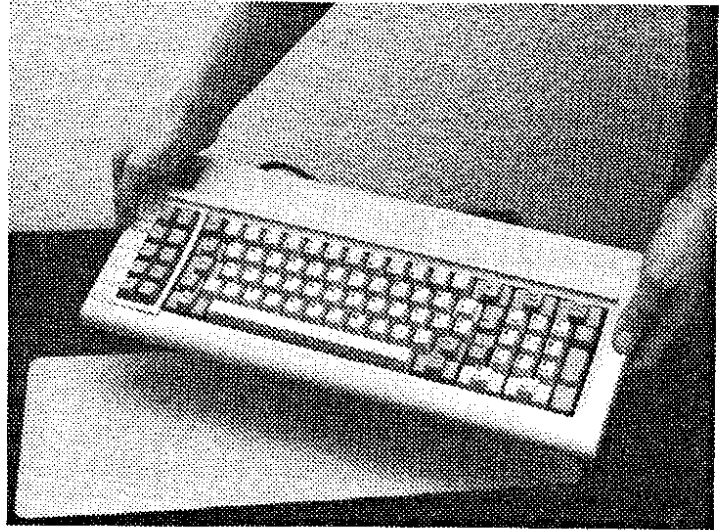
ker auch beim Umstieg auf einen anderen Computer mit entsprechenden professionellen Eigenschaften aufwarten kann. Für den Händler bietet der LX-90 die Annehmlichkeit, daß er anhand eines Grundmo-

dells nur mit Hilfe der Module jeden Heimcomputerbesitzer mit einem qualitativ hochwertigen Drucker versorgen kann, ohne daß er ein ganzes Lager voller verschiedener Drucker führen muß.

Anti-Statik-Matten sorgen für „Entspannung“

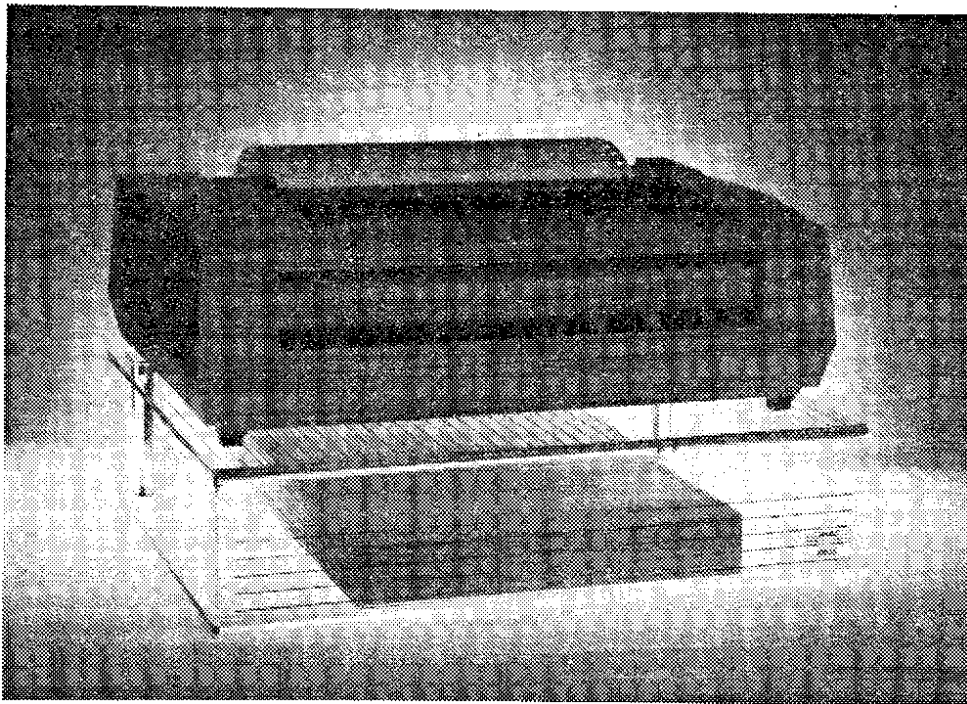
Verlorengegangene oder verfälschte Daten, Übertragungs- und Druckfehler sowie Papierstau haben oft eine einfache, aber folgenschwere Ursache: Mehr als 50 Prozent derartiger Störungen an EDV-Anlagen sind auf elektrostatische Aufladungen zurückzuführen. Aber nicht nur Geräte und Daten sind gefährdet. Den Anwendern drohen meist unangenehme Schläge. Mit den Misco-Anti-Statik-Matten läßt sich hier schnell und wirksam Abhilfe schaffen. Die Tisch- und Tastatur-Matten mit integrierter Erdungsverbindung halten elektrostatische Aufladungen unter Kontrolle und leiten sie gefahrlos ab. Darüber hinaus wirken sie lärmindernd und absorbieren Tonfrequenzen. Beide Ausführungen haben eine hochwertige Schaumpolsterung mit abwaschbarer Plastikoberfläche.

Misco liefert die Tischmatte für Monitor und Tastatur (61 x 66 cm) zum Preis von DM 199,-- (netto). Die reine Tastatur-Matte (48 x 30,5 cm) gibt es für netto DM 89,--. Weitere Anti-Statik-Produkte bietet der kostenlos erhältliche Misco-Katalog bei



Misco EDV-Zubehör GmbH, 6082 Mörfelden-Walldorf

Die platzsparende PBOX für Arbeitsplatzdrucker



Für alle Arbeitsplatzdrucker ist die PBOX der ideale Untersatz. Das Endlospapier liegt, wo es keinen zusätzlichen Platz benötigt, unter dem Drucker.

Zur Ausstattung gehören ein Kabelhalter, eine Meßscala, die den Papierinhalt anzeigt, die Aufnahmemenge beträgt 500 Blatt Endlospapier und innen angebrachte Schutzstreifen, die eine statische Aufladung verhindern.

Die PBOX gibt es in vier Größen und paßt unter über 250 Arbeitsplatzdrucker der verschiedenen Fabrikate.

Das Material ist glasklares Plexiglas. Durch das klassische Design, die hervorragende Qualität der Verarbeitung erfüllt sie höchste Ansprüche und paßt an jeden EDV-Arbeitsplatz.

Der PBOX ist, wie alle DAZU-Produkte, über den Fachhandel erhältlich. Der Preis beginnt ab 142,00 DM (o. MWSt.).


```

1 REM -----
2 REM ---          MINOR-JUMPER          ---
3 REM -----
4 REM ---  DESIGNED BY CARSTEN REINKE  ---
5 REM ---COPYRIGHT(1985) BY EQUUS-SOFT---
6 REM -----

```

2. Teil

```

10 TRAP 1270
20 POKE52,55:POKE56,55
30 FORA=1TO5:NA$(A)=""-----C.R.-----":HP(A)=111:KEYA,CHR$(132+A):NEXT
40 DEFFNA1(YP)=3072+40*YP:DEFFNA2(YP)=2048+40*YP
50 PUDEF"0":VOL2:B=3:L$="01:10"
60 POKE65298,192:POKE65299,56
70 GOSUB280:GOSUB540:LE=1:VE=10
80 XP=21:YP=22:XR=0:YR=0:PU=0:TI$="000000":HL=0
90 COLOR1,8:CHAR1,37,0," " +STR$(LE)
100 POKEFNA1(YP)+XP,29:POKEFNA2(YP)+XP,77
110 IFINT(RND(1)*8)<>2THEN130
120 PO=INT(RND(1)*879)+3152:AL=PEEK(PO):IFAL=27ORAL=29THENPOKEPO,167:POKEPO-1024
,232

```

```

130 J=JOY(1):IFJ>0THENR=J:ELSE190
140 ONRGOTO150,150,160,170,170,170,180,150
150 XR=00:YR=-1:GOTO190

```

```

160 XR=01:YR=00:GOTO190
170 XR=00:YR=01:GOTO190
180 XR=-1:YR=00:GOTO190

```

```

190 XP=XP+XR:YP=YP+YR:P=PEEK(FNA1(YP)+XP)
200 POKEFNA1(YP)+XP,28:POKEFNA2(YP)+XP,123
210 IFP=27THENSOUND2,900,2:GOSUB250:ELSESOUND2,800,2
220 IFP=46ORP=167THEN780:ELSEFORA=1TOVE:NEXT
230 IFP=40THENPU=PU+4:GOSUB250
240 GOSUB260:GOTO100
250 PU=PU+1:CHAR1,23,0," ":PRINTUSING"###";PU:IFPU>199THEN780:ELSERETURN
260 T$=" "+MID$(TI$,3,2)+" "+RIGHT$(TI$,2)
270 CHAR1,8,0,T$:IFT$=L$THENZL=1:GOTO780:ELSERETURN
280 COLOR0,1:COLOR1,3,4:COLOR4,1
290 PRINT"UZEIT :00:00 PUNKTE :000 LEVEL :1  ";:COLOR1,3,4
300 PRINT"^";:FORA=1TO38:PRINTCHR$(95);:NEXT:PRINT"! "
310 FORA=1TO22:PRINTCHR$(34);
320 COLOR1,2:PRINT".....";:COLOR1,3,4:PRINT"#";
:NEXT

```

```

330 PRINT"$XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";:POKE4071,38:RESTORE1230
340 COLOR1,8:CHAR1,13,4," " MINOR-JUMPER ":A=7
350 GOSUB430:CHAR1,12,21," " F1 = ANLEITUNG " "
360 GOSUB430:CHAR1,12,22," " F2 = SPIEL " "
370 GOSUB430:CHAR1,12,23," " F3 = HIGHSCORE " "
380 GETA$:IFA$="" THENA=7:RESTORE1230:GOTO450
390 IFA$="" THENRETURN
400 IFA$="" THENHI=1:GOTO890
410 READF,G:IFF=-1THENRESTORE1230:GOTO410
420 SOUND1,F,G:SOUND2,F+2,G:GOTO350
430 IFA>4THENA=A-1:COLOR1,8,A:ELSEA=7
440 COLOR1,8,A:RETURN
450 COLOR1,8:CHAR1,13,4," " MINOR-JUMPER " "
460 RESTORE1140:FORB=6TO20:READA$:GOSUB430:CHAR1,2,B,A$:NEXT
470 CHAR1,12,21," "....."
480 GOSUB430:CHAR1,12,22," " F2 = SPIEL " "
490 GOSUB430:CHAR1,12,23," " F3 = HIGHSCORE " "
500 GETA$:IFA$="" THENRETURN
510 IFA$="" THENHI=1:GOTO890
520 READF,G:IFF=-1THENRESTORE1230:GOTO520
530 SOUND1,F,G:SOUND2,F+2,G:GOTO480

```

**Hinweis an alle C-16/
116 Kassetten-User**

Wenn Sie Programm-Kassetten der Zeitschrift „Compute mit“ benutzen, beachten Sie bitte, daß die auf Kassette abgespeicherten Programme auf der 2. Seite (B-Seite) abgespeichert wurden. Auf der 1. Seite sind die Programme für die Systeme C-64 und VC-20.


```

1150 DATA"1. AUF DER STRASSE BLEIBT","2. MINDESTENS 200 PUNKTE FRISST","....UND
"
1160 DATA"3. NICHT AUF ORANGE-BLINKENDE FELDER","....KOMMT."
1170 DATA"AUSSERDEM SPIELEN SIE GEGEN DIE ZEIT.(","IST EIN LEVEL DURCH ERFOLG ODE
R"
1180 DATA"MISSERFOLG BEENDET,SO SPRINGT ","DAS PROGRAMM IN DAS NAECHSTE LEVEL,"
1190 DATA"BZW. IN DIE HIGHSCORE-LISTE.(",".....---VIEL ERFOLG---",""
1200 DATA"STEUERUNG: JOYSTICK IN PORT 1"
1210 RESTORE1230:VOL3
1220 READA,B: IFA=-1THEN1210:ELSESOUND1,A,B:SOUND2,A-1,B:SOUND1,1020,4:GOTO1220
1230 DATA685,7,685,10,643,7,596,10,1020,10,739,20,704,10,685,20,1020,10,643,12,5
96,8
1240 DATA643,12,596,8,685,15,643,17,596,15,1020,10,739,20,704,10,685,20,1020,10
1250 DATA643,12,596,8,643,12,596,30,685,7,685,10,643,7,596,10,1020,10,810,20
1260 DATA770,10,739,20,1020,10,685,12,643,8,685,12,643,8,596,30,1020,20,-1,-1
1270 POKE65298,196:POKE65299,208:SCNCLR:COLOR1,1:COLOR0,2:COLOR4,15,6:END

```

Tips & Tricks für den Commodore 16/116

Erweiterung des C-16/116-Befehlssatzes um einen WINDOW-Befehl

Dieser Befehl generiert automatisch ein Fenster, das entweder durch zweimaliges HOME-Drücken oder durch ein neues Fenster gelöscht wird.

Der Basic-Loader muß vor dem Starten abgespeichert werden, da er sich selbst löscht. Das Maschinenprogramm ist 96

Bytes lang und fällt trotz des geringen Speichers kaum ins Gewicht. Als Befehl wird Pi mit der Tokennummer 255 benutzt. Das Programm nutzt die dauernde Kontrolle über die Speicherstellen 2021 - 2024 (07E5 - 07E8), wo die Spalten- und Zeilenkoordinaten des C-16/116 stehen.

2021 07E5 unterer Rand(Zeile)ESC-B
 2022 07E6 oberer Rand(Zeile)ESC-T
 2023 07E7 linker Rand(Spalte)ESC-T
 2024 07E8 rechter Rand(Spalte)ESC-B

```

10 REM *****
20 REM *
30 REM * FENSTERBEFEHL FUER C-16/C-116 *
40 REM *
50 REM *****
60 REM *
70 REM * COPYRIGHT (C) VON :
80 REM * EGGERT ZSCHAU
90 REM *
100 REM * BOYSENKOPPEL 15
110 REM *
120 REM * 2340 KAPPELN
130 REM *
140 REM * TEL.:04642/81615
150 REM *
160 REM *****
170 REM SPEICHER ABGRENZEN
180 POKE53,160:POKE55,160:CLR
190 REM BESCHREIBUNG
200 PRINT"  FENSTERBEFEHL"
210 PRINT"  LR,OR,RR,UR"
220 PRINT"  LR : LINKER RAND"
230 PRINT"  OR : OBERER RAND"
240 PRINT"  RR : RECHTER RAND"
250 PRINT"  UR : UNTERER RAND"
260 PRINT"  NICHT IM DIREKTMODUS!"

```

```

270 REM DATAS EINPOKEN
280 FORI=16288TO16383
290 READA#
300 A=DEC(A#)
310 S=S+A
320 POKEI,A
330 NEXT
340 IFS<>11990THENPRINT"DATAFEHLER":END
350 REM INITIALIEREN
360 SYS16288
370 REM BASIC PROGRAMM LOESCHEN
380 NEW
390 REM MASCHINENPROGRAMM
400 DATAA9,AB,BD,08,03,A9,3F,8D
410 DATA09,03,60,20,73,04,C9,FF
420 DATAF0,03,4C,D9,8B,20,B1,9D
430 DATAE0,28,90,03,4C,1C,99,8E
440 DATAE7,07,86,CA,20,91,94,20
450 DATA84,9D,E0,19,B0,EE,8E,E6
460 DATA07,86,CD,20,91,94,20,84
470 DATA9D,E0,28,B0,DF,EC,E7,07
480 DATA90,DA,8E,EB,07,20,91,94
490 DATA20,84,9D,E0,19,B0,B1,EC
500 DATAE6,07,90,AC,8E,E5,07,20
510 DATA C1,FB,20,79,04,4C,D9,8B

```

DATA-MAKER für den COMMODORE C-16/116

Auf den ersten Blick möchte man den im C-16/116 integrierten Maschinenmonitor als ungeheueren Luxus bezeichnen; doch betrachtet man die Fähigkeiten dieses Monitors genauer, muß man sehr stark einschränken:

er ist sehr sparsam gehalten.

So fehlt die doch eigentlich obligatorische Option, frei wählbare Speicherbereiche in Datazeilen umzuwandeln, sei es, daß man einen Basicloader für ein Maschinenprogramm benötigt oder nur eine kurze Routine in ein Basicprogramm einbinden möchte.

Hier schafft das Programm Data-Maker Abhilfe.

Es ist in der Lage, fast jeden Speicherbereich in Datazeilen umzuwandeln. „Fast“ deshalb, weil sich der Computer durch seine Speicherarchitektur selbst Grenzen setzt. Ein und derselbe Speicherbereich kann natürlich nur einmal belegt sein, und so scheidet der Speicherbereich, der von Data-Maker selbst und dann von den Datazeilen belegt wird, für eine sinnvolle Anwendung in den meisten Fällen aus, da sich das Programm sonst selbst in Datas ablegt, was wohl wenig Sinn hat. Dies bringt jedoch ein kleines Problem mit sich, denn man kann nicht voraussagen, wie lang das Programm mit den Datazeilen im Endeffekt wird, da es ja dynamisch ist und sich selbst modifiziert, folglich auch immer länger wird. Es ist also nicht auf den ersten Blick abzusehen, ob sich Maschinenprogramm (oder was immer in Datazeilen umgewandelt werden soll) und Datas überlagern werden, was ungewünschte Folgen hätte.

Das Problem läßt sich folgendermaßen relativ leicht lösen: Sie sperren einen Teil des Basic-Speicherraums für Basic-Programme und bringen dort ihre Maschinenroutine, Datentabelle oder was sonst in die Datazeilen gepackt werden soll, unter. Dann starten Sie den Data-Maker und können

dann sehen, ob in dem verbleibenden Speicherraum genug Platz für die Datazeilen ist oder nicht. Steigt der Computer mit einem 'OUT OF MEMORY'-Error aus, so haben Sie Pech gehabt und müssen vielleicht auf die Overlay-Technik (Nachladen von Programmteilen) zurückgreifen. Sonst haben sie wahrscheinlich Glück gehabt, wahrscheinlich deshalb, weil unter Umständen die benötigten Variablen nicht mehr in den Speicher passen. Bei umfangreicheren Programmen kann also immer nur ein Testlauf zeigen, ob alles in den Speicher paßt. In den meisten Fällen hat man jedoch erfahrungsgemäß keine Probleme, so schnell lassen sich 16 KByte auch nicht füllen.

Zum Schützen des Speichers hat man zwei Möglichkeiten:

1. Man setzt den Basic-Start hoch.
2. Man setzt das RAM-Ende herunter.

Die erste Möglichkeit hat den Nachteil, daß man den Speicherschutz vor dem Laden des Programms vornehmen muß, vor jedem späteren Benutzen des Programms muß dies also erledigt werden, während man bei der zweiten Möglichkeit diesen Schutz in das Programm einbauen kann, der Anwender dies also nicht selbst zu erledigen braucht.

Zum Schützen geht man folgendermaßen vor:

- man zerlegt die neue Speichergrenze in die Low-/Highbyte-Darstellung:
Highbyte = JNT(Speichergrenze/256)
Lowbyte = Speichergrenze-256*Highbyte
- zum Hochsetzen des Basic-Starts gibt man folgende Zeile ein:
POKE43,Lowbyte:
POKE44,Highbyte:
POKEStartadresse,Ø:NEW

- zum Herabsetzen des RAM-Ende gibt man folgende Zeile ein:
POKE51,Lowbyte:POKE52,Highbyte:
POKE55,Lowbyte:POKE56,Highbyte:
CLR

Nun aber zum eigentlichen Programm. Lassen Sie beim Abtippen die gesamten REM-Zeilen weg; sie sind für den Programmablauf völlig irrelevant und machen das Programm nur länger. Speichern Sie das Programm nach dem Abtippen unbedingt ab, da es sich selbst löscht. Starten Sie das Programm mit RUN.

Sie müssen nun Anfangs- und Endadresse des Speicherbereichs eingeben, den Sie in Datazeilen ablegen wollen. Danach ist das Programm erst einmal eine Weile mit dem Erzeugen der Datazeilen beschäftigt. Daß dabei dauernd etwas auf den Bildschirm geschrieben wird, braucht Sie nicht weiter zu beunruhigen, es ist normal. Wenn der Computer sich mit einem UNDEF'D STATEMENT ERROR (das ist ganz normal) und blinkendem Cursor zurückmeldet, steht der Basic-Loader im Speicher, die Datazeilen beginnen mit der Zeilennummer 60000, die Programmzeilen mit dem Data-Maker sind entfernt und der Basic-Loader enthält sogar einen Einleseteil mit zeilen(!)orientierter Prüfsumme ein Fehler beim eventuellen Abtippen der Datas wird also sehr genau lokalisiert (sollten Sie Ihr Programm veröffentlichen lassen wollen, erleichtert das den Anwendern, die es später abtippen, die Fehlersuche sehr). Zudem ist das Listing sehr übersichtlich, da die Dataswerte als zweistellige Hex-Zahlen dargestellt werden (die Konvertierung in das Dezimalsystem erledigt der Einleseteil beim Einlesen).

Der Basic-Loader kann ganz normal mit SAVE abgespeichert werden, da alle Pointer richtig gesetzt sind.



```

0 REM *****
1 REM *
2 REM *   D A T A - M A K E R   F U E R   D E N   C O M M O D O R E   C 1 6 / 1 1 6   *
3 REM *
4 REM *   E S   W I R D   E I N   B A S I C - L O A D E R   M I T   Z E I L E N O R I E N T I E R T E R   *
5 REM *   P R U E F S U M M E   F U E R   E I N E N   F R E I   W A E H L B A R E N   S P E I C H E R -   *
6 REM *   B E R E I C H   E R Z E U G T !   *
7 REM *
8 REM *   W R I T T E N   1 9 8 5   B Y   W A L D E M A R   R A A Z   *
9 REM *****

```

```

10 REM
11 REM * START- UND ENDADRESSE HOLEN *
12 REM
13 GRAPHICO,1
14 INPUT"Ⓜ STARTADR. ";S:INPUT"Ⓜ ENDADR. ":
E:IFS>=ETHEN13
15 Z=59999
16 REM
17 REM * WINDOW DEFINIEREN *
18 REM
19 GRAPHICO,1:COLOR1,1
20 CHAR1,10,24,"*** EINEN MOMENT ***",1
21 PRINT"Ⓜ";
22 PRINTCHR$(27)"T":CHAR,0,20,"":
PRINTTAB(39)CHR$(27)"B"
23 REM
24 REM * DATAZEILEN ERZEUGEN *
25 REM

```

```

26 Z=Z+1
27 PRINT"Ⓜ"RIGHT$(STR$(Z),5)"DATA";
28 FORI=0TO7
29 W=PEEK(S+(Z-60000)*8+I)
30 P=P+W:PRINTRIGHT$(HEX$(W),2)",, ";
31 NEXT
32 PRINTP:PRINT"S="S":E="E":Z="Z":
GOTO34"
33 POKE1319,19:POKE1320,13:
POKE1321,13:POKE239,3:END
34 IFS+(Z-59999)*8<=ETHEN26
35 REM
36 REM * EINLESE-BLOCK ERZEUGEN *
37 REM
38 Z=Z+1:PRINT"Ⓜ"Z"FORT="
"S"TO"E"STEP8:P=0"

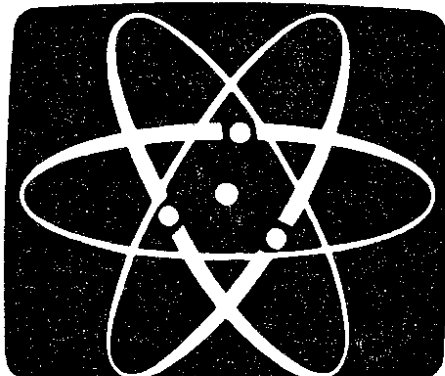
```

```

39 Z=Z+1:PRINTZ"FORI=0TO7:READA$:A=DEC(A$):P=P+A:POKET+I,A:NEXTI"
40 Z=Z+1:PRINTZ"READR:IFP<>RTHENPRINT"CHR$(34)"PRUEFSUMMENFEHLER IN ZEILE";
41 PRINTCHR$(34)"PEEK(63)+PEEK(64)*256:END"
42 Z=Z+1:PRINTZ"NEXT:PRINT"CHR$(34)"DATAS RICHTIG EINGELESEN !!!"CHR$(34)
43 PRINT"GOTO49"
44 POKE1319,19:POKE1320,13:POKE1321,13
45 POKE1322,13:POKE1323,13:POKE1324,13:POKE239,6:END
46 REM
47 REM * DATA-MAKER WIEDER LOESCHEN *
48 REM
49 PRINTCHR$(27)"N":X=0
50 PRINT"Ⓜ"X"ⓂX="X+1":GOTO50":POKE1319,19:POKE1320,13:POKE1321,13:POKE239,3:END

```

Zwei Themen – eine Ausstellung



Hobby-tronic

9. Ausstellung für Funk- und Hobby-Elektronik

COMPUTER- SCHAU

2. Ausstellung
für Computer,
Software
und Zubehör

Dortmund

23.-27. April 1986

Die umfassende Marktübersicht für Hobby-Elektroniker und Computer-Anwender, klar gegliedert:

In Halle 5 das Angebot für CB- und Amateurfunker, Videospiele, DX-er, Radio-, Tonband-, Video- und TV-Amateure, für Elektro-Akustik-Bastler und Elektroniker. Mit dem Actions-Center und Laborversuchen, Experimenten, Demonstrationen und vielen Tips.

In Halle 4 das Super-Angebot für Computer-Anwender in Hobby, Beruf und Ausbildung. Dazu die „Computer-Straße“, als Aktionsbereich, der Wettbewerb „Jugend programmiert“ und der Stand des WDR-Computer-Clubs.

Ausstellungsgelände Westfalenhallen Dortmund täglich 9.00-18.00 Uhr.

Hero für den Commodore 16/116

Sie sind der Flugkapitän eines großen Flugobjektes, dessen Auftrag es ist, das unbekannte Höhlensystem einer riesigen Berglandschaft zu erkunden. Die an Bord befindliche Laserkanone ermöglicht es Ihnen, sich einen Weg durch dieses System

zu bahnen und in die noch nicht erforschten Gebiete dieses Höhlensystems vorzudringen.

Das Labyrinth der Höhlenlandschaft baut sich bei jedem Spiel anders auf, da es von dem Zufallsgenerator gesteuert wird. Sie haben also auf diese Weise ständig andere Bilder und dadurch ein großes Maß an Abwechslung. Die Steuerung des Spiels erfolgt mit dem Joystick.

Riskante Flugmanöver

im unterirdischen Höhlensystem

Ladeanweisungen:

Das Programm besteht aus 2 Teilen. Der erste Teil wird mit DLOAD geladen und mit RUN gestartet. Jetzt wird der neue Zeichensatz und die Daten für die eingebaute Scrollmaschine eingelesen. Durch den Befehl NEW löscht sich das Programm selbst.

Der zweite Teil mit DLOAD und RUN gestartet.

```

2 POKE56,47:CLR:V=65280:POKEV+18,PEEK(V+18)AND251:POKEV+19,PEEK(V+19)AND30R48:PO
KE1344,0
4 FORT=832T0849:READA:POKET,A:NEXT:SYS832:VOLB:COLOR0,1,0:COLOR4,1:COLOR1,2,6
6 FORT=12800T013200:READA:IFA>-1THENPOKET,A:NEXT
8 DATA162,0,189,0,208,157,0,48,189,0,209,157,0,49,202,208,241,96
9 DATA254,254,254,0,239,239,239,0
10 DATA170,85,170,85,170,85,170,85
11 DATA1,24,24,48,16,16,16,16,16,24,24,48,16,48,72,72
12 DATA0,0,0,255,0,0,0,0,255,165,165,165,165,165,255
13 DATA255,170,170,85,85,170,170,85,85,85,170,85,170,85,170,85
14 DATA170,96,96,147,12,21,34,37,9,6,7,233,16,168,100,164,37,38,39,40,16,175,192
,192
15 DATA164,68,164,24,9,246,6,9,16,126,16,16,40,36,34,33
16 DATA24,24,100,130,129,129,129,126,149,80,37,158,93,40,32,137,255,129,129,129,
225,225
17 DATA225,255,40,83,44,68,42,164,48,85,32,32,96,124,96,96,80,72,72
50 DATA-1
75 FORT=32000T032080:READS:POKEI,S:NEXT
76 DATA110,24,50,110,49,50,110,51,50,110,54,50,234,110,58,50,110,60,50,110,61,50
77 DATA110,62,50,234,234,110,16,50,234,234
81 DATA110,40,50,110,41,50,110,42,50,110,43,50,110
82 DATA44,50,110,45,50,110,46,50,110,47,50,110,124,50,110,125,50,110,126,50,110,
127,50
83 DATA110,123,50,110,122,50,110,121,50,110,120,50,96
110 DO:READA$:IFA$="TS"THENEXIT:ELSEPOKE891+X,DEC(A$):X=X+1:LOOP
230 DATAA5,DF,38,E9,01,85,DB,A5
240 DATAE0,E9,00,85,D9,A5,DB,18
250 DATA69,01,85,DA,A5,D9,69,00
260 DATA85,DB,A9,00,85,DC,A4,DE
270 DATAB1,DB,91,DA,88,CO,00,DO
280 DATAF7,E6,DC,A5,DC,CS,DD,DO
290 DATA01,60,A5,DB,18,69,28,85
300 DATAD8,A5,D9,69,00,85,D9,A5
310 DATADA,18,69,28,85,DA,A5,DB
320 DATA69,00,85,DB,18,90,CF,TS
321 NEW

320 DIMA$(32):HI=100:HI$="XARGON"
330 RESTORE500:F0RF=1T032:READA$(F):NEXT:RU=0:PU=0:J=0:TR=0:X0=0:LE=5:RT=0
340 VOLB:GOSUB12000:SP=3072
400 IFLE<1THEN330
410 COLOR1,2,6:VOLB:SP=3072
420 POKE221,25
430 POKE222,39
440 POKE223,SP-INT(SP/256)*256:POKE224,INT(SP/256)
470 C=66:GOSUB10000:B=3300:POKEB,66:D=0:A=0:TRAP5000
480 SYS32000
    
```

```

481 IFJOY(1)>127THENGOSUB8000
491 IFJOY(1)=1THENA=-80:GOSUB3000:SOUND1,99,1
492 SYS32000
493 IFJOY(1)=3THENA=-40:SOUND1,44,3:GOTO496
494 IFJOY(1)=7THENA=-1:GOSUB3000:SOUND1,44,3
495 A=A+40:IFPEEK(B-1+A)=32THENGOSUB3000
496 IFPEEK(B+A)>65THENS000
497 IFPEEK(B+A)=32THENPOKEB,32:POKEB-1024,97:B=B+A
498 A=0:POKEB,C:POKEB-1024,124
500 GOTO480

```

```

522 DATA"  "
523 DATA"  "
524 DATA"  "
525 DATA"  "
526 DATA"  "
527 DATA"  "
528 DATA"  "
529 DATA"  "
530 DATA"  "
531 DATA"  "
532 DATA"  "
533 DATA"  "
534 DATA"  "
535 DATA"  "
536 DATA"  "
537 DATA"  "
538 DATA"  "
539 DATA"  "
540 DATA"  "
541 DATA"  "
542 DATA"  "
543 DATA"  "
544 DATA"  "
545 DATA"  "
546 DATA"  "
547 DATA"  "
548 DATA"  "
549 DATA"  "
550 DATA"  "
551 DATA"  "
552 DATA"  "
553 DATA"  "

```

```

1000 RETURN
2000 IFPEEK(B-80)=32THENPOKEB,32:B=B-80:POKEB,D:GOTO492
2010 GOTO5000
3000 IFPEEK(B-1)<>32THEN4000
3001 POKEB,32:POKEB-1024,97
3002 SYS891:POKEB,67:POKEB-1024,124:D=INT(RND(1)*32+1):J=J+1:CHAR1,0,2,"":PRINTA
$(D)
3003 IFJ=25THENRU=RU+1:J=0:CHAR1,0,INT(RND(1)*13+4),"\\":CHAR1,0,14,"ZONE":PRINT
RU""
3004 IFRU-RT=10THENRT=RT+10:GOSUB13000
3005 RETURN
4000 IFPEEK(B-1)=64THENRETURN
5000 FORI=8TO1STEP-1:POKEB,76:SOUND3,1000,9:POKEB,67:SOUND3,456,9:VOLI:SYS32000:
NEXT
5005 VOLB:FORI=1TO20
5010 IFPEEK(B+40)=32THENPOKEB,32:POKEB-1024,97 :B=B+40:POKEB,67:POKEB-1024,124
5011 IFPEEK(B+40)=32THENSOUND1,I*50,3:SOUND2,I*49,3:NEXT
5020 POKEB,78:FORI=8TO1STEP-1:SOUND3,1000,3:SOUND3,444,3:VOLI:NEXT:VOLB
5028 FORI= 4TO18:CHAR1,18,I,"TTTTTTTTTTTTTTTTTT":NEXT

```

programme

```
5029 FORI=5TO17:CHAR1,19,I,"                ":NEXT
5030 CHAR1,20,6,"CHAMBERS":PRINTRU:LE=LE-1
5031 CHAR1,20,8,"XORKS ":":PRINTXO
5032 CHAR1,20,12,"METERS":":PRINTRU*25+J
5033 PU=RU*25+TR*50+J+XO*20:CHAR1,20,14,"POINTS":":PRINTPU:POKEB-1024,97
5034 CHAR1,20,10,"TEKKLES":PRINTTR:IFPU>HITHENHI=PU:HI$=N$
5035 CHAR1,20,16,"LIVES ":":PRINTLE:GOSUB11100:C=66:A=0:IFLE<1THEN330
5040 FORI=4TO18:CHAR1,18,I,"                ":NEXT:B=3500:FORI=B+38TOB+42:POKEI
,64
5041 NEXT:GOTO480
8000 SOUND3,1000,2:SOUND3,444,2:FORI=B-1TOB-10STEP-1
8001 IFPEEK(I)=32THEN:POKEI,68:POKEI,32:NEXT
8004 IFPEEK(I)=77GOTO9000
8005 IFPEEK(I)=81THENSOUND2,800,5:SOUND1,770,10:      XO=XO+1
8006 IFPEEK(I)<>65ANDPEEK(I)<>32THEN8008
8007 RETURN
8008 IFPEEK(I-40)=81THENSOUND2,810,10:SOUND1,798,7:POKEI-40,32:XO=XO+1
8009 POKEI,78:FORZ=8TO1STEP-2:SOUND3,643,2 :VOLZ:NEXT:POKEI,32:VOLB
8010 POKEI,32:RETURN
9000 FORO=I-41TOI-39:POKEO,78:POKEO+40,78:POKEO+80,78:SOUND3,0/45,3:SOUND3,445,3
:NEXT
9010 FORO=I-41TOI-39:POKEO,32:POKEO+40,32:POKEO+80,32:SOUND3,0/45,3:SOUND3,445,3
:NEXT
9020 TR=TR+1:RETURN
10000 SCNCLR
10001 FORI=0TO1:COLOR1,2,4+I:CHAR1,0,I,"_____
":NEXT
10002 FORI=22TO24:COLOR1,2,27-I:CHAR1,0,I,"_____
"
10023 NEXT:COLOR1,3,4:CHAR1,3,23,"HERO--C.COLOSSUS-SOFT"
10030 COLOR1,2,6:FORI=10TO21:CHAR1,20,I,"_____":NEXT:CHAR1,20,9,"- - - - -"
10999 COLOR1,2,6: RETURN
11100 RESTORE11103:FORE=1TO17:READX,Y:FORI=1TOBSTEP4
11101 VOLI:SOUND1,X,Y:SOUND2,X+5,Y:NEXT:SOUND2,X,Y:NEXT
11103 DATA516,10,571, 5,596, 5,571,10,162, 5,453, 5,516,10,345,20,453, 5,596,10
11104 DATA685,10,704, 5,685, 5,453,20,643, 5,453, 5,596,10
11111 RETURN
12000 SCNCLR
12002 COLOR1,6,6
12003 CHAR1,6,2,"- - - - -":COLOR1,6,5:CHAR1,6,3,"- - - - -"
12004 COLOR1,6,4
12005 CHAR1,6,4,"- - - - -":COLOR1,6,3:CHAR1,6,5,"- - - - -"
12006 COLOR1,6,2:CHAR1,6,6,"- - - - -"
12010 CHAR1,4,10,"C.BY COLOSSUS-SOFT 1985":CHAR1,6,12,"HIGH BY ":PRINTI$ "HI
12042 COLOR1,8,1:CHAR1,8,16," 0- 99 NOVICE":COLOR1,8,2:CHAR1,8,17,"100- 199 BE
GINNER"
12043 COLOR1,8,3:CHAR1,8,18,"200- 499 SEMI ":COLOR1,8,4:CHAR1,8,19,"500- 999 NE
WCOMER"
12044 COLDR1,8,5:CHAR1,7,20,"1000-1999 EXPERT":COLOR1,8,6:CHAR1,7,21,"2000-4999
PROFI "
12045 COLOR1,8,7:CHAR1,7,22,"5000-      H E R O"
12048 GOSUB11100
12049 COLOR1,8,1:CHAR1,4,14,"YOUR NAME":INPUTN$
12050 GETA$:IFA$=" "THENRETURN
12100 GOTO12050
13000 FORI=5TO10:CHAR1,0,I,"TTTTTTTT":SOUND2,118,20:SOUND1,118+I,10:NEXT
13100 CHAR1,1,7,"BONUS"
13104 COLOR1,3,4:CHAR1,1,23,"250 METER-ZONE":COLOR1,2,6
13105 BO=TR*XO*10:PU=PU+BO
13110 FORI=1TOBO:CHAR1,1,8,"":PRINTI:NEXT
13999 RETURN
```


Bewerbung als Programmator

Name: _____ Vorname: _____ Alter: _____

Straße: _____ Wohnort: _____

Titel des Programmes: _____ Computersystem: _____

Erforderlicher Speicherplatz: _____ KBytes

Erforderliche Pheriperie: _____

Ich versichere hiermit, daß ich der Autor des oben genannten Programmes bin und alle Rechte besitze. Ich bin damit einverstanden, daß Sie mein Programm in einer Zeitschrift abdrucken und/oder dieses verkaufen. Das einmalige Honorar beträgt 120 DM pro abgedruckte Seite. Sollte das Programm nicht veröffentlicht werden, bitte ich um Rücksendung meiner Unterlagen.

Folgende Unterlagen liegen diesem Schreiben bei:

Kassette: _____ Diskette: _____ Dokumentation: _____ Listing: _____ Rückporto: _____

Unterschrift des Programmators: _____

Einsenden an: Tronic-Verlag Abt. Programmentwicklung, Landstraße 29, 3444 Wehretal 1



Auftrag für Gelegenheitsanzeigen in „Compute mit ...“

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

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)
Eine Veröffentlichung erfolgt **nur** gegen Vorkasse!

Kleinanzeigen jetzt noch preiswerter

Name und Adresse _____

Unterschrift _____ Datum _____

Ich wünsche folgenden Text zu veröffentlichen:

suche Software suche Hardware Tausch Versch.
 biete an Software biete an Hardware Kontakte Chiffre

Zutreffendes ankreuzen

- suche Software suche Hardware Tausch Versch.
- biete an Software biete an Hardware Kontakte Chiffre

★ Kleinanzeigen ★ Kleinanzeigen ★ Kleinanzeigen ★

Verschiedenes

C-16/116, plus 4

Commodore Sachbuchreihe, Band 6.
ROM-Listing (Deutsch)
Erstklassig Dokumentiert
DM 49 + Nachnahme
und Verpackung
Karl-Heinz Pflaum
Schönblick 17,
53550 Euskirchen 27

BÜCHER für alle Computertypen können Sie bei uns bestellen. Wir liefern auch jeden in dieser Zeitschrift genannten Titel. Bücher Becks Postfach 10 4250 Bocholt

Wohnt nebenan ein CPC-Profi?
Bin Einsteiger (54) und suche Kontakt u. Erfahrungsaustausch. Claus Piotrowski, Am Sandershof 31, 4350 Recklinghausen. Ruft doch mal an. Tel. 023 61/49 88 68.

● ● ● **Hallo C-64 und VC-20 Besitzer** ● ● ●
● Falls Ihr keine Lust auf ●
● langweiliges, zeitaufwen- ●
● diges Abtippen von Pro- ●
● grammen habt, kann ich ●
● Euch nur sagen: kein Pro- ●
● blem. ●
● Für 0,02 DM = 2 Pf pro ●
● Zeile tippe ich jedes Pro- ●
● gramm für VC-20 + C-64. ●
● Arndt-Holger Kiessling, ●
● Vierlingstraße 64, 6710 ●
● ● ● Frankenthal/Pfalz. ● ● ●

Biete Software

☆ **VC-20-Besitzer verkauft:** ☆
8K-Ram-Erweiterung für 60,--
DM und Spiele-Cassetten z. B.
Defender oder Ufo. Adressen-
Register, Black-Jack-1744 für
10,-- DM und Joystick für 15,--
DM. Adresse: Martin Stahmer,
Friedrich-Ebert-Damm 26, 2000
Hamburg 70
☆☆☆☆ **ACHTUNG** ☆☆☆☆

Vokabelstar C-16/C-116

leichte Bedienung durch
Menue - deutsche + franzö-
sische Zeichen-Abfrage
nach dem bewährten Kartei-
kartensystem - gute Voka-
beleditieroutinen vorhan-
den. Gegen Einsendung von
20 DM erhalten Sie Kassette
+ Bedienungsanleitung:
M. Suer, Walgernweg 22,
4410 Warendorf 1

■ **Verkaufe meine VC-20 Pro-** ■
■ **gramme:** GV-Robothouse ■
■ (Action) 5 DM, Geisthaus ■
■ (Action) 6 DM, Schatz der ■
■ Unterwelt + (Grafikadvent.) ■
■ 7 DM. 8K-Welltentor (Text- ■
■ adr.) 8 DM, 16K-Advent.: ■
■ zentrax I 16 DM, Kaptain ■
■ Kidd 12 DM, TV 20 8 DM. ■
■ Beta I 8 DM. Für Porto + Kas- ■
■ sette 4 DM zum Preis. Keine ■
■ Disklief. möglich! M. Platt, ■
■ Carl-Orff-Str. 4, D-8721 ■
■ Hambach. ■
■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■

★ **CPC 464 Vokabeltrainer!** ★
☆ Das Programm enthält ☆
★ viele Vokabeln und über ★
☆ 100 unregelmäßige Ver- ☆
★ ben. Sie können noch be- ★
☆ liebige viele Vokabeln ein- ☆
★ geben und abspeichern. ★
☆ Programm auf Kassette ☆
★ **nur DM 19,-.** Zu bestellen ★
☆ bei: Thilo Schray; Alten- ☆
★ bachweg 8/1, 7120 Bietig- ☆
☆ heim-Bissingen. ☆
★★★★★★★★★★★★★★

SCHNEIDER CPC-464/664 ☆
☆ **COMMODORE 64** ★★
☆☆ **COMMODORE VC-20** ★★
☆☆☆ **COMMODORE 16/116** ★★
☆☆☆☆ **COMMODORE PLUS 4** ★★
Riesige Auswahl an Soft- und
Hardware vorhanden.
Infos bei: A. Bachler, Postfach
429, 4290 Bocholt

★★★ **C-16** ☆☆☆ **C-116** ☆☆☆
selbstgeschriebene Spielpro-
gramme für C-16/116. Info bei
Wolfgang Dunczewski, Schmidt-
bornstr. 18, 6230 Frankfurt 80.
80-Pf.-Rückumschlag beilegen!

Commodore 64, VC-20, C-16/116, 4+6 Spiele, Denktraining, Lotto, Foto-, Film-, Vide-, Musikarchiv, Fibu, EK-VK, Kfz-Paket, Bibla, Karteikasten, Adress, Rechengenie, Tabello, Digitalo, Rechengenie, Handbücher ab 9,80 DM.
Katalog gegen 80-Pf-Briefmarke.
T. Hafstedes Computerservices,
An der Windmühle 8, 5010 Bergheim 5

THE PROFESSIONAL CLOCK
1.2 Super-Uhr für CPC 464, 100%
MC Abweichung 5s/24H, Alar-
meinricht. Großanzeige, RSX-
Befehle, Progr. verschiebbar,
Schaltuhr möglich. Cassette für
DM 12,- incl. Vers. (Scheck,
Briefm.) Georg Huonker, Erleng-
bachhof, 7463 Rosenfeld 6

★ **Lohn- u. Einkommensteuer** ★
Suer Jahresausgleich, Steuer-
kl.-wahl, Monatslohnst.: Cass.
60,- Disk. 70,-/75,- DM. Jährl.
Aktualisierung ohne Neukauf.
Info gegen Porto für C-64/128 u.
Schneider CPC 464/664/6128.
Horst Ilchen, Niederfelderstr. 44,
8072 Manching, ☎ 08459/1669

**Technisch-wissenschaftliche
Programme,** sowie Lehrpro-
gramme in Mathe- und Fremd-
sprachen für Commodore-VC-20
sucht Hans-J. Bobak, Kapellen-
str. 2, 6277 Bad-Camberg, Tel.
06434-8819.

VC-20 PROGRAMME JV: Privat.
Textadventure Schloßzauber +
Psychaterspiel - beides
Deutsch - zusammen auf Cas-
sette 10,-- DM.
Volker Coert, Söllingstr. 4-8
4300 Essen 1
Komplette Liste gegen Rück-
porto! Resettaster 5,- DM VC-20
Lexikon. Reparaturen von Tasta-
turen (Mech). Suche Spiele +
Programme bis 16K.

C-16, C-116, plus/4 Superspiele!
Info 1 und 2 gegen Rückporto bei
Hannes Kaltenbach, Prielmayer-
straße 16, 7990 Friedrichshafen 1

C-16/116 C-16/116 C-16/116
Achtung! Keine Programme? Bei
mir gibt es eine Auswahlens-
dung mit 5 Topspielen für nur 10
DM. Bei Interesse bitte überwei-
sen auf folgendes Konto: Konto-
nummer: 3603768, BLZ:
41450075 Sparkasse Soest.
Bestellungen bitte an: Roland
Fuchs, Blanckenagelweg 9,
4770 Soest. PS. Info liegt bei!
Auch CPC Programme.

★ **Supersoftware beim All-C-C** ★
Brunos Boxing, S-Games I+II, H.
Games I+II, Broad-Street, Print
Shop, Airwolf, The Rocky Horror
Picture Show, Karateka, Explod-
ing, Elite, Franky goest to Holly-
wood u.v.a. Info mit Liste beim
„All-C-C“ auch Tausch (bitte
Liste beilegen). Alles an Andreas
Neufeld, Rovenkampstr. 1, 4460
Nordhorn. All-Commodore-Club.

**Umfangr. Hausverwaltung für
CPC 464-6128,** 3"-Disk. DM
150,-, Kassette DM 140,-. Info
anfordern, Tel. 02129/8103.

Commodore 16/116: Wir haben
die Programme, die Sie suchen!
Software schon ab 1 DM! Infor-
mationen kommen kostenlos von:
A. Bachler, Postfach 429, 4290
Bocholt. Außerdem: Das Überra-
schungspaket mit 5 Top-Spielen
für nur 15,- DM oder die Insider-
Mappe für 6,- DM. Bezahlung per
Schein oder Scheck.

★★★ **COMMODORE 64** ★☆☆
49 Disketten voller Software!!!
Free Soft: Public Dawain Soft-
ware von versch. User-groups
aus den USA. Hunderte von Pro-
grammen: Utilities, Textverarbei-
tung, Lernprogramme, Spiele. Je
Disk DM 10,- alle 49 DM 400,-.
Kopiergebühr incl. Disk+Porto.
Liste 0,80 Marken. Bellingrath,
Trift 10, 5860 Iserlohn.

VC-20 Programme, Video-
Spiele für Grundversion bis 16K-
Modul, Angebotsliste gegen
frankierten Rückumschlag bei
Horst Querfurt, Postfach
843042, 4320 Hattingen 14.

Neu: Sonderzeichen werden durch Klartext ersetzt!

Eingabe-Hinweise für Commodore C-64-Benutzer!

Um Ihnen die Eingabe unserer Listings zu erleichtern, werden zukünftig alle C-64-Listings ohne Steuerzeichen abgedruckt. Alle Steuerzeichen werden in unseren LISTINGS durch Klartext, welcher die Taste kennzeichnet, ersetzt. Eine Tabelle der Tastenfunktionen finden Sie in jedem Tronic-Magazin.

Tastenbezeichnungen für Steuercodes

(TRONIC-NORM)

DOWN	CURSOR UNTEN	TASTE NEBEN RECHTEM SHIFT	RED	ROT	CONTROL-TASTE & 3
UP	CURSOR HOCH	SHIFT-TASTE & TASTE NEBEN RECHTEM SHIFT	CYAN	TUERKIS	CONTROL-TASTE & 4
CLR	CLEAR SCHIRM	SHIFT-TASTE & 2. TASTE GANZ RECHTS OBEN	PURPLE	PURPUR	CONTROL-TASTE & 5
INST	EINFUEGEN	SHIFT-TASTE & TASTE GANZ RECHTS OBEN	GREEN	GRUEN	CONTROL-TASTE & 6
HOME	CURSOR IN ECKE	2. TASTE VON GANZ RECHTS OBEN	BLUE	BLAU	CONTROL-TASTE & 7
DEL	DELETE	TASTE GANZ RECHTS OBEN	YELLOW	GELB	CONTROL-TASTE & 8
RIGHT	CURSOR RECHTS	TASTE GANZ RECHTS UNTEN	RVSON	INVERSE EIN	CONTROL-TASTE & 9
LEFT	CURSOR LINKS	SHIFT-TASTE & TASTE UNTEN RECHTS	RVSOFF	INVERSE AUS	CONTROL-TASTE & 0
SPACE	LEERZEICHEN	LEERTASTE (GROESSTE TASTE)	ORANGE	ORANGE	COMMODORE-TASTE & 1
F1	FUNKTIONSTASTE	FUNKTIONSTASTE F1	BROWN	BRAUN	COMMODORE-TASTE & 2
F3	FUNKTIONSTASTE	FUNKTIONSTASTE F3	LIG.RED	HELLROT	COMMODORE-TASTE & 3
F5	FUNKTIONSTASTE	FUNKTIONSTASTE F5	DGREY	DUNKELGRAU	COMMODORE-TASTE & 4
F7	FUNKTIONSTASTE	FUNKTIONSTASTE F7	MGREY	MITTELGRAU	COMMODORE-TASTE & 5
F2	FUNKTIONSTASTE	FUNKTIONSTASTE F2	LIG.GREEN	HELLGRUEN	COMMODORE-TASTE & 6
F4	FUNKTIONSTASTE	FUNKTIONSTASTE F4	LIG.BLUE	HELLBLAU	COMMODORE-TASTE & 7
F6	FUNKTIONSTASTE	FUNKTIONSTASTE F6	HGREY	HELLGRAU	COMMODORE-TASTE & 8
F8	FUNKTIONSTASTE	FUNKTIONSTASTE F8	CTRL	CONTROL	CONTROL-TASTE ZUSAMMEN MIT DEM NACHFOLGENDEN ZEICHEN. (Z.B. CTRLA ENTSPRICHT CTRL & A)
BLACK	SCHWARZ	CONTROL-TASTE & 1			
WHITE	WEISS	CONTROL-TASTE & 2			

Wichtig: Folgende Zeichen erhalten Sie auf den Tasten: " ^ " = Pfeil nach oben (Potenzierungstaste ↑) . " \ " = Pfundzeichentaste (€)
 " _ " = Pfeil nach links (Taste ganz links oben)

Folgendes Beispiel demonstriert die Arbeitsweise unseres Systems:

STANDARD AUSDRUCK:	
10 A\$="TEST":PRINT" <u>LEUTE</u> HALLO <u>LEUTE</u> !":PRINT"TRONIC VERLAG"	
20 PRINT" <u>ENDE</u> "	
UNSER NEUER AUSDRUCK DER GLEICHEN ZEILEN:	
10 A\$="TEST":PRINT"<CLEAR RIGHT3 DOWN3 R ED>HALLO<RIGHT2 BLUE>LEUTE<SPACE>!":PRINT"TRONIC<SPACE>VERLAG"	<34>
20 PRINT"<CYAN PURPLE GREEN>ENDE"	<117>

Wie in dem Beispiel zu erkennen ist, wird das Herz-Symbol durch das Wort CLEAR ersetzt. Dies bedeutet, Sie müssen die Taste CLEAR drücken, um das Herz-Symbol zu erzeugen. Sollten Sie einmal nicht wissen, welche Taste gemeint ist, hilft Ihnen ein Blick in unsere Tastentabelle bestimmt weiter.

Um im Listing Tastenbezeichnungen von Grafik oder normalen Texten zu unterscheiden, werden alle Tastenkennzeichnungen in geschweifte Klammern gesetzt. Diese dürfen selbstverständlich nicht eingegeben werden. Auch Leerzeichen innerhalb geschweifeter Klammern dienen nur zur Trennung einzelner Tastenfunktionen und dürfen ebenfalls nicht eingegeben werden. Um die Tastenfunktionen noch besser hervorzuheben, werden diese in unterstrichener Kursivschrift (Schrägschrift) abgedruckt.

Steht hinter einer Tastenfunktion eine Zahl, welche ebenfalls unterstrichen ist, bedeutet dies, daß die letzte Tastenfunktion mehrmals betätigt werden muß. Die in unserem Beispiel abgedruckte Funktion RIGHT3 bedeutet, daß die Taste RIGHT (Cursor rechts) 3 mal hintereinander betätigt werden muß. Auch einzelne oder mehrere Leer-

zeichen innerhalb von Anführungszeichen werden auf diese Art gekennzeichnet. Das bislang übliche Abzählen einzelner Zeichen entfällt somit völlig. Alle Zeichen außerhalb von den geschweiften Klammern werden normal abgedruckt und auch eingegeben. Auf den ersten Blick hört sich das sicher etwas kompli-

ziert an, ist jedoch in der Praxis ganz einfach. Wenn man sich erst an die in Klartext geschriebenen Steuerzeichen gewöhnt hat, wird man den großen Vorteil dieser Schreibweise erkennen.

Checksummer C Version 1.0 für Commodore 64

Checksummer C 1.0 ist ein Prüfsummen-Programm, was die Eingabe von Programm-Listings zum wahren Vergnügen macht. Tippfehler werden schon während des Eingabens der einzelnen Programmzeilen erkannt. Dieses System zusammen mit dem neuen Druckverfahren gewährleisten das unsere Listings meist fehlerfrei abgedruckt werden.

Arbeitsweise und Aufbau unseres Checksummers:

Unser Checksummer besteht aus einem kleinen Maschinenprogramm welches als Basic-Loader abgedruckt ist. Tippen Sie diesen Loader ein und speichern ihn auf Kassette oder Diskette, denn Sie können ihn zukünftig immer wieder benutzen.

Der Start erfolgt durch den Befehl „RUN“. Nach kurzer Zeit meldet sich der Rechner mit der Meldung „TRONIC

...“. Der Checksummer ist nun aktiv und man kann ein beliebiges Tronic-Listing eingeben. Nachdem eine Zeile mit RETURN abgeschlossen wird, erscheint links oben auf dem Schirm eine Prüfzahl. Vergleichen Sie diese mit der Zahl im Heft hinter der Zeile. Stimmt die Zahl überein, ist die Zeile richtig eingegeben.

Auf diese Weise können Sie das gesamte Listing schnell und fehlerfrei eingeben.

Interessant ist auch, daß bei der Eingabe von Zeilen die üblichen Abkürzungen benutzt werden können, ohne die Checksumme zu verändern. Leerzeichen außerhalb von Anführungszeichen werden ignoriert, da diese auf die Ausführung der einzelnen Befehle keinen Einfluß haben. Sie können Ihr Programm auch starten, denn der Check-

summer und Ihr Programm beeinflussen sich nicht gegenseitig. Wollen sie den Checksummer abschalten, geben Sie einfach POKE 1,55 ein oder betätigen Sie die Tasten-Kombination RUNSTOP und RESTORE. Aktivieren können Sie den Checksummer jederzeit (auch nach RESET) mit POKE 1,53.

Wer den Checksummer nicht eingeben möchte kann diesen auch unter der folgenden Bestellnummer beziehen:
 Bestellnummer:
 CV10K Kassette 10 DM
 CV10D Diskette 15 DM

0 REM ***** TRONIC CHECKSUMMER *****	<116>	15 DATA73,67,45,86,69,82,76,65,71,32,67,	
1 REM VERSION C 1.0 BEI FRANK BRALL	<10>	72,69,67,75,83,85,77	<129>
2 REM -----	<14B>	16 DATA77,69,82,32,35,35,35,35,139,13,13	
3 FOR I=832 TO 1008:READ A:S=S+A	<68>	,32,32,32,32,32,32,32	<147>
4 POKE I,A:NEXT I:IF S<>16397 THEN PRINT		17 DATA32,32,32,32,32,86,69,82,83,73,79,	
"FEHLER{SPACE}IN{SPACE}DATA-ZEILEN{SPACE	<32>	78,32,67,32,49,46,48	<79>
}!{SPACE}(10-19)":STOP		18 DATA13,13,32,32,32,32,32,32,40,67,41,	
5 SYS 832:S=0:FOR I=58464 TO 58603:READ	<21>	32,70,82,65,78,75,32	<94>
A:S=S+A		19 DATA66,82,65,76,76,32,40,49,48,46,56,	
6 FOKE I,A:NEXT I:IF S<>18919 THEN PRINT	<219>	53,41,13,0	<12>
"FEHLER{SPACE}IN{SPACE}DATA-ZEILEN{SPACE	<33>	29 DATA169,0,141,107,191,141,108,191,141	<7>
}!{SPACE}(29-36)":STOP		,109,191,160,2,24,177,95,200,113	
7 S=0:FOR I=48979 TO 49005:READ A:S=S+A	<23>	30 DATA95,141,109,191,160,3,200,177,95,2	<91>
8 POKE I,A:NEXT I:IF S<>2888 THEN PRINT"	<26>	40,44,201,34,208,10,173,107,191	
FEHLER{SPACE}IN{SPACE}DATA-ZEILEN{SPACE		31 DATA73,1,141,107,191,177,95,174,107,1	<128>
!{SPACE}(46-47)":STOP	<176>	91,208,4,201,32,240,228,238,108	
9 PRINT" {DOWN2}AKTIVIEREN{SPACE}: {SPACE}	<147>	32 DATA191,174,108,191,24,177,95,109,109	<244>
POKE{SPACE}1,53{SPACE}8}AUSSCHALTEN: {SPA	<225>	,191,141,109,191,202,208,244,76,120	
CE}POKE{SPACE}1,55":NEW	<140>	33 DATA228,56,76,183,228,72,32,201,255,1	<101>
10 DATA169,0,133,254,162,1,189,137,3,133	<158>	70,104,144,1,138,96,32,240,255	<89>
,255,160,0,177,254,145,254,136		34 DATA142,105,191,140,106,191,162,0,160	<239>
11 DATA208,249,230,255,165,255,221,139,3		,0,24,32,240,255,169,91,32,12	<202>
,208,238,202,16,230,169,96,141,49		35 DATA225,169,0,174,109,191,32,205,189,	<213>
12 DATA165,169,228,141,50,165,169,53,133		169,93,76,83,191,173,33,208,145	<65>
,1,169,141,133,254,162,3,134,255		36 DATA243,96,105,2,164,145,200,208,4,19	
13 DATA160,0,177,254,240,7,32,202,241,20		7,161,208,247,96	
0,76,120,3,169,2,141,32,208		46 DATA32,12,225,32,63,171,32,63,171,172	
14 DATA96,160,224,192,0,147,17,32,32,35,		,106,191,174,105,191,24,32,240	
35,35,35,32,84,82,79,78		47 DATA255,76,128,164,24,0,0,5,200	

Dieses Programm ermöglicht, eine Befehlsenerweiterung des Commodore 64 um insgesamt 28 neue Befehle. C-64 Basic übersichtlich und anwenderfreundlich erweitert, damit das Programmieren noch einfacher wird.

Befehlsenerweiterung für den C-64

Da der C-64 ein recht schwaches BASIC hat, entschloß sich der Autor eine Erweiterung zu schreiben, die es ermöglicht, vor allem den sehr guten Tongenerator auf einfache Weise anzusprechen. Weiterhin integriert dieses Programm noch einige Programmierhilfen zur Veränderung der Bild- und Schriftfarben usw. Insgesamt wurden 28 neue Befehle definiert, die alle fest als Token in das Betriebssystem eingebunden wurden. Das Maschinenprogramm ist genau 1533 Bytes lang und liegt zwischen 49152 und 50685. Die Erweiterung funktioniert ohne weiteres mit anderen Maschinenprogrammen zusammen, die nicht denselben Speicherbereich

gebrauchen oder, die die Zeiger für die Befehlsausführung verändern. Die einzelnen Routinen im Programm wurden sehr einfach gehalten und sind deshalb sehr schnell.

Das Programm kann zu jeder Zeit mit Sys 50500 gestartet werden.

Nachdem man das Maschinenprogramm erfolgreich abgetippt und die Prüfsummenroutine keine Fehler gefunden hat, fragt der Computer, ob das Programm abgespeichert werden soll. Diese Frage kann man entsprechend mit „j“ oder „n“ beantworten. Danach fragt der Computer, ob das Programm gestartet werden soll. Nachdem die zweite Frage mit ja beant-

wortet wurde, sieht der Bildschirm wie folgt aus:

BETTERBASIC V2.0
38911 BASICBYTES FREE
(C)1985 BY MARTIN DE
HOOGH READY

Erst dann stehen die neuen Befehle zur Verfügung. Ein im Speicher befindliches Programm wird bei der Initialisierung „normalerweise“ nicht verändert. Alle neuen Befehle sind fest in das Betriebssystem integriert und ein RUNSTOP/RESTORE kann ihnen nichts anhaben. Aber nun zu den Befehlen im einzelnen:

programme

RENEW:

Ein Befehl, der jedem bekannt sein dürfte. Ein versehentlich mit NEW gelöscht Programm wird wieder auf Vordermann gebracht.

PRINT AT:

Mit diesem Befehl kann man Texte an beliebiger Stelle auf den Bildschirm bringen. Wichtig ist, daß der Befehl vollständig ausgeschrieben wird.

VOLUME:

Volume setzt die Lautstärke für den Tongenerator.

FWAVE:

Mit diesem Befehl werden die Werte für Attack/decay und Sustain/release eingestellt. Der dritte bestimmt die Wellenform für den ersten Tongenerator.

SWAVE:

Wie vorher, nur für den zweiten Tongenerator.

TWAVE:

Wie vorher, nur für den dritten Tongenerator.

FPULSE:

Stellt die Pulsweite für den ersten Tongenerator ein.

SPULSE:

Wie vorher, nur für den zweiten Tongenerator.

TPULSE:

Wie vorher, nur für den dritten Tongenerator.

SYSTEM:

Dieser Befehl führt einen Kaltstart des Computers aus.

FPLAY:

FPLAY setzt Low- und Highbytes der Tonfrequenz des ersten Tongenerators. Dasselbe gilt natürlich, wie man schon richtig vermutet, für die beiden Befehle SPLAY und TPLAY.

CLS:

Mit CLS wird der Bildschirm gelöscht. Dadurch spart man sich den PRINT-Befehl.

SCREEN:

Es wird die Rand- und Hintergrundfarbe gesetzt. Die beiden Pokes werden dadurch überflüssig.

INK:

Verändert die Schriftfarbe.

REPEAT:

Hiermit wird die Dauerfunktion aller Tasten angeschaltet.

UNREPEAT:

Schaltet die Dauerfunktion der Tasten wieder aus.

EXECUTE:

Dieses ist ein besonderer Befehl, mit ihm können Befehle an das BASIC-Programm übergeben werden.

Achtung: Man sollte diesen Befehl nicht ohne Parameter gebrauchen, da der Computer dadurch aus dem Programm aussteigen könnte.

SRESET:

Es wird ein Reset des SID's ausgelöst.

SPEED:

Mit diesem Befehl wird die Geschwindigkeit des Interrupts festgelegt.

SUSR:

Setzt Low- und Highbyte für die USR-Funktion.

RES:

Auch dieser Befehl ist für die Tonprogrammierung da. Es werden die verschiedenen Filter angeschaltet und es wird die Resonanz festgelegt.

KEYBEEP:

Beim Programmieren ist es immer sehr nützlich, wenn man nicht immer auf den Bildschirm schauen muß, um zu erkennen ob ein Tastendruck registriert wurde oder nicht. Dieser Befehl sorgt für ein akustisches Signal, wenn eine Taste gedrückt wurde.

BEEPOFF:

Dieser Befehl macht den Befehl KEYBEEP wieder rückgängig.

NORM:

Falls einem die Befehlsweiterung auf die Nerven gehen sollte, braucht er nur den Befehl NORM einzugeben, und die Erweiterung wird ausgeschaltet.

MULTICOL:

Dieser Befehl dient dazu, den Mehrfarbenmodus des C-64 anzuschalten. Der erste Wert setzt die Farbe für die Hintergrundfarbe 2. Die mehrfarbigen Zeichen erreicht man über die SHIFT-Taste. Grafiken können in diesem Modus leider nicht mehr dargestellt werden.

Der zweite Wert bestimmt die Farbe der Zeichen im Reversmodus. Diese Farbe ist dann gleichzeitig die Cursorfarbe. Die Schriftfarbe wird weiterhin mit INK festgelegt.

MULTIOFF:

Mit diesem Befehl wird der Mehrfarbenmodus wieder ausgeschaltet.

```

100 REM ***** <39>
105 REM * <202>
110 REM * BETTERBASIC V2.0 * <32>
115 REM * <212>
120 REM *(C)1985 BY MARTIN DE HOOGH* <123>
125 REM * <222>
130 REM *BERGERW. 14 6370 KIRN/NAHE* <171>
140 REM * <237>
150 REM * TEL. 06752-2744 * <254>
160 REM * <1>
170 REM ***** <109>
180 : <238>
190 M=128:O=173741 <237>
200 PRINT " (CLEAR) ":FOR I=49152 TO 50684 <65>
210 READ A:POKE I,A:IFA=OTHEN N=N+1 <13>
215 Z=PEEK(63)+256*PEEK(64):P=P+A <124>
220 PRINT " (HOME) ZEILE "Z" (SPACE) GELESENE <93>
R (SPACE) WERT (SPACE) "A
230 PRINT "ANZ. NULLEN (SPACE) "N" (SPACE) PR <172>
UEFS. "P

```

```

240 PRINT "MOMENTANE (SPACE) ADRESSE " I <221>
250 NEXT:PRINT " (CLEAR) " <238>
260 IF N<>M THEN PRINT " (DOWN) DU (SPACE) HAST <164>
"N" NULLEN (SPACE) IM (SPACE) PROGRAMM":PRINT
"RICHTIG (SPACE) WAERE "M
270 IF N<>M THEN PRINT "DU (SPACE) HAST "N-M" NUL <211>
LEN (SPACE) ZUVIEL "
275 IF N<>M THEN PRINT "DU (SPACE) HAST "M-N" NUL <133>
LEN (SPACE) ZU WENIG "
280 IF P<>O THEN PRINT " (DOWN) LEIDER (SPACE) <53>
IST (SPACE) DIE (SPACE) PRUEFSUMME (SPACE) FAL
SCH "
285 IF P<>O THEN PRINT "DU (SPACE) HAST "P",RIC <154>
HTIG (SPACE) IST "O
390 IF P<>O THEN PRINT "SIE (SPACE) IST (SPACE) U <141>
M "O-P" ZU (SPACE) KLEIN "
395 IF P<>O THEN PRINT "SIE (SPACE) IST (SPACE) U <148>
M "P-O" ZU (SPACE) GROSS "
396 IF M<>N OR P<>O THEN END <121>
400 IF P=O AND N=M THEN PRINT "ALLES (SPACE) IN <188>
(SPACE) ORDNUNG (SPACE) !!! "

```

```

410 IFN=MANDP=0THENPRINT {DOWN}ABSPEICHE
RN{SPACE}J/N":GOTO420
420 BETA$: IFA$=""THEN420
430 IFA$="J"THENINPUT"NAME":B$:SAVEB$
440 PRINT"PROGRAMM{SPACE}STARTEN{SPACE2}
J/N"
450 BETA$: IFA$=""THEN450
460 IFA$="J"THENSYS50500
470 END
900 :
910 :
920 :
930 :
940 :
980 REM *****
985 REM *****MASCHINENPROGRAMM*****
990 REM ** STARTEN MIT SYS 50500 **
991 REM *****
995 :
996 :
997 :
1000 DATA 079,195,141,195,121,192,225
1010 DATA 252,164,195,171,195,196,195
1020 DATA 221,195,246,195,006,196,022
1030 DATA 196,067,229,038,196,054,196
1040 DATA 061,196,067,196,073,196,122
1050 DATA 196,219,196,226,196,012,197
1060 DATA 028,197,160,197,176,197,192
1070 DATA 197,208,197,221,197,242,197
1080 DATA 182,163,182,163,182,163,182
1090 DATA 163,182,163,182,163,182,163
1100 DATA 182,163,182,163,182,163,182
1110 DATA 163,182,163,182,163,182,163
1120 DATA 182,163,182,163,182,163,182
1130 DATA 163,182,163,182,163,182,163
1140 DATA 182,163,182,163,252,192,128
1150 DATA 192,187,192,120,162,005,189
1160 DATA 102,192,157,004,003,202,016
1170 DATA 247,088,096,120,032,083,228
1180 DATA 088,096,016,050,201,255,240
1190 DATA 046,036,015,048,042,201,204
1200 DATA 048,042,056,233,203,170,132
1210 DATA 073,160,255,234,202,240,010
1220 DATA 234,200,185,205,193,016,250
1230 DATA 048,244,234,200,185,205,193
1240 DATA 048,006,032,071,171,208,245
1250 DATA 234,076,239,166,234,076,243
1260 DATA 166,234,076,036,167,032,115
1270 DATA 000,032,196,192,076,174,167
1280 DATA 240,012,233,128,144,012,201
1290 DATA 035,176,012,076,247,167,234
1300 DATA 076,043,168,234,076,165,169
1310 DATA 234,201,075,048,006,208,008
1320 DATA 076,018,168,234,076,008,175
1330 DATA 234,201,127,240,248,056,233
1340 DATA 076,010,168,185,001,192,072
1350 DATA 185,000,192,072,076,115,000
1360 DATA 166,122,160,004,132,015,189
1370 DATA 000,002,016,007,201,255,240
1380 DATA 065,232,208,244,201,032,240
1390 DATA 058,133,008,201,034,240,089
1400 DATA 036,015,112,048,201,063,208
1410 DATA 004,169,153,208,040,201,048
1420 DATA 144,004,201,060,144,032,076
1430 DATA 150,193,132,113,160,000,132
1440 DATA 011,136,134,122,202,200,232
1450 DATA 189,000,002,056,249,158,160
1460 DATA 240,245,201,128,208,048,005
1470 DATA 011,164,113,232,200,153,251
1480 DATA 001,185,251,001,240,054,056
1490 DATA 233,058,240,004,201,073,208
1500 DATA 002,133,015,056,233,085,208
1510 DATA 156,133,008,189,000,002,240
1520 DATA 223,197,008,240,219,200,153
1530 DATA 251,001,232,208,240,166,122
1540 DATA 230,011,200,185,157,160,016
1550 DATA 250,185,158,160,208,180,189
1560 DATA 000,002,016,190,153,253,001
1570 DATA 198,123,169,255,133,122,096
1580 DATA 132,113,160,000,132,011,136
1590 DATA 134,122,202,200,232,189,000
1600 DATA 002,056,249,205,193,240,245
1610 DATA 201,128,208,008,005,011,024
1620 DATA 105,076,076,074,193,166,122
1630 DATA 230,011,200,185,204,193,016
1640 DATA 250,185,205,193,208,220,164
1650 DATA 113,166,122,076,047,193,082
1660 DATA 069,078,069,215,080,082,073
1670 DATA 078,084,032,065,212,078,079
1680 DATA 082,205,083,089,083,084,069
1690 DATA 205,086,079,076,085,077,197
1700 DATA 070,087,065,086,197,083,087
1710 DATA 065,086,197,084,087,065,086
1720 DATA 197,070,080,076,065,217,083
1730 DATA 080,076,065,217,084,080,076
1740 DATA 065,217,067,076,211,083,067
1750 DATA 082,069,069,206,073,078,203
1760 DATA 082,069,080,069,065,212,085
1770 DATA 078,082,069,080,069,065,212
1780 DATA 069,088,069,067,085,084,197
1790 DATA 075,069,089,066,069,069,208
1800 DATA 083,080,069,069,196,083,082
1810 DATA 069,083,069,212,083,085,083
1820 DATA 210,082,069,211,070,080,085
1830 DATA 076,083,197,083,080,085,076
1840 DATA 083,197,084,080,085,076,083
1850 DATA 197,066,069,069,080,079,070
1860 DATA 198,077,085,076,084,073,067
1870 DATA 079,204,077,085,076,084,073
1880 DATA 079,070,198,000,000,000,000
1890 DATA 000,000,000,000,000,000,000
1900 DATA 000,000,000,000,000,000,000
1910 DATA 000,000,000,000,000,000,000
1920 DATA 014,255,111,111,114,118,119
1930 DATA 200,000,000,000,000,000,000
1940 DATA 001,002,003,004,000,000,255
1950 DATA 255,141,169,133,141,000,049
1960 DATA 119,230,240,245,121,000,000
1970 DATA 000,000,000,000,000,000,203
1980 DATA 096,096,096,070,051,105,133
1990 DATA 017,004,009,121,234,070,141
2000 DATA 151,000,000,000,000,000,000
2010 DATA 000,000,000,000,000,000,000
2020 DATA 169,133,141,000,090,013,079
2030 DATA 080,095,000,000,000,000,000
2040 DATA 000,000,000,000,000,000,000
2050 DATA 000,000,000,000,000,000,000
2060 DATA 121,113,114,155,160,169,017
2070 DATA 076,065,066,067,068,069,070
2080 DATA 071,077,111,254,254,254,254
2090 DATA 230,023,076,032,012,080,094
2100 DATA 000,000,000,000,000,000,000
2110 DATA 000,000,000,000,000,000,000
2120 DATA 130,124,177,196,133,164,155
2130 DATA 112,201,224,226,218,229,149
2140 DATA 245,065,129,251,145,219,000
2150 DATA 145,133,169,181,132,115,117
2160 DATA 087,094,027,065,231,222,222
2170 DATA 000,242,141,232,000,001,008
2180 DATA 040,032,019,212,201,133,169
2190 DATA 014,016,056,000,000,000,000
2200 DATA 064,064,051,089,045,011,010
2210 DATA 000,165,043,164,044,133,034
2220 DATA 132,035,160,003,200,177,034
2230 DATA 208,251,200,152,024,101,034
2240 DATA 160,000,145,043,165,035,105
2250 DATA 000,200,145,043,136,162,003
2260 DATA 230,034,208,002,230,035,177
2270 DATA 034,208,244,202,208,243,165
2280 DATA 034,105,002,133,045,165,035
2290 DATA 105,000,133,046,076,099,166
2300 DATA 032,158,183,138,072,032,253
2310 DATA 174,032,158,183,104,168,024
2320 DATA 032,240,255,032,253,174,076
2330 DATA 164,170,032,158,183,142,024
2340 DATA 212,096,032,158,183,142,005
2350 DATA 212,032,253,174,032,158,183
2360 DATA 142,006,212,032,253,174,032

```


programme

2370 DATA 158,183,142,004,212,096,032 <178>
 2380 DATA 158,183,142,012,212,032,253 <71>
 2390 DATA 174,032,158,183,142,013,212 <121>
 2400 DATA 032,253,174,032,158,183,142 <133>
 2410 DATA 011,212,096,032,158,183,142 <133>
 2420 DATA 019,212,032,253,174,032,158 <127>
 2430 DATA 183,142,020,212,032,253,174 <218>
 2440 DATA 032,158,183,142,018,212,096 <199>
 2450 DATA 032,158,183,142,000,212,032 <12>
 2460 DATA 253,174,032,158,183,142,001 <220>
 2470 DATA 212,096,032,158,183,142,007 <150>
 2480 DATA 212,032,253,174,032,158,183 <14>
 2490 DATA 142,008,212,096,032,158,183 <50>
 2500 DATA 142,014,212,032,253,174,032 <248>
 2510 DATA 158,183,142,015,212,096,032 <93>
 2520 DATA 158,183,142,032,208,032,253 <86>
 2530 DATA 174,032,158,183,142,033,208 <192>
 2540 DATA 096,032,158,183,142,134,002 <40>
 2550 DATA 096,169,128,141,138,002,096 <69>
 2560 DATA 169,000,141,138,002,096,169 <137>
 2570 DATA 254,133,057,133,058,234,234 <48>
 2580 DATA 234,032,158,173,032,130,183 <181>
 2590 DATA 170,160,000,177,034,153,000 <187>
 2600 DATA 002,200,202,208,247,169,000 <141>
 2610 DATA 153,000,002,169,000,133,122 <159>
 2620 DATA 169,002,133,123,008,104,104 <228>
 2630 DATA 104,104,104,076,150,164,120 <87>
 2640 DATA 162,136,160,196,142,020,003 <21>
 2650 DATA 140,021,003,088,096,165,203 <114>
 2660 DATA 201,064,208,008,169,203,141 <13>
 2670 DATA 219,196,076,049,234,165,203 <209>
 2680 DATA 205,219,196,208,003,076,049 <38>
 2690 DATA 234,169,015,141,024,212,169 <110>
 2700 DATA 003,141,005,212,169,242,141 <240>
 2710 DATA 006,212,169,026,141,001,212 <45>
 2720 DATA 169,005,141,000,212,169,033 <211>
 2730 DATA 141,004,212,160,069,162,200 <186>
 2740 DATA 202,208,253,136,208,248,169 <239>
 2750 DATA 000,141,004,212,141,005,212 <233>
 2760 DATA 141,006,212,165,203,141,219 <67>
 2770 DATA 196,076,049,234,203,032,158 <244>

2780 DATA 183,142,101,220,096,169,000 <52>
 2790 DATA 141,024,212,141,004,212,141 <49>
 2800 DATA 005,212,141,006,212,141,011 <254>
 2810 DATA 212,141,012,212,141,013,212 <14>
 2820 DATA 141,018,212,141,019,212,141 <224>
 2830 DATA 020,212,141,023,212,141,027 <192>
 2840 DATA 212,141,028,212,096,032,158 <65>
 2850 DATA 183,142,017,003,032,253,174 <174>
 2860 DATA 032,158,183,142,018,003,096 <66>
 2870 DATA 032,158,183,142,023,212,096 <38>
 2880 DATA 004,021,003,011,032,045,032 <125>
 2890 DATA 002,001,019,009,003,013,040 <126>
 2900 DATA 003,041,032,002,025,032,004 <98>
 2910 DATA 021,003,011,045,019,015,006 <55>
 2920 DATA 020,032,032,032,162,000,189 <201>
 2930 DATA 089,197,032,210,255,169,001 <32>
 2940 DATA 157,000,216,232,224,070,208 <222>
 2950 DATA 240,076,108,192,147,005,066 <216>
 2960 DATA 069,084,084,069,082,066,065 <174>
 2970 DATA 083,073,067,032,086,050,046 <151>
 2980 DATA 048,032,032,032,051,056,057 <82>
 2990 DATA 049,049,032,066,065,083,073 <50>
 3000 DATA 067,066,089,084,069,083,032 <129>
 3010 DATA 070,082,069,069,013,040,067 <222>
 3020 DATA 041,049,057,056,053,032,066 <230>
 3030 DATA 089,032,077,065,082,084,073 <129>
 3040 DATA 078,032,068,069,032,072,079 <199>
 3050 DATA 079,071,072,013,000,000,032 <159>
 3060 DATA 158,183,142,002,212,032,253 <226>
 3070 DATA 174,032,158,183,142,003,212 <12>
 3080 DATA 096,032,158,183,142,009,212 <179>
 3090 DATA 032,253,174,032,158,183,142 <58>
 3100 DATA 010,212,096,032,158,183,142 <54>
 3110 DATA 016,212,032,253,174,032,158 <40>
 3120 DATA 183,142,017,212,096,120,169 <104>
 3130 DATA 049,141,020,003,169,234,141 <156>
 3140 DATA 021,003,088,096,032,158,183 <12>
 3150 DATA 142,034,208,032,253,174,032 <208>
 3160 DATA 158,183,142,035,208,169,091 <12>
 3170 DATA 141,017,208,096,169,027,141 <213>
 3180 DATA 017,208,096,000,000,000,000 <153>

Werkstatt: Text-Scroller und Text-Editor für Commodore 64

Das Programm erzeugt eine „zuckfreie“, softscrollende Laufschrift und läuft in allen möglichen Zeichensätzen. Es ist natürlich nicht auf Buchstaben beschränkt, d.h. auch Grafikzeichen können verwendet werden.

Außerdem läuft das Programm im Interrupt, bedeutet, nach Aufruf kann im Basic-Programm weitergemacht werden (z.B. Datas einlesen)!!

Die Zeichen werden ab der Speicherstelle 24576 (\$6000) in ihren Bildschirmcodes (Handbuch Seite 133) abgerufen und auf den Bildschirm gebracht. Man muß also seinen Text ab der Speicherstelle 24576 hintereinander einpoken.

Man kann dies z.B. mit dem Mini-Text-Editor tun. Mit diesem Programm schreibt man einfach den Text auf den Bildschirm

und drückt „F1“. Den Rest erledigt das Programm. Der Mini-Text-Editor ist natürlich nicht vollkommen, aber hier kann ja jeder seine eigene Lösung finden.

Nun muß die Laufschrift auch, wenn der Text zu Ende ist, wieder von vorne beginnen. Dies geschieht folgendermaßen: Findet das Programm ab der Speicherstelle 24576 den Code 127 (Commodore+B), schreibt es ein Leerzeichen und beginnt den Text von vorne. Bevor man also beim Mini-Text-Editor „F1“ drückt sollte man den Text mit Commodore+B abschließen! Auf Drücken von „F1“ stoppt die Laufschrift.

Achtung!!! Nach Starten der Laufschrift wird der Bildschirm ein wenig verkleinert, um Buchstaben an den Rändern herein-

kommen und verschwinden zu lassen. Daher empfiehlt sich ein einfarbiger Bildschirm, da der Effekt hier nicht sichtbar ist.

Nun aber zum Wichtigsten: Gestartet wird das Prg. mit SYS 28672 (\$7000), und die Laufschrift hat immer die selbe Farbe, wie die aktuelle Zeichenfarbe. Hat man nun einen Text geschrieben, kann man mit POKE45,240:POKE45,112:CLR den Text und das Maschinenprogramm und sogar einen wohlmöglich vorhandenen neuen Zeichensatz abspeichern. Man spart sich nach dem Laden des Programmes das Einglesen der Datas da alle Speicherstellen bis 28912 abgespeichert wurden. Wundern sie sich deshalb auch nicht über die längere Lade-, Speicherzeit!

Listing Text-Scroller:

```

0 DATA169,7,141,22,208,169,0,133,251,133
,252,169,0,162,96,133,253,134,254 <80>
1 DATA141,234,3,173,134,2,162,0,157,192,
219,232,224,39,208,248,120,169 <237>
2 DATA66,160,112,141,20,3,140,21,3,169,2
42,141,18,208,173,17,208,41,127 <40>
3 DATA141,17,208,169,129,141,26,208,88,9
6,173,25,208,141,25,208,48,7,173 <131>
4 DATA13,220,88,76,49,234,165,203,201,4,
240,18,173,18,208,201,250,176,36 <62>
5 DATA32,202,112,169,250,141,18,208,76,1
29,234,120,173,17,208,41,127,141 <223>
6 DATA17,208,169,240,141,26,208,169,49,1
62,234,141,20,3,142,21,3,88,165 <68>
7 DATA252,208,3,76,189,112,162,0,189,193
,7,157,192,7,232,224,38,208,245 <251>
8 DATA172,234,3,177,253,201,127,240,14,1
41,230,7,200,140,234,3,208,21,230 <217>
9 DATA254,76,189,112,169,0,162,96,133,25
3,141,234,3,134,254,169,32,141 <1>
10 DATA230,7,169,241,141,18,208,169,7,14
1,22,208,76,129,234,169,0,133,252 <171>

```

```

11 DATA165,251,201,1,208,2,230,252,170,2
02,138,41,7,133,251,141,22,208,96 <126>
12 FORT=0T0224:READD:POKE7*4096+T,D:A=A+
D:NEXT <48>
13 IFA<>30115THENPRINT"DATA{SPACE}ERROR" <142>
14 END <142>
15 : <73>
16 : TEXT-SCROLLER (C) BY LOTUS PRODUC
TIONS <226>
17 : EDZARD WITTIG / VERDEN <139>
18 : <76>
19 : <77>

```

Listing Text-Editor

```

10 REM ***** MINI-TEXT-EDITOR ZUM TE
XT-SCROLLER ***** <191>
20 BETA$: IFA$="{SPACE}"THEN40 <84>
30 PRINTA$;:BOTO20 <199>
40 FORT=0T0999:POKE6*4096+T,PEEK(1024+T) <33>
50 IFPEEK(1024+T)=127THENEND <42>
60 NEXT <190>

```

Tips & Tricks für den Commodore 64

„SPRITE RUN“

Viele BASIC-Programme werden durch die Joystickabfrage und die Spritesteuerung sehr langsam. Um da Abhilfe zu schaffen, hat der Autor eine kleine Maschinenhilfsroutine geschrieben, die per Interrupt läuft und somit keinen Geschwindigkeitsverlust bei Ihrem Programm zur Folge hat.

Das Programm belegt den Speicherplatz über 49152 (c000), verbraucht also keinen BASICspeicher.

Vor dem Starten legen Sie im Block 13 (832-896) die Daten für den zu bewegenden Sprite ab. Anschließend poken Sie, welche Sprites zu erscheinen haben (also hier POKE 53248+21,4).

Danach können Sie noch die Position des Sprites bestimmen.

Sonst ist die Position x=0 Y=0.

Nun können Sie Sprite Run mit einer FOR..NEXT Schleife einpoken lassen und es mit SYS 49152 aufrufen.

Ihr BASIC-Programm läuft gleich nach dem SYS-Aufruf weiter, nur das jetzt der gewünschte Sprite über den ganzen Bildschirm bewegt werden kann (also von x=0 bis x=320).

Sie können den Sprite in 8 verschiedene Richtungen steuern. Verschwindet der Sprite am rechten Rand, taucht er am linken wieder auf.

Ausschalten läßt sich das Ganze, indem

Sie einfach Sprite Nr. 4 ausschalten, also POKE 53248+21,0. Der Autor versuchte das Programm so allgemein wie möglich zu halten, so daß es in jedem BASIC-Programm eingesetzt werden kann.

●	10 FOR T=49152 TO 49378	<221>	●
	20 :READ Q	<79>	
	30 :POKE T,Q	<199>	
	40 NEXT	<170>	
	50 END	<178>	
●	60 DATA120,169, 23,160,192,141, 20, 3,14		●
	0, 21, 3, 88,169, 13,141,250, 7	<90>	
	70 DATA169, 0,141,236,194, 96,173, 0,220		
	,201,119,240, 31,201,123,240, 36	<15>	
●	80 DATA201,125,240, 47,201,126,240, 37,2		●
	01,118,240, 45,201,117,240, 53,201	<233>	
	90 DATA121,240, 61,201,122,240, 69, 76,		
	49,234, 32,139,192,238, 4,208, 76	<215>	
●	100 DATA 49,234, 32,139,192,206, 4,208,		●
	76, 49,234,206, 5,208, 76, 49,234	<1>	
	110 DATA238, 5,208, 76, 49,234, 32,139,1		
	92,238, 4,208,206, 5,208, 76, 49	<5>	
●	120 DATA234, 32,139,192,238, 4,208,238,		●
	5,208, 76, 49,234, 32,139,192,206	<203>	
	130 DATA 4,208,238, 5,208, 76, 49,234, 3		
	2,139,192,206, 4,208,206, 5,208	<133>	
●	140 DATA 76, 49,234,173,236,194,201, 1,2		●
	40, 24,173, 4,208,240, 59,201,255	<213>	
	150 DATA240, 1, 96,169, 4,141, 16,208,16		
	9, 1,141,236,194,238, 4,208, 96	<165>	
●	160 DATA173, 4,208,201, 0,240, 5,201, 90		●
	,240, 15, 96,169, 0,141, 16,208	<28>	
	170 DATA141,236,194,169,255,141, 4,208,		
	96,169, 0,141, 16,208,141,236,194	<127>	
●	180 DATA169, 0,141, 4,208, 96,169, 4,141		●
	, 16,208,169, 1,141,236,194,169	<252>	
	190 DATA 89,141, 4,208, 96,255	<92>	

FUNKTIONSTASTEN des Commodore 64

Für viele erscheint die Nutzung der Funktionstasten am C-64 rätselhaft. Mit einem einfachen Programm kann man diese vier Tasten achtmal, mit ein paar Tricks kann man sie sogar 32 mal belegen. Nach dem Motto „in der Kürze liegt die Würze“ wurden Beispielprogramme gestaltet und Tips und Tricks vorgeführt. Fehlende Befehlsätze in Basic 2 sind selbst entwickelt, die dazu notwendige Boolesche Algebra eingehend behandelt.

Jeder hat wohl am Anfang mit den rätselhaften Funktionstasten kritische Auseinandersetzungen gehabt. Beim Betätigen erzeugen sie weder einen Abdruck auf dem Bildschirm noch lösen sie irgendeine

Steuerungstätigkeit aus. Im Hochkommando-Modus geben sie aber auf dem Bildschirm inverse Zeichen wieder. Im Handbuch gibt es auch nur den Hinweis: diese Tasten können im Rahmen eines Programms be-

legt werden.

In der folgenden kurzen tabellarischen Darstellung sind die wichtigsten Merkmale der Funktionstasten dargelegt:

Tabelle-1

Tasten	ungeshiftet		geschiftet oder zusammen mit Commodore-Taste		Belegung der Speicherplatz 203
	ASCII	Zeichen	ASCII	Zeichen	
F1/F2	133	␣	137	␣	4
F3/F4	134	␣	138	␣	5
F5/F6	135	␣	139	␣	6
F7/F8	136	␣	140	␣	3

Bitte beachten Sie, daß die Belegung des Speicherplatzes 203 (Tastencodenummer) nicht ganz im Einklang mit der Reihenordnung der Tastaturbenennung steht. Einfachste Methode der Funktionstastenbelegung ist, eine bedingte Verzweigung des Programmablaufes programmatisch aufzufassen. Ein Beispiel: Mit GetA\$ wird die Tastatureingabe geholt und mit dem If..Then-Befehl das Programm an die gewünschte Zeile abgezweigt, in der Ihr Teilprogramm beginnt. Unser Beispiel:

```

10 geta$:if a$="" then 10
20 if a$="␣" then 110
30 if a$="␣" then 120
40 if a$="␣" then 130
50 if a$="␣" then 140
60 if a$="␣" then 150
70 if a$="␣" then 160
80 if a$="␣" then 170
90 if a$="␣" then 180
100 goto 10
110 print"funktionstaste 1":goto 10
120 print"funktionstaste 3":goto 10
130 print"funktionstaste 5":goto 10
140 print"funktionstaste 7":goto 10
150 print"funktionstaste 2":goto 10
160 print"funktionstaste 4":goto 10
170 print"funktionstaste 6":goto 10
180 print"funktionstaste 8":goto 10
    
```

Eigentlich wäre das Alles, was man zur Belegung einer Taste braucht.

Wir gestalten das Programm mit einem Trick viel eleganter. Dazu einige Tips zur Stringverwaltung.

Als erstes erzeugen wir einen INSTRING-Ersatz Befehl, der im Commodore Basic 2 fehlt.

Der Programmablauf ist, daß ein Teilstring in einem Mutterstring gesucht wird. Ist es gefunden, wird der Stellenwert des Teilstringes wiedergegeben.

In unserem Fall ist A\$ der Teilstring, der mit Tastendruck erzeugt wurde. Mutterstring wäre, wenn wir ihn IN\$ nennen,



Unser Programm nimmt dann die folgende Form an.

```

10 in$=" "
20 geta$:forx=1tolen(in$):ifa$=mid$(in$,x,1)then40
30 next:goto 20
40 on x goto 110,120,130,140,150,160,170,180:goto2
0
100 :

```

In Zeile 20 ist eine bedingte Verzweigung mit dem If..Then-Befehl gestaltet. In der For..Next-Schleife wird A\$ in IN\$ gesucht, wobei X bis zu einer Größe von der Stringlänge von IN\$ steigt. Wird A\$ in IN\$ gefunden, wird das Programm an Zeile 40 abgezweigt. Wenn es nicht in IN\$ vorhanden

ist, wird es die Schleife verlassen und mit Goto-Befehl an den Anfang der Zeile 20 geleitet. Der If..Then-Verzweigungsbefehl erzeugt eine Wand, die nur dann durchlässig wird, wenn die gestellte Bedingung erfüllt ist. In unserem Falle ist es, daß A\$ in IN\$ vor-

handen ist. Einen anderen Befehl: If..Then..
Eise kann man hier gut gebrauchen. Leider gibt es diesen auch nicht im Commodore Befehlsatz. Wir gestalten ihn mit Boolescher Algebra. Danach brauchen wir die Zeile 30 nicht mehr.

```

20 geta$:forx=1tolen(in$):on-(a$=mid$(in$,x,1))got
o 40:next:goto 20

```

Die Boolesche Algebra stützt sich auf Boolescher Aussage. Sie ist sehr aufrichtig und gibt nach einer Anfrage, ob die Bedingung erfüllt ist oder nicht, eine eindeutige „ja“ oder „nein“ Antwort. Es gibt kein „jein“. Zahlenmäßig erfaßt heißt das, ja=1 und nein=0. Da aber die Antwort sehr aufrichtig ist, werden alle 8 Bits der Antwortbyte mit „1“ oder „0“ besetzt. Für den Mikroprozessor bedeutet eine „1“ im 7. Bit (Bits werden 0 bis 7 genannt) eine negative Zahl. Also eine aufrichtige Antwort wird negativ bewertet. Die Boolesche Aussage kann deshalb entweder 0 oder -1 sein. Gestützt auf diese Form der Aussage (0 oder -1) kann man alle bedingten Verzweigungen umformen. Man formuliert den

gesamten Vorgang in Form einer Gleichung (bzw. Ungleichung) und bekommt eine eindeutige Boolesche Antwort für einen gegebenen Wert von Variablen in der Gleichung. Mit einer For..Next-Schleife kann man den Wert steigend (bzw. fallend) ändern. Bis zu einem bestimmten Wert, mit dem die Gleichung stimmt, bekommt man nur 0 als Antwort. Danach kann man mit On..Goto-Befehl sehr gut operieren. Dieser reagiert nicht auf 0 und bildet auch keine Wand, sondern springt einfach zum nächsten Befehl hinter dem Doppelpunkt. Daß die Boolesche Algebra nicht nur für die Verzweigung verwendbar ist, wird im Laufe dieses Artikels mit mehreren Bei-

spielen deutlich gemacht. Bildung eines absoluten Wertes oder Sonstiges wird an Hand der Beispielprogramme so deutlich, daß weitere Aufklärung dafür unnötig ist. Viele Wiederholungen in den Zeilen von 110 bis 180 kann man folgendermaßen vermeiden und damit das Programm sichtbar straffen. Es besteht zwischen dem Wert von X und dem von den Funktionstasten ein Zusammenhang:

X	1	2	3	4
	5	6	7	8
Fn	1	3	5	7
	2	4	6	8

Die ersten vier haben die Beziehung:
 $F_n = 1 + 2x(x-1)$

und die letzten vier die Beziehung:
 $F_n = 2x(x-4)$.

Wir gestalten das Programm folgendermaßen um:

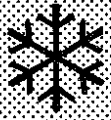
```

10 in$=" "
20 geta$:forx=1tolen(in$):on-(a$=mid$(in$,x,1))got
o 40:next:goto 20
30 :
40 print"funktionstaste":on-(x<5)goto 50:print2*(
x-4):goto 20
50 print1+2*(x-1):goto 20

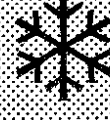
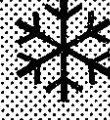
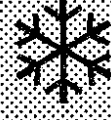
```

Damit haben wir ein Programm entwickelt, das genau soviel leistet wie vorher.

Die Fortsetzung dieser Serie finden Sie in der nächsten Ausgabe von „Compute mit“.



STARSHIP



Ein Geschicklichkeitsspiel für den Commodore 64

++++ top secret ++++ top secret ++++ top secret ++++ top secret ++++

++++
top secret
++++

++++
top secret
++++

... Oberster Rat Terra: Sternzeit 2985 ...

Einer geheimen Information nach formieren sich jenseits unserer Galaxis, im fernen ANDROMEDA - Nebel, ausserirdische Verbaende zur Invasion gegen unser Sonnensystem!

Ein RAUMKREUZER der Gewichtsklasse C wird zur Zeit fuer einen SPEZIALEINSATZ ausgeruestet.

BESONDERHEITEN: Wirkungsvolle Bewaffnung
Rundum wirkender Schutzschirm
Weitreichendes Ortungssystem
Ueberlicht in 3 sec

START: SOFORT von ANTARES III ... Beauftragen Sie Ihren BESTEN PILOTEN ...

++++ top secret ++++ top secret ++++ top secret ++++ top secret ++++

... hier setzt nun der Spieler ein. Zunächst gilt es, das Schiff von ANTARES III zu starten.

ANTARES ist ein unwirklicher Planet ohne Atmosphäre, aber sehr hoher Dichte, was zur Folge hat, daß unzählige Meteore den Start eines Raumschiffes erschweren (deshalb auch kaum von den Außerirdischen beachtet!).

Um nun die nötige Energie aufzubringen, die gewaltige Anziehungskraft des Himmelskörpers zu überwinden, muß „STARSHIP“ 15 Meilen beschleunigen, ehe der AUTOPILOT einsetzen kann. Eine Kollision mit einem Meteor beschädigt den Kreuzer dank seines Schutzschirmes nicht, bremst die Fahrt aber erheblich ab. Nun müssen erneut die 15 Meilen „Anlauf“ genommen werden.

(Anzeige der zurückgelegten Strecke ist der Balken oben links am Bildschirm). Ein AUSWEICHEN erfolgt durch Steuerung mittels Joystick nach links oder rechts. Ist das Schiff endlich aufgestiegen, beginnt DAS EIGENTLICHE SPIEL, und die Szene

wechselt in den KOMMANDOSTAND von STARSHIP.

Trotz Vollautomatisierung müssen wichtige Funktionen vom Piloten eingeleitet werden. Da wäre zunächst einmal die Beobachtung der Anzeigen in der untersten Zeile des Bildschirms. Die erste (2-stellige) Zahl ist der Trefferzähler, der sich nach jedem erfolgreichen Abschub erhöht. Dann folgt die ORTUNGSANZEIGE. Ist hier ein Zeiger zu sehen, befindet sich ein gegnerisches Schiff im Umfeld und zwar in Richtung des Pfeiles. Vorsicht ist geboten! Kommt der Gegner in Sichtweite, wird er das Feuer eröffnen. Die empfindliche Zone am STARSHIP ist der BUG vorne. Durch geschicktes Ausweichen oder im Notfall durch Einschalten des Schutzschirmes mit -F7- verhindert man einen Treffer.

Die 5-stellige Anzeige rechts neben der Ortung ist der ENERGIEZÄHLER. Hier wird die noch zur Verfügung stehende Energie abgelesen. Diese dezimiert sich durch Bewegung des Schiffes, aber hauptsächlich

beim Aktivieren des Schutzschirmes. Zeigt die Ortung nur einen Punkt, ist kein Gegner im betreffenden Raumsektor anzutreffen und ein anderer Sektor muß angefliegen werden.

Drücken der (SPACE)-Taste beschleunigt das Schiff rasant und bringt es innerhalb von Sekunden in den nächsten Sektor.

In manchen Raumabschnitten befinden sich besonders kleine, wendige Abfangjäger, die sehr schwer zu treffen sind. Hier liegt es an der Routine des jeweiligen Piloten, ob er weiterfliegt oder das Wagnis eines Kampfes auf sich nimmt.

Das Spiel ist zu Ende mit dem eigenen Abschub, wobei natürlich keine Auswertung erfolgt – oder mit dem Verlust der Energie. Ist die Energie fast zu Ende, drückt man kurz die (SPACE) Taste, und die Automatik bringt STARSHIP zur Trefferauswertung an die Basis zurück.

Vor dem Laden oder Abtippen des Programmes den BASIC-START mit dem Befehl **POKE 44,35 : POKE 8960,0 : NEW** eingeben.

```

0 REM VOR DEM LADEN ODER ABTIPFEN DES PR
PROGRAMMES VON HAND EINGEBEN: <24>
1 : <59>
2 REM POKE44,35:POKE 8960,0: NEW <228>
3 : <61>
4 REM***** <197>
5 REM* S T A R S H I F * <102>
6 REM* (C) 1985 * <48>
7 REM* HANS BAER * <131>
8 REM* 7700 SINGEN SIEDLUNGSWEG 4 * <132>
9 REM***** <202>
10 REM BASIC-ENDE AUF 30700 <176>
11 POKE55,236:POKE56,119 <22>
12 POKESI,236:POKE52,119:CLR <219>
13 GOSUB 5000: REM DATENPRUEFUNG <166>
15 FORI=3000TO3085:READQ:POKEI,Q:NEXT <117>
16 FORI=3100TO3508:READQ:POKEI,Q:NEXT <121>
17 FORI=35464TO35667:READQ:POKEI,Q:NEXT <222>
20 IQ=56334 : V=53248 : SI=54272 : R=532
80 <44>
21 IS=8192 : POKESI+24,15 <183>
22 POKESI+20,25:POKESI+6,0 <81>
23 GOSUB1000 : REM ROM INS RAM <1>
25 FORI=39072TO39165:READQ:POKEI,Q:NEXT <173>
30 FORI=39173TO39267:READQ:POKEI,Q:NEXT <232>
40 FORI=39280TO39559:READQ:POKEI,Q:NEXT <26>
45 FORI=39568TO39690:READQ:POKEI,Q:NEXT <37>
46 FORI=39693TO39732:READQ:POKEI,Q:NEXT <243>
60 FORI=0TO143:READQ:POKEZS+(65*8)+I,Q:N
EXT :REM ZEICHEN <0>
100 PRINT"GEDULD":SYS39693 <180>
112 FORI=2048 TO 2597:POKEI,0:NEXT <49>
115 FORI=2137 TO 2161:READQ:POKEI,Q:NEXT <216>
116 FORI= 2067TO 2082:READQ:POKEI,Q:NEXT <254>
122 FORI=2208 TO 2238:READQ:POKEI,Q:NEXT <22>
124 FORI=2270 TO 2302:READQ:POKEI,Q:NEXT <171>
126 FORI=2304 TO 2366:READQ:POKEI,Q:NEXT <22>
128 FORI=2368 TO 2430:READQ:POKEI,Q:NEXT <249>
129 FORI=2500TO2548:READQ:POKEI,Q:NEXT <29>
130 POKE 2460,56:POKE 2463,56:POKE 2466,
56 <80>
132 FORI=2597TO2622:READQ:POKEI,Q:NEXT <51>
140 FORI=35000TO35221:READQ:POKEI,Q:NEXT <63>
143 POKE30515,2:REM TEMPO <73>
144 FORI=30720TO30847:READQ:POKEI,Q:NEXT <225>
145 FORI=30864TO30991:READQ:POKEI,Q:NEXT <29>
147 FORI=31008TO31316:READQ:POKEI,Q:NEXT <139>
150 FORI=32998TO33169:READQ:POKEI,Q:NEXT <113>
155 FORI=33296TO33578:READQ:POKEI,Q:NEXT <112>
160 FORI=33788TO34312:READQ:POKEI,Q:NEXT <237>
170 FORI=704TO766:POKEI,0:NEXT <206>
171 POKE760,192:POKE763,255:POKE766,3 <45>
175 GOSUB600:POKE39747,0:GOSUB4000 <116>
180 POKE2040,38 :POKE2041,38 :POKE2042,1
1 <74>
182 POKE2044,34 :POKE2045,35 :POKE2046,3
6 :POKE2047,37 : REM ZEIGER <245>
184 POKEV+29,48 :POKEV+28,240:POKEV+23,0
:POKEV+37,52 <233>
186 POKEV+43,13:POKEV+44,13 <222>
187 POKEV+39,1:POKEV+40,1:POKEV+41,1 <201>
188 POKEV+45,7 :POKEV+46,5 <217>
190 POKEV+37,5:POKEV+38,1 <165>
195 POKEV+9,197:POKEV+11,197 <65>
196 POKEV+8,138:POKEV+10,187 <253>
199 POKE39744,0 <59>
200 POKEV+30,0:POKE39745,0:REM TREFFERL <20>
210 POKE32998,0:POKE33298,255 <154>
250 POKESI+6,0 <32>
260 POKEV+21,240:POKE32999,0 <185>
270 : <73>
280 : <83>
290 REM HAUPTPROGRAMM <223>

```

```

299 : <102>
300 SYS33792 <239>
310 IFPEEK(32999)=1THEN2000: REM ENERGIE
= 0 :AUSWERTUNG <44>
320 IF (PEEK (V+21)AND48)=0THEN3000: REM S
CHIFF ZERSTOERT <33>
390 POKEV+21,PEEK (V+21)AND59 <137>
400 FORZ=1TO20:A=INT (RND (1)*14) <16>
410 POKESI+1,A:POKESI+4,0:POKESI+5,10:PO
KESI+4,129 <58>
420 POKER,A:POKER+1,A+1:POKER+2,A:POKER+
3,A+1 <27>
430 NEXT:POKER+2,11:POKER+3,12 <160>
440 POKE R,0:POKE R+1,0:SYS33075 <217>
442 FORI=1TO1000:NEXT <92>
443 GOSUB600 <5>
445 POKE39745,0 <56>
450 SYS33412 <81>
500 GOTO300 <52>
510 : <57>
520 : <68>
600 POKEIQ,PEEK (IQ)AND254 <244>
610 POKEV+13,30:POKEV+15,30 <70>
620 POKEV+12,180:POKEV+14,204 <175>
625 POKEV+16,0 <222>
630 POKEIQ,PEEK (IQ)OR1 <241>
650 RETURN <26>
700 : <248>
800 : <93>
900 REM INTERRUPT VERBIEGEN U. ZEICHENSA
TZ INS RAM <144>
950 : <243>
1000 POKEIQ,PEEK (IQ)AND254 <133>
1010 POKE788,136:POKE789,138:REM IRQ-VEK
TOR <239>
1020 POKE1,PEEK (1)AND251 <248>
1030 IFPEEK (2)=1THEN1050:REM UEBERSPRUNG <92>
1040 FORI=0TO528 :POKEZS+I,PEEK (V+I):NEX
T:POKE2,1 <227>
1050 POKE1,PEEK (1)OR4 <4>
1060 POKE53272,(PEEK (53272)AND241)OR8 <172>
1070 POKEIQ,PEEK (IQ)OR1 <171>
1080 RETURN <202>
1090 : <128>
2000 REM AUSWERTUNG <120>
2002 E=PEEK (2000)-48:Z=PEEK (1999)-48:TR=
Z*10+E:S=0 <226>
2003 IF (PEEK (V+21)AND48)<>48THENS=1 <88>
2005 POKEV+21,0:PRINT "CLEAR"; <159>
2010 POKER,10:POKER+1,10:POKE646,1 <162>
2020 PRINT "{HOME} *****{SPACE}A{SPAC
E}U{SPACE}S{SPACE}W{SPACE}E{SPACE}R{SPAC
E}T{SPACE}U{SPACE}N{SPACE}G{SPACE}*****
" <152>
2150 PRINT "{BLUE}SIE{SPACE}HABEN"; "{WHIT
E}",TR; "{BLUE SPACE2}SCHIFFE{SPACE}ZERST
DERT":PRINT:PRINT <204>
2160 PRINT "{WHITE}KOMMENTAR{SPACE}HAUPTQ
UARTIER:":PRINT <143>
2170 IFTR<5THENT$="JAEMMERLICH,LACHHAFT"
:GOTO2200 <195>
2175 IFTR<10THENT$="FAST{SPACE}BESCHAEHM
END":GOTO2200 <157>
2180 IFTR<15THENT$="DURCHSCHNITTLICH":GO
TO2200 <68>
2185 IFTR<20THENT$="HERVORRAGEND":GOTO22
00 <118>
2190 T$="SUPER,SPITZE,NICHT{SPACE}ZU{SPA
CE2}UEBERBIETEN" <161>
2200 PRINT "IHRE{SPACE}LEISTUNG{SPACE}WAR
:":PRINT:PRINTT$ <167>
2210 IFS=1THENPRINT"!!!{SPACE}STARKER{SP
ACE}SCHADEN{SPACE}AM{SPACE}BUG{SPACE}!!!"

```

programme

" : GOTO 2250	<220>	4590 FOR I=1984 TO 1998: POKE I, 80: POKE I+26, 8	<190>
2220 PRINT "(DOWN2 BLUE) KEIN (SPACE) SCHADE		0: NEXT: POKE 1997, 78: POKE 2012, 78	<89>
N (SPACE) AM (SPACE) SCHIFF (SPACE) FESTGESTEL	<134>	4595 FOR I=1958 TO 1971: POKE I, 80: NEXT	
LT (SPACE) ! " : GOTO 2300		4600 POKE 1999, 48: POKE 2000, 48: FOR I=2006 TO	<86>
2250 PRINT "(DOWN) DAS (SPACE) SCHIFF (SPACE)		2010: POKE I, 57: NEXT	<252>
IST (SPACE) NICHT (SPACE) IHR (SPACE) EIGENTUM		4700 RETURN	<12>
!!! (SPACE) PASSEN (SPACE) SIE (SPACE) BESSER	<196>	4800 :	<113>
(SPACE) AUF "		4900 :	<5>
2300 PRINT "(DOWN2 SPACE) FEUER (SPACE) D	<204>	4950 REM DATENPRUEFUNG	<173>
RUECKEN " : WAIT 56320, 16, 16	<115>	4960 :	<240>
2310 GOTO 175	<7>	5000 PRINT "(CLEAR SPACES) DATENPRUEFUNG: "	
2500 :	<108>	5010 FOR B=1 TO 7: PRINT "*** (SPACE) BLOCK "; B;	<194>
2600 :	<220>	*** " : GOSUB 5100: NEXT	
2800 REM SCHIFF ZERSTOERT	<153>	5020 PRINT "(DOWN2 SPACE) DATEN (SPACE) RICHT	<192>
2900 :		TIG (SPACE) ... (SPACE) BITTE (SPACE) WARTEN "	<99>
3000 FOR Z=1 TO 15: A=INT (RND (1) * 15) : B=INT (R	<245>	5030 RESTORE: RETURN	<253>
ND (1) * 15)		5040 :	<7>
3010 POKE SI+1, A: POKE SI+4, 0: POKE SI+5, 15: P	<228>	5050 :	<89>
OKE SI+4, 129	<27>	5100 IF B=1 THEN Z= 495: S = 58508: GOTO 5200	<250>
3020 POKE R, A: POKE R+1, B	<50>	5120 IF B=2 THEN Z= 393: S = 45074: GOTO 5200	<77>
3050 POKE R, B: POKE R+1, A	<129>	5130 IF B=3 THEN Z= 443: S = 53198: GOTO 5200	<113>
3060 NEXT		5140 IF B=4 THEN Z= 450: S = 45929: GOTO 5200	<104>
3070 PRINT "(BLACK DOWN12 RIGHT13) SCHIFF (<23>	5150 IF B=5 THEN Z= 787: S = 90390: GOTO 5200	<96>
SPACE) ZERSTOERT (SPACE) ! ! "	<130>	5160 IF B=6 THEN Z= 455: S = 57633: GOTO 5200	<27>
3080 POKE R, 10: POKE R+1, 10 : SYS 39693	<126>	5170 Z=525 : S = 64773	<153>
3090 FOR I=1 TO 80: SYS 39487: NEXT	<201>	5200 A=0	<83>
3100 FOR I=1 TO 1000: NEXT	<152>	5210 FOR I = 1 TO Z: READ Q: A=A+Q: NEXT	
3105 POKE 198, 0: POKE 53280, 0: POKE 53281, 0	<12>	5220 IF A = S THEN PRINT "(SPACE) D. K (SPAC	<50>
3110 SYS 39576		E) " : RETURN	<71>
3120 PRINT "(HOME WHITE DOWN14 RIGHT10) G (<10>	5230 PRINT "FEHLER (SPACE) ! ! " : END	<198>
SPACE) A (SPACE) M (SPACE) E (SPACE) O (SPACE) V	<152>	5240 :	<208>
(SPACE) E (SPACE) R "	<190>	5250 :	<218>
3130 GET A\$: IF A\$="" THEN 3110	<243>	5260 :	<251>
3150 GOTO 175	<88>	5300 PRINT "ZAHL: "; Z; "SUMME: "; S: STOP	<190>
3500 :	<206>	5997 :	<208>
3600 :	<32>	5998 REM ***** BLOCK 1 *****	<192>
3700 REM STARTSZENE	<255>	5999 :	
3800 :	<126>	6000 DATA 238, 183, 11, 173, 183, 11, 201, 8, 240	<175>
4000 POKE R, 0: POKE R+1, 0: PRINT "(CLEAR) "	<111>	, 9, 201, 16, 240, 17, 201, 24, 240, 26, 96, 162	
4020 POKE 53270, PEEK (53270) OR 16	<124>	6001 DATA 1, 142, 34, 208, 142, 35, 208, 32, 247,	<229>
4030 POKE R+2, 1: POKE R+3, 2	<205>	11, 96, 162, 1, 142, 34, 208, 232, 142, 35, 208	
4040 FOR I=1824 TO 1863: POKE I, 69: NEXT	<238>	6002 DATA 32, 242, 11, 96, 162, 0, 142, 183, 11, 2	<143>
4042 FOR I=1864 TO 1903: POKE I, 70: NEXT	<70>	32, 142, 35, 208, 232, 142, 34, 208, 96, 169, 9	
4044 FOR I=1904 TO 1943: POKE I, 71: NEXT	<223>	6003 DATA 76, 249, 11, 169, 10, 162, 40, 157, 31,	<113>
4046 FOR I=1944 TO 1983: POKE I, 72: NEXT		219, 157, 71, 219, 157, 111, 219, 157, 151, 219	<24>
4048 FOR I=1984 TO 2023: POKE I, 73: NEXT	<81>	6004 DATA 157, 191, 219, 202, 208, 238, 96	
4050 SYS 39576: POKE V+21, 3: POKE V+29, 1: POKE	<188>	7000 DATA 169, 32, 141, 1, 212, 169, 0, 141, 4, 21	<123>
V+23, 0: POKE V+16, 0: POKE 2041, 39	<240>	2, 169, 170, 141, 5, 212, 169, 129, 141, 4, 212	
4055 SYS 39693	<246>	7001 DATA 96, 174, 27, 212, 224, 80, 176, 249, 96	<119>
4060 POKE V, 155: POKE V+1, 225: POKE 2040, 40	<32>	, 174, 27, 212, 224, 24, 144, 249, 96, 174, 27	
4070 POKE V+28, 1: POKE V+37, 0: POKE V+38, 1: PO	<78>	7002 DATA 212, 224, 85, 176, 6, 169, 1, 141, 16, 1	<125>
KE V+39, 0: POKE V+40, 6	<178>	2, 96, 224, 170, 176, 6, 169, 2, 141, 16, 12, 96	
4090 SYS 3219: SYS 3486	<104>	7003 DATA 169, 0, 141, 16, 12, 96, 173, 3, 208, 20	<90>
4100 :	<228>	1, 224, 240, 1, 96, 173, 2, 208, 56, 233, 12, 170	
4200 :	<195>	7004 DATA 176, 8, 173, 16, 208, 73, 2, 141, 16, 20	<83>
4500 : REM BLOCKGRAFIK STARSHIP	<234>	8, 142, 2, 208, 173, 29, 208, 9, 2, 141, 29, 208	
4505 :	<189>	7005 DATA 173, 23, 208, 9, 2, 141, 23, 208, 96, 3,	<223>
4510 PRINT "(CLEAR) " : SYS 39576: SYS 39693	<189>	208, 233, 169, 8, 141, 174, 12, 169, 1, 162, 13	
4520 FOR I=56096 TO 56295: POKE I, 15: NEXT	<199>	7006 DATA 142, 18, 12, 157, 0, 216, 202, 208, 250	<35>
4530 POKE 53282, 11: POKE 53283, 12: REM MUL	<59>	, 141, 19, 12, 169, 0, 141, 30, 208, 173, 30, 208	
TICOLOR	<84>	7007 DATA 208, 246, 162, 8, 160, 0, 136, 208, 253	<57>
4540 FOR I=56271 TO 56282: POKE I, 1: NEXT	<253>	, 202, 208, 248, 169, 75, 174, 18, 12, 157, 255	
4550 POKE 1828, 74: POKE 1839, 77: POKE 1849, 79	<118>	7008 DATA 3, 202, 208, 250, 173, 19, 12, 240, 55,	<18>
: POKE 1859, 76		206, 18, 12, 208, 1, 96, 206, 19, 12, 174, 27, 212	
4560 FOR I=1829 TO 1837: POKE I, 75: POKE I+21, 7	<89>	7009 DATA 224, 180, 176, 14, 173, 16, 208, 41, 25	<89>
5: NEXT: POKE 1838, 75		3, 141, 16, 208, 32, 57, 12, 76, 240, 12, 173, 16	
4565 FOR I=1880 TO 1888: POKE I, 75: NEXT: POKE 1	<123>	7010 DATA 208, 9, 2, 141, 16, 208, 32, 49, 12, 142	<123>
878, 80	<158>	, 2, 208, 169, 192, 141, 3, 208, 32, 28, 12, 32	
4570 POKE 1879, 81: POKE 1889, 82: FOR I=1868 TO	<165>	7011 DATA 126, 13, 173, 16, 12, 240, 39, 201, 1, 2	<158>
1877: POKE I, 80: POKE I+22, 80: NEXT		40, 19, 238, 2, 208, 208, 11, 173, 16, 208, 73	
4580 POKE 1904, 74: FOR I=1905 TO 1983: POKE I, 7	<126>	7012 DATA 2, 141, 16, 208, 238, 2, 208, 76, 42, 13	<165>
5: NEXT: POKE 1943, 76		, 206, 2, 208, 208, 11, 173, 16, 208, 73, 2, 141	
4585 POKE 1957, 77: POKE 1972, 79		7013 DATA 16, 208, 206, 2, 208, 238, 3, 208, 240,	<126>
		73, 32, 94, 12, 173, 0, 220, 106, 106, 106, 144	

7014 DATA21,106,176,34,173,0,208,201,254
,176,27,238,0,208,238,0,208,238,0,208 <150>

7015 DATA208,16,173,0,208,201,75,144,9,2
06,0,208,206,0,208,206,0,208,173,30,208 <162>

7016 DATA41,1,240,12,234,169,0,141,3,208
,32,64,122,76,142,12,32,184,11,76,173 <228>

7017 DATA12,238,19,12,76,173,12,173,29,2
08,41,253,141,29,208,173,23,208,41,253 <31>

7018 DATA141,23,208,169,32,162,16,157,25
5,3,202,208,250,234,234,234,76,65,12 <79>

7019 DATA169,6,141,39,208,32,64,122,32,1
13,153,32,184,11,32,184,11,206,1,208 <219>

7020 DATA208,239,96 <43>

8900 : <32>

8950 REM***** BLOCK 2 ***** <114>

8999 : <132>

9000 DATA174,27,212,224,1,240,11,224,2,2
40,18,224,3,240,25,76,181,138,173,67 <20>

9001 DATA155,73,1,141,67,155,76,181,138,
173,68,155,73,1,141,68,155,76,181,138 <135>

9002 DATA169,1,141,69,155,206,66,155,208
,82,169,2,141,66,155,169,0,205,67,155 <212>

9003 DATA240,35,238,14,208,208,11,238,14
,208,173,16,208,73,128,141,16,208,238 <177>

9004 DATA12,208,208,11,238,12,208,173,16
,208,73,64,141,16,208,76,15,139,206,12 <1>

9005 DATA208,208,11,206,12,208,173,16,20
8,73,64,141,16,208,206,14,208,208,11 <3>

9006 DATA206,14,208,173,16,208,73,128,14
1,16,208,76,15,139,76,59,139,169,0,205 <24>

9007 DATA68,155,240,16,169,1,141,68,155,
206,13,208,206,15,208,240,3,76,12,139 <152>

9008 DATA169,0,141,68,155,238,13,208,238
,15,208,169,181,205,15,208,176,212,76 <83>

9009 DATA22,139,206,63,155,208,17,169,9,
141,63,155,174,248,2,172,254,2,142,254 <85>

9010 DATA2,140,248,2,76,49,234 <29>

12000 DATA169,0,133,78,169,4,133,79,160,
39,177,78,160,0,145,78,169,40,24,101 <123>

12001 DATA78,176,5,133,78,76,168,152,133
,78,230,79,162,7,228,79,240,3,76,168 <232>

12002 DATA152,169,0,133,78,169,4,133,79,
160,38,177,78,200,145,78,136,169,40,24 <148>

12003 DATA101,78,176,5,133,78,76,211,152
,133,78,230,79,162,7,228,79,240,3,76 <213>

12004 DATA211,152,169,0,133,78,169,4,133
,79,136,16,214,96 <213>

13000 DATA169,0,133,78,169,4,133,79,96,3
2,5,153,160,0,177,78,160,39,145,78,165 <166>

13001 DATA78,24,105,40,176,5,133,78,76,1
7,153,133,78,230,79,162,7,228,79,240 <70>

13002 DATA3,76,17,153,32,5,153,160,1,177
,78,136,145,78,200,165,78,24,105,40,176 <162>

13003 DATA5,133,78,76,55,153,133,78,230,
79,162,7,228,79,240,3,76,55,153,32,5 <76>

13004 DATA153,200,204,98,153,240,4,76,55
,153,40,96 <134>

13995 : <27>

13996 REM***** BLOCK 3 ***** <227>

13999 : <31>

14000 DATA40,160,0,185,40,4,153,0,4,185,
208,6,153,248,6,200,204,112,153,208,238 <115>

14001 DATA160,0,185,80,4,153,40,4,185,16
8,6,153,208,6,200,204,112,153,208,238 <80>

14002 DATA160,0,185,120,4,153,80,4,185,1
28,6,153,168,6,200,204,112,153,208,238 <239>

14003 DATA160,0,185,160,4,153,120,4,185,
88,6,153,128,6,200,204,112,153,208,238 <183>

14004 DATA160,0,185,200,4,153,160,4,185,
48,6,153,88,6,200,204,112,153,208,238 <15>

14005 DATA160,0,185,240,4,153,200,4,185,
8,6,153,48,6,200,204,112,153,208,238 <191>

14006 DATA160,0,185,24,5,153,240,4,185,2
24,5,153,8,6,200,204,112,153,208,238 <96>

14007 DATA160,0,185,64,5,153,24,5,185,18
4,5,153,224,5,200,204,112,153,208,238 <225>

14008 DATA160,0,185,104,5,153,64,5,185,1
44,5,153,184,5,200,204,112,153,208,238 <128>

14009 DATA160,0,185,0,4,153,104,5,185,24
8,6,153,144,5,200,204,112,153,208,238 <175>

15000 DATA76,63,154,19,20,234,32,5,153,1
60,1,177,78,136,145,78,200,200,204,61 <139>

15001 DATA154,208,244,160,38,177,78,200,
145,78,136,136,204,60,154,208,244,160 <189>

15002 DATA39,177,78,160,20,145,78,160,0,
177,78,160,19,145,78,165,78,24,105,40 <172>

15003 DATA176,5,133,78,76,66,154,133,78,
230,79,162,7,228,79,240,3,76,66,154,96 <29>

16000 DATA174,27,212,224,80,176,249,96,1
69,255,141,15,212,169,129,141,18,212 <75>

16001 DATA162,0,169,32,157,0,4,157,0,5,1
57,0,6,202,208,244,162,31,157,0,7,202 <200>

16002 DATA208,250,76,249,154,234,32,144,
154,157,0,4,32,144,154,157,80,4,32,144 <64>

16003 DATA154,157,160,4,32,144,154,157,2
40,4,32,144,154,157,64,5,157,48,6,32 <120>

16004 DATA144,154,157,144,5,157,208,6,32
,144,154,157,224,5,157,128,6,136,208 <101>

16005 DATA202,96,169,66,160,1,32,190,154
,169,67,160,1,32,190,154,169,65,160,2 <66>

16006 DATA32,190,154,96 <222>

16500 DATA160,0,169,7,153,0,216,153,0,21
7,153,0,218,153,32,218,136,208,241,96 <101>

16501 DATA160,0,169,2,153,0,216,153,0,21
7,153,0,218,153,32,218,136,208,241,96 <42>

16995 : <223>

16998 REM***** BLOCK 4 ***** <183>

16999 : <227>

17000 DATA0,0,0,24,0,0,0,0,0,0,0,0,0,0,8,28,8,
0,0,16,16,56,16,16,0,0,0 <103>

17001 DATA0,4,6,255,255,6,4,0 <20>

17100 DATA170,85,85,255,255,170,170,
170,170,85,85,85,85,255 <212>

17101 DATA255,255,255,255,255,255,170,17
0,170,170,170,170,170,85,85 <225>

17102 DATA85,85,85,85,85,85,255 <22>

17200 DATA0,3,3,15,15,63,63,255,255,255,
255,255,255,255,255,255 <224>

17201 DATA0,192,192,240,240,252,252,255,
255,253,253,245,245,213,213,85 <190>

17202 DATA85,85,85,85,85,85,85,85,255,12
7,127,95,95,87,87,85 <181>

17203 DATA170,170,170,170,170,170,170,17
0,85,83,83,95,95,127,127,255 <92>

17204 DATA85,213,213,245,245,253,253,255 <62>

17500 DATA60,0,0,166,0,2,221,128,11,119,
96,9,221,224,11 <187>

17501 DATA119,96,2,221,128,0,166,0,0,60 <30>

17502 DATA24,0,0,255,0,3,255,192,254,153
,127,7,255,224,0,36 <90>

18000 DATA15,0,0,55,192,0,215,128,3,95,1
28,13,123,128,53,235,128,215,167 <206>

18001 DATA131,94,151,141,122,87,181,233,
87,170,170,171 <70>

18501 DATA240,0,0,220,0,0,215,0,3,245,19
2,2,237,112,2,235,92,2,218,215,2 <242>

18502 DATA214,181,194,213,173,114,213,10
7,94,234,170,170 <75>

19000 DATA3,0,0,5,0,0,21,0,0,84,0,0,80,0
,11,64,0,165,128,2,85,128,10,170 <229>

19001 DATA128,10,170,174,234,170,187,188
,204,128,8,204,128,10,170,128,3,51 <125>

19002 DATA64,0,170,64,0,15,80,0,0,84,0,0
,21,0,0,5,0,0,3,0,0 <219>

19501 DATA0,0,192,0,0,80,0,0,84,0,0,21,2
24,0,5,90,0,1,85,128,2,170,160,2 <226>

19502 DATA170,160,2,170,171,186,51,62,23
8,51,32,2,170,160,2,204,192,2,170,0,1 <63>

programme

```

19503 DATA240,0,1,0,0,5,0,0,21,0,0,84,0, <62>
0,80,0,0,192
19600 DATA62,0,1,255,192,3,241,224,2,127 <53>
,176,2,189,240,15,247,248,31
19601 DATA246,60,31,254,60,25,191,48,27, <235>
179,240,31,191,252,15,191,248,15,243
19602 DATA240,7,243,176,7,191,224,1,255, <1>
192,0,126
19700 DATA21,0,0,4,0,0,4,0,0,170,128,2,8 <133>
5,96,57,217,219
19701 DATA32,85,66,0,21,0,0,8,0 <224>
19995 : <163>
19997 REM***** BLOCK 5 ***** <136>
19999 : <167>
20000 DATA169,1,205,64,155,240,75,169,2, <114>
205,64,155,240,1,96,169,56,141,156,9
20001 DATA141,162,9,234,234,206,64,155,1 <5>
69,5,141,15,212,169,0,141,18,212,169
20002 DATA119,141,19,212,169,33,141,18,2 <120>
12,173,21,208,9,3,141,21,208,169,190
20003 DATA141,1,208,141,3,208,169,8,141, <30>
0,208,173,16,208,9,2,141,16,208,169,81
20004 DATA141,2,208,160,5,238,0,208,136, <24>
208,250,206,1,208,160,5,206,2,208,208
20005 DATAB,173,16,208,41,253,141,16,208 <101>
,136,208,240,206,3,208,169,180,205,1
20006 DATA208,144,18,169,0,141,156,9,169 <211>
,170,205,1,208,144,6,169,0,141,162,9
20007 DATA234,173,1,208,201,157,144,8,16 <223>
9,0,141,30,208,96,234,234,169,2,141,15
20008 DATA212,169,0,141,18,212,169,11,14 <166>
1,19,212,169,129,141,18,212,169,0,141
20009 DATA64,155,141,30,208,173,30,208,4 <169>
1,67,201,67,240,9,173,21,208,41,252,141
20010 DATA21,208,96,173,12,208,201,192,1 <148>
76,12,173,12,208,201,150,144,5,169,1
20011 DATA141,65,155,76,119,137 <122>
22000 DATA160,0,185,40,4,153,0,4,185,80, <64>
4,153,40,4,185,120,4,153,80,4,185,160
22001 DATA4,153,120,4,185,200,4,153,160, <33>
4,185,240,4,153,200,4,185,24,5,153,240
22002 DATA4,185,64,5,153,24,5,185,104,5, <8>
153,64,5,185,144,5,153,104,5,185,184
22003 DATA5,153,144,5,185,224,5,153,184, <106>
5,185,8,6,153,224,5,185,48,6,153,8,6
22004 DATA185,88,6,153,48,6,185,128,6,15 <37>
3,88,6,185,168,6,153,128,6,185,208,6
22005 DATA153,168,6,185,248,6,153,208,6, <55>
185,0,4,153,248,6,200,192,40,208,131
22006 DATA96 <226>
23000 DATA160,0,185,208,6,153,248,6,185, <71>
168,6,153,208,6,185,128,6,153,168,6,185
23001 DATABB,6,153,128,6,185,48,6,153,88 <170>
,6,185,8,6,153,48,6,185,224,5,153,8,6
23002 DATA185,184,5,153,224,5,185,144,5, <113>
153,184,5,185,104,5,153,144,5,185,64
23003 DATA5,153,104,5,185,24,5,153,64,5, <201>
185,240,4,153,24,5,185,200,4,153,240
23004 DATA4,185,160,4,153,200,4,185,120, <73>
4,153,160,4,185,80,4,153,120,4,185,40
23005 DATA4,153,80,4,185,0,4,153,40,4,18 <199>
5,248,6,153,0,4,200,192,40,208,131,96
24000 DATA169,1,205,31,121,240,103,173,1 <92>
6,208,41,64,201,64,208,1,96,173,16,208
24001 DATA41,251,141,16,208,173,12,208,1 <137>
72,13,208,141,4,208,140,5,208,173,21
24002 DATA208,9,4,141,21,208,169,32,141, <217>
8,212,169,0,141,11,212,169,202,141,12
24003 DATA212,169,129,141,11,212,169,1,1 <124>
41,31,121,141,30,121,169,31,141,28,121
24004 DATA172,27,212,192,85,144,12,192,1 <153>
70,144,16,169,3,141,29,121,76,142,121
24005 DATA169,1,141,29,121,76,142,121,16

```

```

9,2,141,29,121,169,3,205,29,121,240,47 <161>
24006 DATA169,2,205,29,121,240,55,172,30 <249>
,121,206,4,208,208,11,206,4,208,173,16
24007 DATA208,73,4,141,16,208,238,5,208, <159>
32,242,121,136,208,231,238,30,121,206
24008 DATA28,121,240,114,234,234,234,96, <170>
172,30,121,238,5,208,32,242,121,136,208
24009 DATA247,76,184,121,172,30,121,238, <220>
4,208,208,11,238,4,208,173,16,208,73
24010 DATA4,141,16,208,238,5,208,32,242, <187>
121,136,208,231,76,184,121,169,207,205
24011 DATA5,208,144,57,169,0,141,30,208, <12>
173,30,208,41,20,201,20,240,15,206,27
24012 DATA121,208,251,173,30,208,41,36,2 <192>
01,36,240,15,96,32,64,122,173,21,208
24013 DATA41,239,141,21,208,76,50,122,32 <141>
,64,122,173,21,208,41,223,141,21,208
24014 DATA76,50,122,173,21,208,41,251,14 <53>
1,21,208,169,0,141,31,121,96,169,5,141
24015 DATAB,212,169,0,141,11,212,169,12, <76>
141,12,212,169,129,141,11,212,96 <223>
24900 : <47>
24995 REM***** BLOCK 6 ***** <66>
24999 :
25000 DATA0,0,160,57,206,218,7,173,218,7 <116>
,201,47,240,1,96,140,218,7,206,217,7
25001 DATA173,217,7,201,47,240,1,96,140, <45>
217,7,206,216,7,173,216,7,201,47,240
25002 DATA1,96,140,216,7,206,215,7,173,2 <150>
15,7,201,47,240,1,96,140,215,7,206,214
25003 DATA7,173,214,7,201,47,240,1,96,16 <34>
9,1,141,231,128,96,238,208,7,173,208
25004 DATA7,201,58,240,1,96,169,48,141,2 <97>
08,7,238,207,7,173,207,7,201,50,240,1
25005 DATA96,169,1,141,230,128,96,173,21 <39>
,208,41,64,208,6,169,65,141,212,7,96
25006 DATA172,13,208,192,29,176,6,169,30 <248>
,141,212,7,96,173,16,208,41,64,208,19
25007 DATA173,12,208,201,151,176,6,169,3 <201>
1,141,212,7,96,169,68,141,212,7,96,173
25008 DATA12,208,201,166,176,237,76,130, <62>
129
26000 DATA60,245,255,0,169,255,141,19,13 <95>
0,234,234,234,234,234,169,1,141,1
26001 DATA212,169,0,141,4,212,169,225,14 <170>
1,6,212,169,129,141,4,212,169,0,141,31
26002 DATA121,141,64,155,173,21,208,41,2 <163>
48,141,21,208,174,19,130,160,5,136,208
26003 DATA253,202,208,248,206,19,130,240 <140>
,49,160,57,32,248,128,173,231,128,201
26004 DATA1,240,47,234,234,234,234,234,2 <198>
34,173,21,208,41,63,141,21,208,32,84
26005 DATA129,32,113,153,238,17,130,208, <94>
3,238,17,130,32,30,131,76,67,130,234
26006 DATA234,234,32,152,154,173,27,212, <5>
201,128,144,7,96,160,255,136,208,253
26007 DATA96,174,27,212,224,3,176,249,23 <255>
2,142,187,138,32,143,130,174,27,212,224
26008 DATA128,176,35,169,36,141,254,7,16 <161>
9,37,141,255,7,173,21,208,9,192,141,21
26009 DATA208,173,28,208,9,192,141,28,20 <202>
8,173,29,208,41,63,141,29,208,96,32,143
26010 DATA130,174,27,212,224,128,144,22, <174>
173,21,208,9,64,141,21,208,169,33,141
26011 DATA254,7,173,28,208,41,255,141,28 <197>
,208,96,32,143,130,169,32,141,254,7,174
26012 DATA27,212,224,128,144,17,173,29,2 <38>
08,41,191,141,29,208,173,21,208,9,64
26013 DATA141,21,208,96,173,29,208,9,64, <23>
141,29,208,173,21,208,9,64,141,21,208
26014 DATA96,173,16,130,141,0,212,173,17 <6>
,130,141,1,212,96
28000 REM***** BLOCK 7 *****

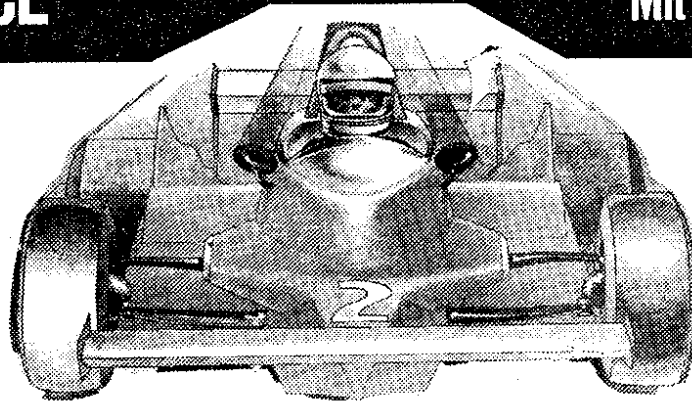
```

29000 :	<243>	41,2,240,28,238,13,208,238,15,208,32	<128>
31000 DATA4,10,6,0,160,5,136,208,253,165	<8>	31014 DATA227,133,206,254,131,208,8,32,1	<169>
,203,201,60,208,16,32,20,130,169,0,141		44,120,169,8,141,254,131,32,188,133,32	
31001 DATA6,212,173,231,128,201,1,208,1,	<18>	31015 DATA232,128,138,41,4,240,69,120,23	<134>
96,173,21,208,41,64,240,8,169,11,141		8,12,208,208,11,238,12,208,173,16,208	
31002 DATA250,7,76,47,132,169,39,141,250	<154>	31016 DATA73,64,141,16,208,238,14,208,20	<182>
,7,32,84,129,165,203,201,3,208,50,234		8,11,238,14,208,173,16,208,73,128,141	
31003 DATA234,234,160,57,32,6,129,32,33,	<57>	31017 DATA16,208,88,238,4,208,208,11,238	<232>
155,173,28,208,141,255,131,169,0,141		,4,208,173,16,208,73,4,141,16,208,206	
31004 DATA28,208,169,96,141,249,121,160,	<102>	31018 DATA254,131,208,8,32,160,152,169,8	<24>
32,140,1,212,160,0,140,4,212,160,9,140		,141,254,131,32,188,133,32,232,128,138	
31005 DATA5,212,160,17,140,4,212,76,123,	<38>	31019 DATA41,8,240,69,120,206,12,208,208	<23>
132,32,13,155,234,234,234,169,240,234		,11,206,12,208,173,16,208,73,64,141,16	
31006 DATA141,28,208,169,169,141,249,121	<61>	31020 DATA208,206,14,208,208,11,206,14,2	<41>
,173,69,155,201,1,208,3,32,32,121,169		08,173,16,208,73,128,141,16,208,88,206	
31007 DATA0,141,69,155,173,31,121,201,1,	<153>	31021 DATA4,208,208,11,206,4,208,173,16,	<127>
208,17,206,208,131,208,12,169,5,141,208		208,73,4,141,16,208,206,254,131,208,8	
31008 DATA131,32,32,121,234,234,234,234,	<4>	31022 DATA32,14,153,169,8,141,254,131,32	<60>
173,64,155,201,0,240,8,160,57,32,248		,188,133,32,232,128,76,244,133,169,9	
31009 DATA128,32,184,136,173,21,208,41,4	<23>	31023 DATA141,1,212,169,0,141,4,212,169,	<222>
8,208,3,76,210,133,173,65,155,201,1,208		105,234,141,5,212,169,129,141,4,212,96	
31010 DATA3,96,234,234,169,0,141,65,155,	<166>	31024 DATA169,1,141,33,208,141,32,208,32	<172>
173,0,220,73,255,41,31,170,41,16,240		,68,229,169,0,141,21,208,96,173,15,208	
31011 DATA13,173,64,155,201,1,240,6,234,	<253>	31025 DATA201,201,176,1,96,173,21,208,41	<24>
169,2,141,64,155,138,41,1,240,28,206		,63,141,21,208,96,206,251,131,208,13	
31012 DATA13,208,206,15,208,32,227,133,2	<47>	31026 DATA206,252,131,208,8,169,7,141,25	<26>
06,254,131,208,8,32,0,120,169,8,141,254		2,131,32,152,154,76,0,132	<253>
31013 DATA131,32,188,133,32,232,128,138,		32000 END	

MINI-CAR-RACE

Mit dem Commodore 64

Ein kurzes Spielprogramm, bei dem man schon mit wenigen Programmzeilen Spielvergnügen erreicht. Einer langen und ausführlichen Spielbeschreibung bedarf es an dieser Stelle nicht. Tippen Sie einfach die wenigen Programmzeilen ein und warten Sie, was passiert.



Wir sind der Meinung, daß auch diese Version, genau wie die bereits veröffentlichte für den Schneider CPC 464, eine gelungene Demonstration darstellt, daß man auch mit wenigen Programmzeilen bereits ein funktionsfähiges Programm erarbeiten kann.

Nehmen Sie Platz und starten Sie durch mit Ihrem „heißen“ Commodore!

```

5 REM**MINI-CAR-RACE (C= 64 VERSION)**
10 PRINT"J":B=15:A=20:T=100:P=0:C=A
20 FORI=1TOT:NEXTI:PRINTSPC(B);"■";SPC(10);" ":PRINT"TI":SPC(C);" "
30 PRINTSPC(A);"■#"
40 IFACB+10RA>B+10THENGOTO130
50 P=P+1:R=INT(RND(1)*2):IFR=0THENB=B-1
60 IFR=1THENB=B+1
70 IFB<0THENB=0
80 IFB>27THENB=27
90 C=A:T=T-1:IFT<2THENT=2
100 IFPEEK(56336)=123THENA=A-1
110 IFPEEK(56336)=119THENA=A+1
120 GOTO20
130 FORI=0TO16:POKE53280,I:NEXTI:PRINT"■PUNKTE:";P:IFPEEK(56336)<117THENRUN
140 GOTO130
    
```

Fehlen Ihnen noch Kassetten – oder Disketten von „Compute mit“ aus dem vorangegangenen Jahr? Möchten Sie noch Zeitschriften nachbestellen?

Im Moment sind wir noch in der Lage, alle veröffentlichten Programme und Zeitschriften des Jahrgangs '85 zu liefern. Unser Vorrat schwindet jedoch so schnell, daß Sie sich beeilen sollten, um die letzten Exemplare noch zu erreichen.

Greifen Sie also schnell zu der beiliegenden Bestell-Karte und füllen Sie diese aus. Die notwendigen Angaben für Ihre Bestellung entnehmen Sie bitte aus der Bestellliste. Die angegebenen Bestellnummern gelten jeweils für eine Heft-Ausgabe.

Heft 5/85		Heft 6/85		Heft 7/85		Heft 8/85	
Commodore		Commodore		Commodore		Commodore	
Bestell-Nr. COM C 1/1	16,- DM	Bestell-Nr. COM C 2/1	16,- DM	Bestell-Nr. COM CK 3/1	16,- DM	Bestell-Nr. COM CK 4/1	16,- DM
Bestell-Nr. COM D 1/1	20,- DM	Bestell-Nr. COM D 2/1	20,- DM	Bestell-Nr. COM CD 3/1	20,- DM	Bestell-Nr. COM CD 4/1	20,- DM
Schneider		Schneider		Schneider		Schneider	
Bestell-Nr. COM S 1/1	16,- DM	Bestell-Nr. COM S 2/1	16,- DM	Bestell-Nr. COM SK 3/1	16,- DM	Bestell-Nr. COM SK 4/1	16,- DM
				Bestell-Nr. COM SD 3/1	39,- DM	Bestell-Nr. COM SD 4/1	39,- DM
Heft 9/85		Heft 10/85		Heft 11/85		Heft 12/85	
Commodore		Commodore		Commodore		Commodore	
Bestell-Nr. COM CK 5/1	16,- DM	Bestell-Nr. COM CK 6/1	16,- DM	Bestell-Nr. COM CK 7/1	16,- DM	Bestell-Nr. COM CK 8/1	16,- DM
Bestell-Nr. COM CD 5/1	20,- DM	Bestell-Nr. COM CD 6/1	20,- DM	Bestell-Nr. COM CD 7/1	20,- DM	Bestell-Nr. COM CD 8/1	20,- DM
Schneider		Schneider		Schneider		Schneider	
Bestell-Nr. COM SK 5/1	16,- DM	Bestell-Nr. COM SK 6/1	16,- DM	Bestell-Nr. COM SK 7/1	16,- DM	Bestell-Nr. COM SK 8/1	16,- DM
Bestell-Nr. COM SD 5/1	39,- DM	Bestell-Nr. COM SD 6/1	39,- DM	Bestell-Nr. COM SD 7/1	39,- DM	Bestell-Nr. COM SD 8/1	39,- DM

Alle Bestellungen werden innerhalb

1 Woche

nach Eingang in unserer Software-Abteilung bearbeitet.

Alle Bestellungen werden sofort bearbeitet, so daß keine unnötigen Wartezeiten entstehen. Außerdem wird eine „Ladegarantie“ übernommen. Das bedeutet, Sie können fehlerhafte oder unladbare Programme zum Umtausch an uns zurücksenden.

In unserer Bestellliste finden Sie Software für alle Anwendungsgebiete. Schauen Sie einmal unverbindlich auf unser Angebot. Vielleicht können wir auch Sie von unseren hervorragenden Angeboten überzeugen, und Sie senden uns den ausgefüllten Bestellcoupon ein oder greifen einfach

zum Telefon und bestellen telefonisch unter den Nummern:

**0 56 51 / 4 06 43
oder 4 06 93**

Heft 1/86

Commodore	
Bestell-Nr. COM CK 1/2	16,- DM
Bestell-Nr. COM CD 1/2	20,- DM
Schneider	
Bestell-Nr. COM SK 1/2	16,- DM
Bestell-Nr. COM SD 1/2	39,- DM

Heft 2/86

Commodore	
Bestell-Nr. COM CK 2/2	16,- DM
Bestell-Nr. COM CD 2/2	20,- DM
Schneider	
Bestell-Nr. COM SK 2/2	16,- DM
Bestell-Nr. COM SD 2/2	39,- DM

Aus diesem Heft

Commodore	
Bestell-Nr. COM CK 3/2	16,- DM
Bestell-Nr. COM CD 3/2	20,- DM
Schneider	
Bestell-Nr. COM SK 3/2	16,- DM
Bestell-Nr. COM SD 3/2	39,- DM

Programm **Preis/** **Preis/** **Bestell-Nr.**
Kassette **Diskette**



Multi-Key/S-Tool 64/Interrupt-Programme	16,-- DM	21,-- DM	UC 2/85
Spritehilfe/Diskloader/Directory	21,-- DM	-	UC 3.1/85
Tape-Directory/Asmon/Data-Generator	15,-- DM	21,-- DM	UC 3.2/85
Fast Load	10,-- DM	-	UC 4/84
Diskmonitor/Zeichensatz	20,-- DM	-	UC 5/85
Reassembler/Maskengenerator	15,-- DM	20,-- DM	UC 6/85
Del 64/Treiberprogramm für 1526/MPS 802 Deceleator	14,-- DM	19,-- DM	UC 7/85
Supertapedirectory/Renew/Kontrollabfrage	15,-- DM	-	UC 8/85
Video-Utilities, Multi-Basic V3, Bildeditor	17,-- DM	23,-- DM	UC 9/85
Better Basic/Autnumber/Hardcopy/Terminalprogramm	17,-- DM	23,-- DM	UC 10/85
Speicher-Oszilloskop/Makro-Assembler/Change Type	49,-- DM	54,-- DM	UC 1/86
Reactor/Concentration/Datenbank	17,50 DM	23,50 DM	HC/C-1
Warlords/Caverns of Death	14,50 DM	19,50 DM	HC/C-2
Alien-Destroyer/Duell	24,50 DM	29,50 DM	HC/C-3
Ocean Game/Tennis	17,50 DM	23,50 DM	HC/C-4
The Caves	17,50 DM	23,50 DM	HC/C-5
Gardener	17,50 DM	23,50 DM	HC/C-6
Ufo/Skateboard Sam	24,50 DM	29,50 DM	HC/C-1-4
Mauern/Widerstand	8,-- DM	15,-- DM	C 41
Space-Comets/Erdsplatte/Sprite-Data	15,-- DM	23,50 DM	C 51
Autostart/Bestellschein/Roadpainter	16,50 DM	23,50 DM	C 61
Hardcopy/Space-Fighter/Data-Generator	15,50 DM	19,50 DM	C 71
Monster-Attack/Block-Painter/Epson-Drucker	16,50 DM	23,50 DM	C 81
Projekt/Datenbank	16,-- DM	23,50 DM	C 91
Spiders/The Basic	16,50 DM	23,50 DM	C 101
High Noon/Skeet/Grafik-Designer	17,50 DM	23,50 DM	C 121
Painter/Star-Baddle/Editor	17,50 DM	23,50 DM	C 22
Wüstenrally/Jet-Pac/Black Moore Castle	17,50 DM	23,50 DM	C 32
Brieftaube/Cadelon	19,50 DM	24,50 DM	C 42
Ritter Erik/Grand Prix Spritehilfe	19,50 DM	24,50 DM	C 52
Firebird/High Music	24,50 DM	29,50 DM	C 62
Moonsweeper/Scotti	24,50 DM	29,50 DM	C 72
Tron/Mercurious	24,50 DM	29,50 DM	C 13
Interceptor-Base/Schotter	24,50 DM	29,50 DM	C 23

Neu in unserer Software-Bestell-
liste der



Fantasy-Country			
Horrer-Castle	18,-- DM	23,-- DM	O 23
Zeichendesigner	14,-- DM	-	UO 8/85
OLD-Routine-Merge-Routine/Change Type	14,--	19,--	UO 1/86

Programm **Preis/** **Preis/** **Bestell-Nr.**
Kassette **Diskette**

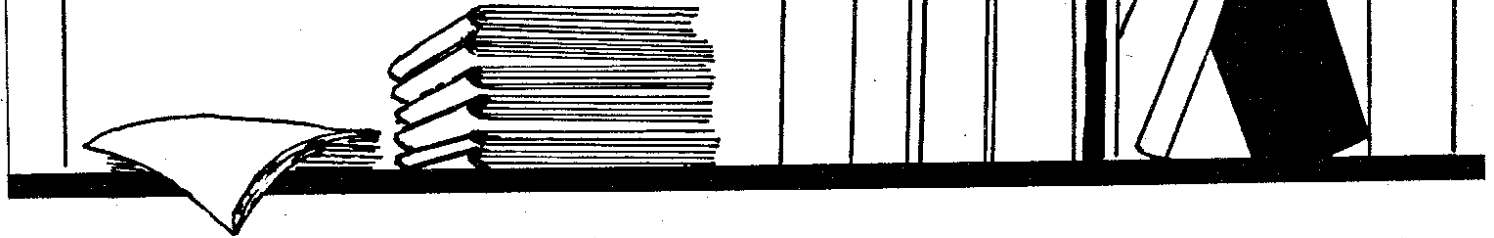


Bestellschein/Glücky	8,50 DM	15,-- DM	V 61
Multigraph/All Rammer	11,-- DM	15,50 DM	V 71
Zyklus/Meteorit	11,-- DM	19,50 DM	V 81
Garten/Schloß Gruselstein	14,-- DM	19,50 DM	V 91
Fressman/Outlaw	14,-- DM	19,50 DM	V 101
Prost/Bufalo Bill	14,-- DM	19,50 DM	V 121
Joy Man/Powerpack	14,-- DM	19,50 DM	V 22
Der rasende Malocher/Frankie goes to Pharao	14,-- DM	19,50 DM	V 32
Matron/Obst	14,-- DM	19,50 DM	V 42
Race On/Cagy	14,-- DM	19,50 DM	V 52
Nürburg 3D/Düsi	14,-- DM	19,50 DM	V 62
Breaker/Expulsion	14,-- DM	19,50 DM	V 72
Brückenbau/Jango	16,-- DM	21,50 DM	V 13
Inka-Schatz/Miner der Fassadenstreicher	16,-- DM	21,50 DM	V 23
Zyklus/Geldsammler	11,50 DM	-	HC/V-1
Star-Wars/Punktefresser	14,50 DM	-	HC/V-2
Catch the Fish/Mister Jump	14,50 DM	-	HC/V-3
Ghost's Eggs/Fox Hunt	14,50 DM	-	HC/V-4
Collector/Break Out	14,50 DM	-	HC/V-5
Q-Bert/Salvage Crew	16,50 DM	-	HC/V-6
Dog Fight/Schwitzbad	16,50 DM	-	HC/V-1-4
Sprites/Space-Battle	13,-- DM	19,-- DM	UV 2/85
Grafik-Painter	10,-- DM	-	UV 5/85
VIC-Clock	9,-- DM	-	UV 6/85
Decelerator	9,-- DM	-	UV 7/85
Joypaint	12,-- DM	-	UV 8/85
Disassembler, Old-Programme	10,-- DM	-	9/85
Directory/Farb-JRQ	10,-- DM	-	UV 10/85



Autorennen	11,-- DM	-	US 3/85
Universal-Datei V1	9,-- DM	-	US 4/85
Computerschrift/Symbol-Swap	10,-- DM	-	US 5/85
Keyboard Toolkit/Farbdecodierung	11,-- DM	-	US 6.1/85
Aladin	9,-- DM	-	US 6.2/85
Painter/Box-Befehl	14,-- DM	24,-- DM	US 7/85
Maschinensprache-Monitor/Disk-Hilfe	15,-- DM	25,-- DM	US 8/85
Basic, Maschinen-Kit	12,-- DM	22,-- DM	US 9/85
Backup/Directory/Ellipse/Disk	-	27,-- DM	US 10/85
Data-Generator/Grafik Toolkit/ASCII List (Turbo Pascal)	15,-- DM	25,-- DM	US 1/86
Fallschirmspringer	9,-- DM	-	HC/SR-3
Geisterschloß	9,-- DM	-	HC/SR-4
Zeichendesigner	12,50 DM	-	HC/SR-5
Mini Car Race/Interceptor 3 D	17,50 DM	26,50 DM	HC/SR-6
Secret Agent	12,50	24,50	HC/SR-1-4
Super-Miner	14,-- DM	24,-- DM	SR 42
CPC-Bert	14,-- DM	24,-- DM	SR 52
Conan's Castle	15,50 DM	25,50 DM	SR 62
Snider's Mace	16,50	26,50 DM	SR 72
Schneider Panik/ Killership	18,50 DM	28,50 DM	SR 13
Horrer-Caves/Midnight	18,50 DM	28,50 DM	SR 23

Was fehlt auf Ihrem Bücherbord?



Textverarbeitung auf dem Commodore 64

Leicht verständliche Einführung mit Programmbeispielen

Das lebendig geschriebene Buch beginnt mit einer kurzen Übersicht über die Programmiersprache BASIC, wobei das Schwergewicht auf die Befehle gelegt wird, die zur Textverarbeitung benötigt werden.

Danach werden die Anforderungen diskutiert, die an ein Textsystem gestellt werden. Hieraus werden der Aufbau und die Funktionsweise eines einfachen Textprogrammes entwickelt, das sich in übersichtliche Programmbausteine aufteilen läßt. Diese Bausteine werden anschließend programmiert und fügen sich nach und nach zu dem fertigen Textsystem zusammen. Der Leser nimmt auf diese Weise aktiv an der Entwicklung des Textverarbeitungsprogramms teil.

Es ist allerdings auch möglich das Programm nur einzugeben, ohne sich sehr intensiv mit der Materie zu beschäftigen. In

Autor: J. Kwiatkowski - Carl Hauser Verlag
München; ISBN 3-446-14154-5



einem weiteren Abschnitt, der für den fortgeschrittenen Leser gedacht ist, wird die Materie weiter vertieft und auf die speziellen Möglichkeiten des Commodore 64 eingegangen. Dadurch ist es möglich, daß der Leser seinen Commodore 64 optimal auszunutzen versteht. Der letzte Abschnitt stellt schließlich einige kommerziell erhältliche Textsysteme und deren Möglichkeiten vor.

Einige Kapitel aus dem Inhalt:

- Hardware
- Grundlagen
- Wir überarbeiten das Hauptmenue
- Die Untermenues werden programmiert
- Es geht weiter mit der Textverarbeitung
- Professionelle Textsysteme
- etc.

Denksport auf dem Commodore 64

43 Programme: Logik - Strategie - Mathematik

Autor: John W. Stephenson: Carl Hauser Verlag München; ISBN 3-446 14333-5

Eine Fülle von unterhaltsamen und spannenden Computernobelspielen, die geistig herausfordern. Diese fesselnden

Denksportaufgaben sind hauptsächlich zur Unterhaltung gedacht, sie haben jedoch einen weiteren erfreulichen Neben-

effekt:

Der Anwender lernt spielend eine Menge über Programmierung und Mathematik.

Konzentration und Vorstellungskraft werden verbessert, Kopfrechnen geübt, Logik trainiert.

Die auch für Anfänger besonders geeigneten Programme sind kurz gehalten, um die Eingabezeit zu verringern und Tippfehler möglichst auszuschalten.

Das Buch bietet so die Möglichkeit, sich im Denksportspiel mit der analytischen Geometrie, linearen Gleichungen, quadratischen Gleichungen, linearen und nichtlinearen Gleichungssystemen, Lagrange-Interpolation, Wahrscheinlichkeitsrechnung und Statistik, dem Pascalschen Dreieck, komplexen Zahlen, Zahlensystemen, Kartesischen Polarkoordinaten und



Euklidischem Abstand auseinanderzusetzen. Außerdem werden Logik- und Konzentrationsspiele geboten.

Der C-64 Anwender erhält mit diesem Buch die Möglichkeit, seinen Computer in allen Bereichen des Befehlsaufbaus und der Programmablauftechnik näher kennenzulernen.

Die Denksportaufgaben stellen eine sinnvolle und zugleich lehrreiche Abwechslung unter den Computerspielen dar.

Jeder der gern sein logisches Denken trainieren und ausbauen möchte findet hier die die richtigen Ansatzpunkte.

Datenübertragung mit Heim- und Personal-Computern

MODEM und Hacker Handbuch

von S. D. Roberts und E. Floegel

Dieses Buch wendet sich an Alle, die mit Datenübertragung näher in Kontakt kommen möchten. Es erläutert den heutigen Stand der Technik und zeigt anhand von Beispielprogrammen, wie Daten mit einem Computer übertragen werden. Gleichzeitig wird dabei erläutert, welche Dienste hierbei die Bundespost leistet und welche Netzwerke verwendet werden können. Im Einzelnen werden in den verschiedenen Kapiteln die Funktionsweise und der Aufbau der RS-232 Schnittstelle erläutert, zwei Bauanleitungen für den Bau einer RS-232 Schnittstelle veröffentlicht und die Schaltungen mit Hilfe eines Programmes erläutert. Außerdem wird das Modem erläutert und das Thema der Datenübertragung und Mailboxen näher durchleuchtet. Zum Schluß des Buches wird noch das Dutex-P Netz der Bundespost beschrieben und im Anhang des Buches werden Fachausdrücke erläutert



und Mailbox-Nummern bekanntgegeben. Alles in allem kann man sagen, daß dieses Buch einen Ein- und Überblick über die Datenübertragung mit Heim- und Personal-Computern gibt.

Immer gefragt...

ist Fachliteratur rund um den Computer. Zahlreiche Kontakte zu verschiedenen Verlagen im Inland, die uns freundlicherweise ihre neuesten Ausgaben zur Verfügung stellen, ermöglichen uns auch hier, immer am Ball zu bleiben. Erstes Fazit – ebenso wie auf dem Zeitschriftenmarkt quellen die Regale über von Computer-Fachliteratur. So undurchdringlich wie der Zeitschriftenschwungel ist mittlerweile auch der Markt der Computer-Taschenbücher bzw. der gesamten Fachliteratur.

Wir stellen Ihnen in jeder Ausgabe einige Neuerscheinungen vor. Zu beachten ist, daß durch die Vielzahl der angesprochenen Themen nur Buchempfehlungen weitergegeben werden können, die keine Wertung der einzelnen enthalten. Es ist durchaus möglich, daß Sie als Leser bei der Beurteilung eines Buches zu einem völlig anderen Ergebnis kommen können. Wir sind der Meinung, die Information ist für uns oberstes Gebot und werden daher auch weiterhin – dank der freundlichen Unterstützung der Buchverlage – die jeweils aktuellsten Bücher in kurzen Abhandlungen vorstellen.

Checksummer VC 1.0 für VC-20

Wie schon der C64-Checksummer besteht auch das VC-20-Programm aus einer kleinen Maschinenroutine. Aus Speicherplatzgründen konnte jedoch dieses Programm nicht „unter“ dem ROM abgelegt werden, sondern hier mußte der bekannte Kassettenpuffer erhalten. Dies hat den Vorteil, daß kein Basic-Speicher verlorengeht. Der Nachteil besteht darin,

daß nach der Aktivierung des Checksummers keine Kassettenoperationen durchführbar sind, da diese den Kassettenpuffer benötigen. Um die Kassettenoperationen wieder zuzulassen, muß der Checksummer durch den Befehl „SYS 58459“ sowie durch das Betätigen der Tasten STOP/RESTORE abgeschaltet werden. Aktiviert wird der Checksummer mit SYS 828. Die

Funktion und Arbeitsweise stimmt mit dem C64-Checksummer überein, so daß eine nochmalige Beschreibung an dieser Stelle entfallen kann.

Für alle, die nicht gerne abtippen, ist der Checksummer unter folgender Bestellnummer zu beziehen:

WV 10 K	Kassette	10 DM
WV 10 D	Diskette	15 DM

```

1 REM ***** <176>
2 REM CHECKSUMMER VC 1.0 <126>
3 REM VC 20 VERSION <8>
4 REM COPYRIGHT FRANK BRALL <126>
5 REM TRONIC-VERLAG <228>
6 REM ***** <181>
7 REM <150>
10 PRINT "CLEAR DOWN SPACE CHECKSUMMER(S
PAGE)VC(SPACE)1.0(SPACE2)" <216>
20 PRINT "DOWN COPYRIGTH(SPACE)FRANK(SP
ACE)BRALL" <221>
30 PRINT "DOWN-----[SPACE]03.01.86(SPA
CE)-----" <152>
35 PRINT "[DOWN SPACE4]TRONIC-VERLAG" <123>
60 DA=0:FOR I= 828 TO 1020 <158>
61 READ DA:SU=SU+DA <163>
62 POKE I,DA <64>
63 NEXT I <83>
65 IF SU<>22919 THEN PRINT "[DOWN]FEHLER(S
PACE)IN(SPACE)DATA-ZEILEN(SPACE)!" :END <14>
70 PRINT "[DOWN]" <222>
71 PRINT "AKTIVIEREN(SPACE)=SYS(SPACE)828
" <9>
72 PRINT "ABSCHALTEN(SPACE)=SYS(SPACE)584
59" <74>
73 PRINT "VOR(SPACE)BETRIEB(SPACE)DER(SPA
CE)DATA-" <40>
74 PRINT "SETTE(SPACE)DEN(SPACE)CHECKSUMM
ER" <38>
75 PRINT "ABSCHALTEN(SPACE)UND(SPACE)DIE" <80>
76 PRINT "TASTEN(SPACE)STOP/RESTORE" <198>
77 PRINT "BETAETIGEN(SPACE)!" <19>
80 POKE 816,116:POKE 818,116:POKE 817,19
6:POKE 819,196:REM SAVE/LOAD AUS <151>
90 SYS 828:PRINT "[DOWN]CHECK(SPACE)1.0(S
PACE)IST(SPACE)AKTIV(SPACE)!" <99>
100 DATA 169,71,141,2,3,169,3,141,3,3,96
,32,105,3,134,122,132 <85>
101 DATA 123,32,115,0,170,240,243,162,25
5,134,58,144,8,134,255,32,121 <100>
102 DATA 197,76,225,199,162,1,134,255,76
,156,196,166,255,224,1,240,3 <161>
103 DATA 76,96,197,169,0,141,248,3,141,2
49,3,141,250,3,160,2,24 <211>
104 DATA 177,95,200,113,95,141,250,3,160
,3,200,177,95,240,44,201,34 <143>
105 DATA 208,10,173,248,3,73,1,141,248,3
,177,95,174,248,3,208,4 <94>
106 DATA 201,32,240,228,238,249,3,174,24
9,3,24,177,95,109,250,3,141 <31>
107 DATA 250,3,202,208,244,76,138,3,56,3
2,240,255,142,246,3,140,247 <203>
108 DATA 3,162,0,160,0,24,32,240,255,169
,91,32,9,225,169,0,174 <232>
109 DATA 250,3,32,205,221,169,93,32,9,22
5,32,63,203,32,63,203,172 <21>
110 DATA 247,3,174,246,3,24,32,240,255,1
62,0,134,255,76,111,3,0 <121>
111 DATA 0,0,0,0,32,0 <118>

```

Ein Lauf- und Kletterspiel für den VC-20 mit mindestens 8 K-Ram

„Robin at Ravenwreck“

Das Spiel ist in drei Teile mit je fünf verschiedenen Runden untergliedert. Gleich zu Beginn begibt sich Robin auf die Diamantenjagd (blinkende Punkte). Seine Aufgabe besteht darin, die gesamten kostbaren und weit verteilten Edelsteine einzusammeln, bevor die Uhr abgelaufen ist. Um dies zu schaffen, bedarf es einiger

Übung und Geschicklichkeit. Im weiteren Verlauf des Spiels wird durch das Verändern des Screens die Diamantenjagd wesentlich erschwert: So findet ein ständiger Positionswechsel des zu durchquerenden Ausgangstores statt, erscheinen plötzlich neu einzusammelnde Diamanten auf dem Bildschirm, verän-

dern sich die Laufstege usw. Robin muß also seine „Brötchen“ (Diamanten) sauer verdienen.

Als Krönung des Ganzen verringert sich im höchsten Level auch noch die Spielzeit, sodaß Robin keine Ruhepause vergönnt ist und das Einsammeln nur noch in rasender Hast erfolgen kann.

Zum Abschluß noch einige Erläuterungen:

- Die Aufzüge transportieren Robin eine Etage höher.
- Auf den Fließbändern kann man sich nur in Richtung des Bandes fortbewegen.

- Im Gegensatz zu den grünen Laufstegen können die weißen nur einmal überquert werden. Nach ihrer Benutzung verschwinden diese und Robin fällt noch zu Boden, falls er sich im gefährdeten Bereich befindet.

Anzeigen:

Zeitleiste

6-stellige Score-Anzeige

Anzahl der noch einsammelnden Punkte

Anzahl der noch zu Verfügung stehenden Figuren

Hinweise zum Abtippen:

Vor dem Abtippen und vor jedem Einladen des Programms ist es wichtig, daß man diese Zeile eintippt:

POKE44,24:POKE6144,0:NEW

Erst danach das Programm einladen! Besondere Sorgfalt beim Eintippen verdient der Data-Block 4 im Vorprogramm und

die Zeilen 9500-9530 im Hauptprogramm, da hier die Assembler-Routinen liegen, ohne die das Programm nicht funktioniert.

Programmteil 1:

```

10 PRINT"PLEASE (SPACE)WAIT!":FORI=0TO264
:POKE 5120+I,PEEK(32768+I):NEXT <49>
40 X=1:FORI=828TO872:READA:POKEI,A:S=S+A
:NEXT:IFSK<>548260TO400 <253>
50 S=0:X=2:FORI=0TO336:READA:POKE5384+I,
A:S=S+A:NEXT:IFSK<>3949460TO400 <114>
60 S=0:X=3:FORI=0TO15:READA:POKE5360+I,A
:S=S+A:NEXT:IFSK<>271260TO400 <253>
70 S=0:X=4:FORI=0TO372:READA:POKE4640+I,
A:S=S+A:NEXT:IFSK<>4179960TO400 <56>
100 PRINTCHR$(147);:FORI=0TO22:PRINTCHR$(
(17);:NEXT:PRINTCHR$(5); <161>
105 POKE36865,42:POKE36879,8:CLR:C#=CHR$(
(145) <121>
110 FORI=1TO20:ONIGOSUB300,900,310,900,3
20,330,900,340 <103>
115 IFI<17THEN120 <1>
116 ONI-16GOSUB350,351,900,900 <73>
120 SYS828:NEXT:GOTO1000 <241>
300 PRINTC$"(SPACE)ROBIN(SPACE)AT(SPACE)
RAVENWRECK":RETURN <251>
310 PRINTC$"(C)MCLMXXV(SPACE)R.KOERBER"
:RETURN <224>
320 PRINTC$"(SPACES)HAZELNUT":RETURN <112>
330 PRINTC$"(SPACE)SOFTWARE":RETURN <17>
340 PRINTC$"(SPACE)CHR$(34)"A(SPACE)SIG
N(SPACE)OF(SPACE)QUALITY"CHR$(34):RETURN <139>
350 PRINTC$"DO(SPACE)YOU(SPACE)WISH(SPAC
E)THE(SPACE)IN-":RETURN <145>
351 PRINTC$"STRUCTIONS?(SPACE)Y/N":RETUR
N <58>
352 PRINTC$"THE(SPACE)GAME(SPACE)WILL(SP
ACE)LOAD.":RETURN <89>
353 PRINTC$"PLEASE(SPACE)WAIT.":RETURN <192>
360 REM ***** DATA-BLOCK 1 ***** <164>
**
361 DATA32,76,3,206,1,144,173,1,144,201,
38,208,243,76,97,3,169,1,133,2,162 <107>
362 DATA120,160,0,136,208,253,202,208,24
8,198,2,165,2,208,240,96,169,42,141 <237>
363 DATA1,144,76,117,233 <122>
400 PRINT"PRUEFSUMMENFEHLER(SPACE)IN(SPA
CE)BLOCK";X <206>
410 END <27>
600 PRINTC$"GUIDE(SPACE)ROBIN(SPACE)THRO
UGH":RETURN <117>
601 PRINTC$"THE(SPACE)CASTLES(SPACE)OF":
RETURN <105>
602 PRINTC$CHR$(28)"DURGHAN(SPACE)THE(SP
ACE)EVIL.":RETURN <90>
603 PRINTC$CHR$(5)"COLLECT(SPACE)THE(SPA
CE)TWINKLING":RETURN <40>
604 PRINTC$"DIAMONDS,(SPACE)BUT(SPACE)DO

```

```

N'T":RETURN <21>
605 PRINTC$"RUN(SPACE)OUT(SPACE)OF(SPACE
)TIME!":RETURN <214>
606 PRINTC$CHR$(28)"USE(SPACE)JOYSTICK(S
PACE)CONTROL":RETURN <45>
607 PRINTC$"TO(SPACE)MOVE.(SPACE)ELEVATO
RS":RETURN <142>
608 PRINTC$"RAISE(SPACE)YOU(SPACE)UP,(SP
ACE)TO(SPACE)GO":RETURN <25>
609 PRINTC$"DOWN,(SPACE)YOU(SPACE)HAVE(S
PACE)TO":RETURN <142>
610 PRINTC$"FALL(SPACE)UNTIL(SPACE)YOU'L
L":RETURN <148>
611 PRINTC$"HIT(SPACE)THE(SPACE)GROUND."
:RETURN <35>
612 PRINTC$CHR$(5)"YOU(SPACE)ONLY(SPACE)
HAVE(SPACE)THREE":RETURN <148>
613 PRINTC$"LIVES(SPACE)SO(SPACE)BE(SPAC
E)CAREFUL!":RETURN <140>
614 PRINTC$CHR$(28)"IF(SPACE)ALL(SPACE)D
IAMONDS(SPACE)ARE":RETURN <158>
615 PRINTC$"COLLECTED,RUN(SPACE)TO(SPACE
)THE":RETURN <196>
616 PRINTC$"EXIT.":RETURN <157>
617 PRINTC$"YOU'LL(SPACE)FIND(SPACE)MORE
(SPACE)DIA-":RETURN <164>
618 PRINTC$"MONDS(SPACE)IN(SPACE)THE(SPA
CE)NEXT":RETURN <195>
619 PRINTC$"LEVEL!":RETURN <118>
620 PRINTC$CHR$(5)"HINT:YOU(SPACE)MAY(SP
ACE)RUN":RETURN <215>
621 PRINTC$"(SPACES)FASTER(SPACE)ON(SPAC
E)THE":RETURN <2>
622 PRINTC$"(SPACES)ASSEMBLY-LINES":RETU
RN <177>
623 PRINTC$"(SPACES)BY(SPACE)PRESSING(SP
ACE)THE":RETURN <133>
624 PRINTC$"(SPACES)JOYSTICK(SPACE)INTO"
:RETURN <104>
625 PRINTC$"(SPACES)THE(SPACE)DIRECTION"
:RETURN <7>
626 PRINTC$"(SPACES)YOU(SPACE)WANT(SPACE
)TO(SPACE)MOVE":RETURN <77>
900 RETURN <21>
1000 POKE198,0:WAIT198,1:GETA$:IFA$="N"HT
HENS000 <206>
1010 IFA$<>"Y"THEN 1000 <131>
1015 PRINT:PRINT <210>
1020 FORI=1TO38:SYS 828 <46>
1030 ONIGOSUB600,601,602,603,604,605,606
,607,608,609,610,611,612,613,614,615,616
,617 <66>
1031 IFI<1860TO1040 * <222>
1032 ONI-18GOSUB618,619,620,621,622,623,
624,625,626 <222>
1040 SYS828:NEXT <198>

```

programme

```

1050 PRINT"REPEAT<SPACE>INSTRUCTIONS?";:
GOTO1000
5000 PRINT
5010 FORI=1TO8:ONIGOSUB900,900,900,352,9
00,353,900,900
5020 SYS828:NEXT
5030 POKE631,131:POKE198,1:NEW
5050 REM ***** DATA-BLOCK 2 *****
***
5051 DATA255,255,255,255,255,255,255,255,
,36,36,36,,,,,255,189,247,255
5052 DATA190,247,223,255,231,195,129,129
,129,129,129,230,230,230,230,230
5053 DATA230,230,230,40,40,255,235,235,2
55,40,40,255,255,255,255,187,239,127
5054 DATA247,255,255,24,255,187,239,127,
247,56,56,16,120,20,16,40,72,56,56
5055 DATA16,60,80,16,40,36,56,56,16,56,8
4,16,40,40,56,56,146,124,16,16,40
5056 DATA68,,,,,126,,,,,24,24,,,2,4,
8,16,32,64,,126,78
5057 DATA78,78,78,78,126,0,30,62,110,78,
14,14,14,0,126,78,14,126,64,78,126
5058 DATA0,126,78,14,62,14,78,126,0,78,7
8,78,126,14,14,14,0,126,78,64,126
5059 DATA14,78,126,0,126,78,64,126,78,78
,126,0,126,14,14,14,28,28,28,0,126
5060 DATA78,78,126,78,78,126,0,126,78,78
,126,14,78,126,0,251,251,251,255,255
5061 DATA255,255,255,191,191,191,255,255
,255,255,255,14,24,48,96,48,24,14
5062 DATA0,48,48,48,48,48,48,0,10,7,1
0,26,8,14,140,105,119,160,230,130
5063 DATA14,42,185,146,255,255,255,255,2
55,255,255,255,254,254,254,254,254
5064 DATA254,254,254,252,252,252,252,252
,252,252,248,248,248,248,248,248
5065 DATA248,248,240,240,240,240,240,240
,240,240,224,224,224,224,224,224,224
5066 DATA224,192,192,192,192,192,192,192
,192,128,128,128,128,128,128,128,128
5067 DATA,,,,,238,68,68,78,,,,,
,,0
5070 REM ***** DATA-BLOCK 3 *****
***
5071 DATA248,252,254,254,254,254,252,248
,31,63,127,127,127,127,63,31
5080 REM ***** DATA-BLOCK 4 *****
***
5081 DATA169,127,141,34,145,173,32,145,4
1,128,133,251,169,255,141,34,145,173
5082 DATA31,145,41,28,24,101,251,133,251
,173,31,145,41,32,133,252,96,162,0
5083 DATA189,128,18,201,58,240,6,232,224
,6,208,244,96,254,127,18,169,48,157
5084 DATA128,18,76,67,18,162,0,189,128,1
8,221,138,18,144,9,240,2,176,6,232
5085 DATA224,6,208,239,96,162,5,189,128,
18,157,138,18,202,16,247,96,0,0,0
5086 DATA48,48,48,48,48,48,0,0,0,48,48
,48,48,48,48,166,144,254,0,16,189
5087 DATA0,16,201,72,240,1,96,202,134,14
4,240,1,96,169,21,133,144,169,0,133
5088 DATA0,96,120,162,194,160,234,142,20
,3,140,21,3,88,96,162,0,169,48,157
5089 DATA128,18,232,224,6,208,248,96,162
,255,189,0,16,201,35,240,69,201,36
5090 DATA240,65,201,39,240,61,201,30,240
,65,201,31,240,61,201,58,240,57,201
5091 DATA59,240,53,201,38,240,54,234,234
,189,255,16,201,35,240,50,201,36,240
5092 DATA46,201,39,240,42,201,30,240,46,
201,31,240,42,201,58,240,38,201,59

```

```

5093 DATA240,34,201,38,240,35,202,208,18
1,96,169,5,157,0,148,76,237,18,169
5094 DATA4,76,22,19,169,12,76,22,19,169,
5,157,255,148,76,16,19,169,4,76,40
5095 DATA19,169,12,76,40,19,238,131,18,3
2,67,18,162,0,189,128,18,157,22,16
5096 DATA232,224,6,208,245,96,169,127,14
1,12,144,238,132,18,32,59,19,32,144
5097 DATA18,165,0,240,16,162,16,160,0,13
6,208,253,202,208,248,238,12,144,76
5098 DATA81,19,169,1,133,0,96,32,24,229,
169,8,141,15,144,32,210,255,169,1
5099 DATA141,134,2,169,31,141,14,144,120
,162,194,142,20,3,88,169,205,141,5
5100 DATA144,96

```

Programmteil 2:

```

20 SYS4979:C=RND(-TI):POKE144,21
30 FORI=0TO188:READA:NEXT:GOSUB9500:SYS8
33
100 CLR:GOTO3100
120 CLR:M=3:Z=255:POKE0,1:SYS4793
140 POKE144,21:L=L+1:IFL>STHENL=1:RZ=RZ+
1:PRINT"(CLEAR WHITE)":GOTO6100
141 IFRZ>2THENRZ=.
150 POKE36867,,:ONL:GOSUB1000,1020,1040,1
060,1080
155 SYS4806:POKE36867,46
190 POKEPP,37:POKEPP+33792,12:POKEPP-22,
37:POKEPP+33770,12
220 PRINTCHR$(19)CHR$(5)"",CHR$(28)"
";SYS4752
221 IFM=. THEN900
222 AP=4163:POKEAP,43:POKEAP+33792,1
223 FORI=0TOM-1:POKE4137+I,43:NEXT:SYS49
23
229 FORI=0TO50:NEXT
230 SYS4640:IFPEEK(0)<>1THEN600
232 PRINTCHR$(19)CHR$(158)CHR$(17),XZ-X"
(CLEAR SPACE)":SYS4752:A=PEEK(251)
235 IFRZ>1THENSYS4752
240 IFA=140GOTO400
250 IFA=28GOTO300
260 GOSUB5000:GOTO229
300 IFPEEK(AP+1)=35THEN230
310 IFPEEK(AP+1)=36THENPOKEAP,32:AP=AP+1
:GOTO4000
311 IFPEEK(AP+1)=38THENSYS4920:X=X+1:AP=
AP+1:GOSUB8000:AP=AP-1
320 POKEAP,32:AP=AP+1:GOSUB5000:POKEAP,4
2:POKEAP+33792,1:GOTO230
400 IFPEEK(AP-1)=35THEN230
410 IFPEEK(AP-1)=36THENPOKEAP,32:AP=AP-1
:GOTO4000
411 IFPEEK(AP-1)=38THENSYS4920:X=X+1:AP=
AP-1:GOSUB8000:AP=AP+1
420 POKEAP,32:AP=AP-1:GOSUB5000:POKEAP,4
1:POKEAP+33792,1:GOTO230
600 PRINTCHR$(19)CHR$(158)TAB(8)"SORRY":
POKE0,1
610 FORI=148TO220STEP.7:POKE36876,I:NEXT
:FORI=128TO200:POKE36876,I:NEXT
620 FORI=200TO128STEP-.5:POKE36876,I:NEX
T:POKE36876,0
630 M=M-1:POKEAP,32:POKE4137+M,32:GOTO22
0
700 IFX<>XZTHEN222
710 SYS(4940):POKE36876,.
730 TP=4118:POKE36875,,:X=.:JZ=INT(18*RN
D(1))+1:PRINT"(CLEAR DOWNIO)"CHR$(144)
740 ONJZGOSUB750,751,752,753,754,755,756

```



```

,757,758,759,760,761,762,763,764,765,766
,767 <179>
745 FORI=0T012:POKE38130+I,1:POKE38140+(
12-I),1:FORK=1T070:NEXTK,I <232>
746 FORI=1T01500:NEXT:GOTO140 <23>
750 PRINT"(SPACE6)FANTASTIC":RETURN <199>
751 PRINT"(SPACE)YOU(SPACE)ARE(SPACE)THE
(SPACE)CHAMPION":RETURN <197>
752 PRINT"(SPACE)OTHER(SPACE)WANNA(SPACE)
PLAY(SPACE)TOO":RETURN <85>
753 PRINT"(SPACE4)YOU(SPACE)ARE(SPACE)UN
IQUE":RETURN <185>
754 PRINT"(SPACE)WAKE(SPACE)ME(SPACE)UP(
SPACE)NEXT(SPACE)WEEK":RETURN <114>
755 PRINT"(SPACE4)I(SPACE)AM(SPACE)SPEAK
LESS":RETURN <8>
756 PRINT"(SPACE)YOUR(SPACE)GAME(SPACE)I
S(SPACE)ENDLESS":RETURN <0>
757 PRINT"(SPACE2)TELL(SPACE)ME(SPACE)YO
UR(SPACE)SCORE":RETURN <20>
758 PRINT"(SPACE)YOU(SPACE)MUST(SPACE)BE
(SPACE)A(SPACE)WIZARD":RETURN <69>
759 PRINT"(SPACE4)PREPARE(SPACE)TO(SPACE)
DIE":RETURN <98>
760 PRINT"(SPACE)02721(SPACE)10434(SPACE)
TELEPHONE":RETURN <226>
761 PRINT"(SPACE3)WHAT(SPACE)A(SPACE)SEN
TENCE":RETURN <166>
762 PRINT"(SPACE3)EINSTEIN(SPACE)PLAYING
":RETURN <125>
763 PRINT"(SPACE2)RUN(SPACE)FOR(SPACE)YO
UR(SPACE)LIFE":RETURN <110>
764 PRINT"HERE(SPACE)IS(SPACE)NOTHING(S
PACE)TO(SPACE)IT":RETURN <79>
765 PRINT"(SPACE3)HEAVEN(SPACE)AND(SPACE)
HELL":RETURN <212>
766 PRINT"(SPACE2)FOR(SPACE)AMUSEMENT(S
PACE)ONLY":RETURN <178>
767 PRINT"HAVE(SPACE)A(SPACE)SIGN(SPACE)
OF(SPACE)QUALITY":RETURN <69>
900 PRINTCHR$(19)CHR$(144)"(SPACE2)YOUR(
SPACE)GAME(SPACE)IS(SPACE)OVER(SPACE3)";
:SYS4701 <80>
910 FORI=0T09:POKE37888+I,3:POKE37898+(9
-I),3:FORK=1T070:NEXTK,I <199>
911 SYS4780:FORA=15T00STEP-1:POKE36878,A
:FORB=220-AT0160-ASTEP-4 <240>
920 POKE36876,B:POKE36876,0:NEXT:NEXT:GO
SUB9400 <65>
940 FORI=9T00STEP-1:POKE37888+I,0:POKE37
898+(9-I),0 <239>
950 FORK=1T070:NEXTK,I:POKE37154,255:CLR
960 PRINTCHR$(147):SYS833:POKE36878,31:G
OTO3000 <23>
1000 PRINT"(CLEAR DOWN3 WHITE)#(SPACE)&(
SPACE)&(SPACE)&"SPC(8)"&(SPACE)&(SPACE)&
(SPACE)#" <181>
1001 PRINT"#(((((#"; <7>
1002 PRINT"###(SPACES)###(SPACES)###(SPACE
4)#" <90>
1003 PRINT"#(SPACE7)#(SPACES)###(SPACE2)
&(SPACE)#" <226>
1004 PRINT"#(SPACES).....(SPACE3)#.....
" <229>
1006 PRINT"#(SPACES)#(SPACE8)#(SPACE4)##
" <37>
1007 PRINT"#(SPACE2)&(SPACE2)#(SPACE4)&(
SPACE3)#(SPACE4)##" <190>
1008 PRINT".....(SPACE3).....(SPACE2)#(S
PACE3)'" <69>
1009 PRINT"#(SPACE3)#(SPACE10)#(SPACE3)#
(SPACE)#" <31>
1010 PRINT"#(SPACE3)#(SPACE10)#(SPACE3)#
(SPACE)#" <61>

```

```

1011 PRINT".....(SPACE2).....(SPACE3)#(S
PACE3)'" <51>
1012 PRINT"###SPC(18)"###";:PRINT"#SPC(
18)"###"; <20>
1013 PRINT"#'(((('(((('(((('(((('(((('
#(SPACE2)
)"; <144>
1014 PRINT"(SPACE)#(SPACE4)#(SPACE5)#(SP
ACE4)#" <16>
1015 PRINT"(SPACE3)#(SPACE)&(SPACE2)#(SP
ACE2)&(SPACE2)#(SPACE)&(SPACE2)#" <251>
1016 PRINT"(SPACE3).....";PP
=4371:XZ=12:RETURN <218>
1020 PRINT"(CLEAR DOWN3 WHITE SPACE2)&(S
PACE)&(SPACE)&(SPACE)&(SPACE)&(SPACE)&(S
PACE)&(SPACE)&(SPACE)&" <121>
1021 PRINT"#'.....((.....#(SPACE2)
)#"SPC(16)"#" <120>
1022 PRINT"(SPACE2)#(SPACE)&"SPC(12)"&(S
PACE)#":PRINT"(SPACE2).....(SPACE2)'"
.....(SPACES)#"SPC(14)"#" <252>
1023 PRINT"(SPACE)&(SPACE)#"SPC(14)"#(SP
ACE)&":PRINT"(SPACE).....(SPACE4)'"
....." <96>
1024 PRINT"(SPACE)##(SPACE3)###(SPACE4)##
##(SPACE3)##" <198>
1025 PRINT"(SPACE)##(SPACE)&(SPACE)###(S
PACE4)###(SPACE)&(SPACE)##(SPACE2)##'.....
#.....#" <184>
1026 PRINT,"(LEFT)##":PRINT,"(LEFT)##",
#"(':::~^_::;^'('##"; <153>
1027 PRINT",,,,"(UP2)#(DOWN LEFT)#(DOWN L
EFT)##'.....";PF=4512:XZ=15 <152>
1039 RETURN <161>
1040 PRINT"(CLEAR DOWN3 WHITE)#(SPACE3)&
(SPACE2)&(SPACE10)&(SPACE2)##'(SPACE)'"
.....(SPACE7)'" <165>
1041 PRINT"#(SPACE2)#(SPACE6)#(SPACE8)##
(SPACE)#&(SPACE)#(SPACE)&(SPACE4)#(SPACE
3)&&(SPACE3)##" <127>
1042 PRINT"#'.....(SPACE).....(SPACE)'"
.....(SPACE8)###(SPACE7)##" <189>
1043 PRINT"(SPACE2)&(SPACE4)###(SPACE2)&
(SPACE4)##" <223>
1044 PRINT"(SPACE2).....#'(.....(SPAC
E6)##":PRINT"(SPACE2)##(SPACE3)&&(SPACE
3)" <179>
1050 PRINT"#(.....)";PRINTT
AB(10)"#(DOWN LEFT)##" <154>
1051 PRINT":::~^_::;^'('##"; <206>
1059 PP=4377:XZ=14:RETURN <53>
1060 PRINT"(CLEAR DOWN4 WHITE)....."SPC(
17)"#####(SPACE7)&(SPACE)&(SPACE7)#####(
SPACE4);;;(PURPLE)!{WHITE}:.....^" <183>
1061 PRINT"#####'(SPACE2);#(SPACE3)##":PR
INT"#(SPACE6);#(SPACE)&(SPACE)&##" <115>
1062 PRINT"#(SPACE)&(SPACE4)##'.....(
SPACE4)#(''(SPACE7)##" <140>
1063 PRINT"#(SPACE2)#(SPACE4)&&(SPACE)##
" <210>
1064 PRINT"#(SPACE2)#(SPACE2).....(SPA
CE).....;^'(SPACE3)'" <249>
1065 PRINT"(SPACE4)##":PRINT"(SPACE4)#(SP
ACE4)&&(SPACE5)";PRINT"(SPACE4)'" <29>
1066 PRINTTAB(11)"#(SPACE2)##";:PRINT"#S
PC(10)"#(SPACE)&(SPACE5)&" <0>
1067 PRINT"(LEFT)##'.....#" <240>
"; <214>
1070 PP=4502:XZ=11:RETURN
1080 PRINT"(CLEAR DOWN3 WHITE)##"SPC(20)"
##(SPACE)&(SPACE8)&(SPACE5)&(SPACE3)##"; <122>
1081 PRINT"#'(((('(((('(((('(((('(((('
#(SPACE2)'" <165>
";
1082 PRINT"(SPACES)#(SPACE2)##(SPACE9)##" <8>

```

programme

```

1083 PRINT" {SPACE2}&{SPACE2}*{SPACE2}*{SPACE6}&{SPACE2}*" <187>
1084 PRINT"#{SPACE}'*'*'_{;};;^''''" <80>
1085 PRINT"#{SPACE17}*{SPACE}*" <251>
1086 PRINT"#{SPACE17}*{SPACE}*" <17>
1087 PRINT"_{;};;^''''#{SPACE}'*'*'_{;};;^''''" <2>
1088 PRINT"#{SPACE18}###" <1>
1089 PRINT"#{SPACE}&{SPACE16}*" <54>
1090 PRINT"#{SPACE}'_{;};;^''''#{SPACE}'*'*'_{;};;^''''" <212>
1091 PRINT"###{SPACE16}###" <8>
1092 PRINT"#{SPACE5}&{SPACE}&{SPACE8}*" <166>
" <230>
1093 PRINT"#{SPACE}'*'*'_{;};;^''''#{SPACE}'*'*'_{;};;^''''" <162>
1094 X%=8:PP=4204:RETURN
3000 CLR:PRINT"(CLEAR DOWN2 BLACK SPACE2
}ROBIN{SPACE}AT{SPACE}RAVENWRECK{DOWN4}" <242>
3005 FORI=0T010:POKE37932+I,2:POKE37942+
(10-I),2:FORK=1T050:NEXTK,I <164>
3010 FORI=1T022STEP2:POKE37888+I,12:POKE
4096+I,38:POKE36876,210+2*I <149>
3011 FORK=1T040:NEXTK,I <27>
3020 FORI=21T00STEP-2:POKE37976+I,12:POK
E4184+I,38:POKE36876,210+2*I <221>
3021 FORK=1T040:NEXTK,I:POKE36876,0 <174>
3030 A*="{DOWN WHITE SPACES}HAZELNUT{DOW
N LEFT4}SOFTWARE{DOWN}":FORI=1T0LEN(A*):
PRINTMID$(A*,I,1) <245>
3035 FORK=1T050:NEXTK,I <57>
3040 PRINT:PRINT"(CYAN SPACE2)A{SPACE}SI
GN{SPACE}OF{SPACE}QUALITY" <135>
3050 PRINT"(DOWN3 RED SPACE2)COPYRIGHT{S
PACE}MCM LXXXV" <182>
3060 PRINT"(DOWN SPACE4)RONALD{SPACE}KOE
RBER{DOWN3 WHITE}" <249>
3080 GOTO9000 <79>
3100 A*="THE{SPACE}HAZELNUT{SPACE}SOFTWA
RE{SPACE}COMPANY" <164>
3120 A*="A*+{SPACE}PROUDLY{SPACE}PRESENT
S.....":GOSUB6000:GOTO3000 <44>
4000 AP=AP-66:POKEAP,43:POKEAP+33792,1 <194>
4010 FORK=200T0220:POKE36876,K:NEXTK:POK
E36876,0:GOTO230 <65>
5000 IFPEEK(AP+22)=32THENGOSUB7000:POKE3
6876,0:GOTO5000 <134>
5010 IFPEEK(AP+22)=40THENPOKEAP+22,32 <134>
5015 IFPEEK(AP+22)=58THENPOKEAP,32:AP=AP
+1:POKEAP,42:POKEAP+33792,1 <157>
5016 IFPEEK(AP+22)=59THENPOKEAP,32:AP=AP
-1:POKEAP,41:POKEAP+33792,1 <162>
5020 IFPEEK(AP)<>37THENRETURN <243>
5030 GOTO700 <0>
6000 A*="{SPACE22}"+A*+"{SPACE}" <139>
6010 FORI=1T0LEN(A*):PRINT"(HOME DOWN12)
" MID$(A*,I,22):POKE36876,190 <179>
6020 SYS4640:IFPEEK(252)GOTO6040 <200>
6030 POKE36876,0:RETURN <197>
6040 FORK=1T070:NEXTK:POKE36876,.:NEXTI:
RETURN <169>
6100 IFR%>1THEN6200 <198>
6105 A*="CONGRATULATIONS{SPACE}.....{SPA
CE}YOU{SPACE}HAVE{SPACE}COMPLETED{SPACE}
THE{SPACE}FIRST{SPACE}FIVE" <181>
6110 A*="A*+{SPACE}LEVELS{SPACE}AND{SPAC
E}YOU{SPACE}WILL{SPACE}NOW{SPACE}BEGIN{S
PACE}THE{SPACE}SECOND{SPACE}STAGE." <166>
6120 A*="A*+{SPACE}BUT{SPACE}BE{SPACE}W
ARNED{SPACE}.....{SPACE}EVERY{SPACE}LEVEL
{SPACE}HAS{SPACE}ITS{SPACE}PITS{SPACE}NO
W" <109>
6130 GOSUB6000:GOTO141 <163>
6200 A*="OKEY{SPACE}SUCKER.....YOU{SPACE
}WON{SPACE}THIS{SPACE}TIME{SPACE}TOO{SPA
CE}-{SPACE}BUT{SPACE}THE{SPACE}LAST{SPAC
E}" <95>
6210 A*="A*+{SPACE}WHD{SPACE}WINS{SPACE}WILL{SP
ACE}BE{SPACE}ME.....GO{SPACE}...{SPACE}T
RY{SPACE}YOURSELF{SPACE}AGAIN" <71>
6220 A*="A*+{SPACE}HA{SPACE}HA":GOSUB6000:GO
T0141 <2>
7000 POKEAP,32:AP=AP+22:POKEAP,44:POKEAP
+33792,1:Z=Z-5:IFZ<160THENZ=254 <180>
7010 POKE36876,Z:RETURN <237>
8000 IFR%=. THENRETURN <119>
8030 ONLGOTO9600,9700,9800,9900,9950 <114>
9000 RESTORE:LZ=0:V%=12:FORI=0T05:HI$=HI
$+CHR$(PEEK(4746+I)):NEXT <249>
9001 READT,D,F:LZ=LZ+1 <22>
9010 IFF=1THENPOKE36877,250:GOTO9050 <82>
9020 IFF=2THENPOKE36875,147:GOTO9050 <152>
9025 IFF=5THENPOKE36877,250:GOTO9050 <113>
9030 IFF=3THENPOKE36875,195:GOTO9050 <206>
9040 IFF=4THENPOKE36875,183:GOTO9050 <176>
9050 POKE36876,T:IFLZ=35THENV%=14 <156>
9060 FORI=V%T01STEP-0.013*V%D:NEXTI <89>
9061 IFLZ>62THENCLR:GOTO9000 <176>
9062 GOSUB9400 <212>
9070 A=PEEK(37151):IF(AAND32)=. THENGOSUB
9400:PRINT"(CLEAR)":GOTO120 <47>
9072 POKE646,(LZAND7):PRINT"{SPACES}HIGH
{SPACE}"HI$"{LEFT16}":GOTO9001 <2>
9080 DATA183,10,2,183,20,,183,20,,183,10
,1,183,20,,183,20,,183,10,2,147,10,,163,
10,1 <65>
9085 DATA183,10,0 <91>
9090 DATA175,10,3,175,20,0,175,20,0,175,
10,1,175,20,0,175,20,0,175,10,3,135,10,0 <244>
9095 DATA159,10,1,175,10,0 <40>
9100 DATA183,10,4,183,20,,183,20,,183,10
,1,183,20,,183,20,,183,10,2,191,10,0 <108>
9105 DATA195,10,1,201,10,0 <177>
9110 DATA195,10,4,183,10,0,175,10,5,159,
10,0,147,5,2,147,2.5,0 <195>
9120 DATA183,5,2,183,7.5,1,183,20,0,183,
10,2,147,10,0,163,10,1,183,10,0 <124>
9330 DATA175,5,3,175,7.5,1,175,20,0,175,
10,3,135,10,0,159,10,1,175,10,0 <126>
9340 DATA183,5,4,183,7.5,1,183,20,0,183,
10,2,191,10,0,195,10,1,201,10,0 <42>
9350 DATA195,10,4,183,10,0,175,10,5,159,
10,0,147,4.5,2,147,4,0 <226>
9400 FORI=0T03:POKE36874+I,0:NEXT:RETURN <107>
9470 REM <177>
9480 REM INTERRUPT <203>
9490 REM <198>
9500 DATA120,173,20,3,133,247,173,21,3,1
33,248,162,92,160,3,142,20,3,140,21 <206>
9510 DATA3,169,32,133,2,88,96,162,7,189,
208,21,24,106,144,2,9,128,157,208 <34>
9520 DATA21,189,216,21,24,42,144,2,9,1,1
57,216,21,202,16,229,198,2,208,12 <138>
9530 DATA169,32,133,2,173,14,144,73,83,1
41,14,144,108,247,0 <89>
9540 FORI=833T0907:READA:POKEI,A:NEXT:RE
TURN <91>
9600 POKEPP,32:POKEPP-22,32:JX=INT(5*RND
(1))+1 <0>
9605 ONJXGOTO9610,9620,9630,9640,9650 <78>
9610 PP=4499:GOTO9660 <147>
9620 PP=4182:GOTO9660 <87>
9630 PP=4510:GOTO9660 <61>
9640 PP=4371:GOTO9660 <104>
9650 PP=4505 <73>
9660 POKEPP,37:POKEPP+33792,12:POKEPP-22
,37:POKEPP+33770,12 <249>
9670 RETURN <121>
9700 IFAP=4364THENH=4495:GOSUB9750:RETUR
N <166>
9710 IFAP=4377THENH=4498:GOSUB9750:RETUR
N

```

```

N <252>
9720 IFAP=4232THENH=4501:GOSUB9750:RETUR
N <226>
9730 IFAP=4245THENH=4504:GOSUB9750:RETUR
N <57>
9740 RETURN <192>
9750 POKEH,38:POKEH+33792,12:XZ=XZ+1:RET
URN <112>
9800 IF(X/2)=INT(X/2)THENPOKE4436,36:H=5
:GOSUB9830:RETURN <133>
9820 POKE4436,35:H=1:GOSUB9830:RETURN <128>
  
```

```

9830 POKE38228,H:POKE38206,H:RETURN <242>
9900 IFJ9<>1THENPOKE38303,9:XZ=XZ-1:J9=1 <238>
9910 IFAP<>4511THENRETURN <166>
9920 POKE36879,25:POKE36877,170:FORI=0TO
2000:NEXT:POKE36879,8 <107>
9930 POKE36877,,:L=L-2:X=.:M=M-1:J9=.:GO
TO140 <252>
9950 IFAP<>4201THENS=SC-33:RETURN <220>
9960 POKE4256,38:POKE38048,12:POKE4278,4
0:POKE38070,1:XZ=XZ+1:RETURN <90>
  
```

TIPS & TRICKS für den Commodore VC-20

Kopieren des Zeichensatzes vom RAM ins ROM

```

10 for i=0 to 22 : read a : poke 850+i,a : next
20 data 169,255,189,0,128,157,0,28,202,208,247,169,255,189,0,129
30 data 157,0,29,202,208,247,96
  
```

Mit 'sys 850' wird diese Routine aufgerufen.

Bei fast allen Spielen wird zur Verbesserung der graphischen Darstellung ein neuer Zeichensatz definiert. Das vorherige Kopieren vom ROM ins RAM, das meistens gemacht wird, damit bestimmte Zeichen weiterhin normal geprintet werden können, wird erheblich schneller. Die Ausführungszeit benötigt den Bruchteil einer Sekunde. Sie kopiert den Zeichensatz von 32768-33024 in den Bereich von 7168-7679.

Lesekopfschoner für das Laufwerk VC 1541

Viele geschützte Programme verursachen ein Rattern des Lesekopfes; mit der Zeit kann das zu einem falsch eingestellten Lesekopf führen. Mit folgender Befehlsfolge, bevor man ein Programm vom Laufwerk lädt, kann man den Kopf schützen und schonen:

```

OPEN 15,8,15
PRINT?15;"M-W"; CHR$(106)CHR$(0)CHR$(1)
CHR$(133):CLOSE 15
  
```

TRACE ganz einfach simuliert

Mit dem Aufruf

SYS 56768

erhält man den Ausdruck von **IN** mit der gerade laufenden Zeilennummer. Benutzt man **TR** als Variable mit dem Wert **56768**, kann man mit **SYSTR** an den entsprechenden Stellen des Programms immer einen Hinweis für mögliche Fehler bekommen.

LIST innerhalb eines Programms - ohne Programmabbruch -

Folgende Zeile vor Abruf von LIST macht das möglich:

```

POKE768,174:POKE769,199:
LIST (nnnnn-nnnnn)
  
```

(nnnnn-nnnnn) ist der Bereich der gelistet werden soll, kann aber auch entfallen. Anschließend muß der Zeiger zurückgestellt werden mit:

```
POKE 768,139: POKE 768,196
```

Kombiniert man die Befehlsfolge auf einer Zeile des Programms, erfolgt nach dem LISTen kein Abbruch.

VTAB Ersatz

Basic V 2.0 kennt den Befehl VTAB nicht; mit etwas Stringmanipulation läßt sich VTAB leicht simulieren:

```
10 DIM VTAB$(23):A$ ="home":
VTAB$(0)=A$
```

OUT OF MEMORY vermeiden

Diese Meldung passiert oft, weil vorzeitig aus einer Warteschleife gesprungen wird: Eine Zeile wie z. B. **FOR T=1TO2000:GET A\$:IF S\$="" THEN NEXT** fordert diese Meldung geradezu heraus. Besser wäre die Zeile so zu gestalten: **- FOR T=1 TO 2000:T=T+(2000-T)*PEEK(198):NEXT:POKE198,0** damit die Schleife geschlossen wird. Um die Schleife nach einem bestimmten Tastendruck gefahrlos zu verlassen, setzt man folgendes ein: **T=T-(2000-T)*(PEEK(197)=X)**, wobei X dem Tastenwert aus der Tastaturmatrix entspricht.

Unterprogramme mit Namensaufuf

Durch die Art wie GOSUB nach dem Einlesen der Zeilennummer funktioniert - es wird als nächstes Zeichen immer nur ein Doppelpunkt gesucht - ist es ohne weiteres, und vor allem ohne REM, möglich die Bezeichnung des Unterprogrammes mit aufzunehmen, indem dieser mit vorangestelltem Komma nach der anzuspringenden Zeilennummer steht, z. B. **10 GOSUB 10000,DATENLESEN:** nächster Befehl

```
20 FOR R=1 TO 23:A$=A$+"crsr dn":
VTAB$(R)=A$: NEXT
```

Der Aufruf von **VTAB\$(R)** simuliert jetzt VTAB. Durch entsprechende Änderungen der beiden Programmzeilen läßt sich auch HTAB simulieren.

Autonumber – Routine für den Commodore VC-20 mit mindestens 3K-Erweiterung

Autonumber – ein nützliches Utility in Maschinensprache, das dem Basicprogrammierer das lästige Eingeben der Zeilennummern beim Programmieren erspart.

Sie brauchen dazu nur die gewünschte Schrittweite zu wählen und eine Zeile vorzugeben. Die nächste Zeilennummer wird dann automatisch vom Programm ausgegeben. Da das Programm vollständig in Maschinensprache geschrieben ist, merkt man von seinem Ablauf eigentlich gar nichts. Abgedruckt ist der Basic-Loader der Maschinenroutine. Geben Sie diesen ein und speichern ihn zur Sicherheit ab. Nach dem Start mit RUN dauert es ein wenig, bis die Datawerte eingelesen sind. Haben Sie beim Abtippen der Datawerte einen Fehler begangen, teilt das Programm Ihnen dieses mit. Sie müssen dann die entsprechenden Zeilen noch einmal überprüfen. Ansonsten meldet sich das Programm mit der Abfrage „Abspeichern (J/N) ?“. Kassettenanwender müssen hier leider nein (N) eingeben. Da die Routine im Kassettenpuffer liegt, ist es ohne weiteres leider nicht möglich, diese auf Kassette abzuspeichern. Kassettenanwender müssen also jedesmal, wenn sie mit dem Utility arbeiten wollen, den Basic-Loader einladen und starten. Auch jede Kassettenoperation zerstört das Programm; es muß

danach neu eingeladen werden. Zudem muß die Autonumber-Routine vor jeder Kassettenoperation des-aktiviert werden (wie wird noch beschrieben), da sich der Computer sonst aufhängen kann. Besser ergeht es da den Diskettenanwendern. Diskettenoperationen greifen nicht auf den Kassettenpuffer zu, folglich bleibt der Ablauf der Routine unbeeinflusst. Diskettenanwender können die Routine als reines Maschinenprogramm auf Diskette speichern; es belegt dann nur einen Block und das lästige Einlesen der Datas entfällt. Geben Sie zum Abspeichern also einfach ja (J) ein, nachdem Sie eine Diskette eingelegt haben. Es wird dann ein Programmfile mit dem Namen „AUTONUMBERCODE“ erzeugt. Eingeladen wird die Routine dann mit **LOAD"AUTONUMBER.CODE",8.1**. Danach muß **NEW** eingegeben werden, um alle Basicpointer zu normalisieren. Initialisiert wird die Routine mit **SYS828,x**. Dabei ist **x** ein Zahlenwert zwischen **0** und **65535** und gibt die gewünschte Schrittweite an. Geben Sie nun eine beliebige Basic-Programmzeile (Zeilennummer + Text) ein, gibt die Routine

automatisch die nächste Zeilennummer aus. Sie brauchen dann nur noch den gewünschten Programmtext anzuhängen und mit RETURN die Eingabe abzuschließen – es wird wiederum die nächste Zeilennummer ausgegeben. Um die Routine zu des-aktivieren, geben Sie nach Ausgabe einer Zeilennummer keinen Text ein, sondern schließen einfach mit RETURN ab. Die Routine schaltet sich dann selbst ab, kann jedoch mit **SYS828,x** wieder aktiviert werden, solange in der Zwischenzeit keine Kassettenoperation durchgeführt wurde, kein RESET erfolgt ist oder der Kassettenpuffer auf eine andere Weise gelöscht worden ist.

Für Interessierte ist in das Programm ein kommentiertes Assemblerlisting integriert. Der abgedruckte Basic-Loader entstand auf einem VC-20 mit 16kByte-Erweiterung. Es kann also sein, daß Sie mit gar keiner Ausbaustufe die REM-Anweisungen weglassen müssen. Sie sind jedoch für den Programmablauf völlig irrelevant und die Maschinenroutine selbst, die recht kurz ist, läuft auf jeder Ausbaustufe.

```
0 REM *****
1 REM *
2 REM *   A U T O N U M B E R - R O U T I N E   F U E R   D E N   C O M M O D O R E   V C 2 0
3 REM *
4 REM *   B E L E G T   W I R D   D E R   S P E I C H E R B E R E I C H   V O N   8 2 8   B I S   9 4 3
5 REM *
6 REM *   S Y N T A X :   S Y S 8 2 8 , S C H R I T T W E I T E
7 REM *
8 REM *   W R I T T E N   1 9 8 5   B Y   W A L D E M A R   R A A Z
9 REM *****
10 :
11 :
10000 DATA 20,FD,CE :REM JSR $CEFD ; --> CHKCOM
10002 DATA 20,BA,CD :REM JSR $CD8A ; --> FRMNUM
10004 DATA 20,F7,D7 :REM JSR $D7F7 ; --> GETADR
10006 DATA A5,14 :REM LDA $14 ; SCHRITTWEITE HOLEN
10008 DATA A4,15 :REM LDY $15 ;
10010 DATA B5,FB :REM STA $FB ; UND IN DER ZERO-PAGE ABSPEICHERN
```

```

10012 DATA B4,FC      :REM STY $FC      ;
10014 DATA A9,60     :REM LDA #$60      ; BASIC-WARMSTART-VEKTOR
10016 DATA A0,03     :REM LDY #$03      ; AUF EIGENE ROUTINE UMBIEGEN
10018 DATA BD,02,03  :REM STA $0302     ;
10020 DATA BC,03,03  :REM STY $0303     ;
10022 DATA 4C,74,C4  :REM JMP $C474     ; --> READY-MODUS
10024 DATA A9,B3     :REM LDA #$B3      ; EINSPRUNG ZUM DES-AKTIVIEREN
10026 DATA A0,C4     :REM LDY #$C4      ;
10028 DATA D0,F1     :REM BNE $0351     ;
10030 DATA 20,60,C5  :REM JSR $C560     ; BASICZEILE IN EINGABEPUFFER HOLEN
10032 DATA B6,7A     :REM STX $7A      ; CHRGET-POINTER AUF EINGABEPUFFER
10034 DATA B4,7B     :REM STY $7B      ;
10036 DATA 20,73,00  :REM JSR $0073     ; --> CHRGET NAECHSTES ZEICHEN
10038 DATA AA       :REM TAX          ;
10040 DATA F0,F3     :REM BEQ $0360     ; WEITER WARTEN, WENN PUFFER LEER
10042 DATA A2,FF     :REM LDX #$FF      ;
10044 DATA B6,3A     :REM STX $3A      ; FLAG FUER DIREKTMODUS SETZEN
10046 DATA 90,06     :REM BCC $0379     ; ZIFFER, DANN ZUR EIGENEN ROUTINE
10048 DATA 20,79,C5  :REM JSR $C579     ; BASICZEILE IN INTERPRETERCODE WANDELN
10050 DATA 4C,E1,C7  :REM JMP $C7E1     ; ZEILE AUSFUEHREN
10052 DATA 20,6B,C9  :REM JSR $C96B     ; ZEILENUMMER HOLEN
10054 DATA A6,7A     :REM LDX $7A      ; NAECHSTES ZEICHEN AUS PUFFER HOLEN
10056 DATA BD,00,02  :REM LDA $0200,X   ;
10058 DATA F0,D7     :REM BEQ $035A     ; PUFFER LEER, DANN DES-AKTIVIEREN
10060 DATA 18       :REM CLC          ; SONST SCHRITTWEITE ADDIEREN
10062 DATA A5,FB     :REM LDA $FB      ;
10064 DATA 65,14     :REM ADC $14      ;
10066 DATA AA       :REM TAX          ;
10068 DATA A5,FC     :REM LDA $FC      ;
10070 DATA 65,15     :REM ADC $15      ; = NAECHSTE ZEILENUMMER
10072 DATA B5,62     :REM STA $62      ;
10074 DATA B6,63     :REM STX $63      ;
10076 DATA A2,90     :REM LDX #$90      ;
10078 DATA 38       :REM SEC          ;
10080 DATA 20,49,DC  :REM JSR $DC49     ; --> INTEGER IN DEN FAC SCHREIBEN
10082 DATA 20,DF,DD  :REM JSR $DDDF     ; --> FAC NACH ASCII WANDELN
10084 DATA A2,FF     :REM LDX #$FF      ; NAECHSTE ZEILENUMMER IN DEN
10086 DATA EB       :REM INX          ;
10088 DATA BD,00,01  :REM LDA $0100,X   ; TASTATURPUFFER SCHREIBEN
10090 DATA 9D,77,02  :REM STA $0277,X   ;
10092 DATA D0,F7     :REM BNE $039C     ;
10094 DATA A9,20     :REM LDA #$20      ;
10096 DATA 9D,77,02  :REM STA $0277,X   ;
10098 DATA EB       :REM INX          ;
10100 DATA B6,C6     :REM STX $C6      ; POINTER SETZEN (GEDRUECKTE TASTEN)
10102 DATA 4C,9F,C4  :REM JMP $C49F     ; AKTUELLE ZEILE EINFUEGEN

```

```

10200 :
10210 :   RESTORE : P=0
10220 :   FOR I = 828 TO 943
10230 :     READ A$
10240 :     GOSUB 10500
10250 :     P = P + A
10260 :     POKE I , A
10270 :     NEXT I
10280 :
10290 :   IF P = 15173 THEN 10400
10299 :
10300 PRINT"LIHNEN IST BEIM ABTIP-";
10305 PRINT"PEN DER DATAZEILEN EIN";
10310 PRINT"FEHLER UNTERLAUFEN!"

```

```

10315 PRINT"UEBERPRUEFEN SIE BITTE";
10320 PRINT"NOCH EINMAL DIE ENT-";
10325 PRINT"SPRECHENDEN PROGRAMM-";
10330 PRINT"ZEILEN!":END
10399 :
10400 PRINT"LSIE HABEN ALLE DATA-";
10405 PRINT"WERTE RICHTIG EINGEGE-";
10410 PRINT"BEN. WENN SIE MIT DIS-";
10415 PRINT"KETTE ARBEITEN,KOENNEN";
10420 PRINT"SIE DAS PROGRAMM ALS";
10425 PRINT"REINE MASCHINENROUTINE";
10430 PRINT"ABSPEICHERN. ES BELEGT";
10435 PRINT"IN DER FORM NUR EINEN";

```



```

10440 PRINT"BLOCK. FUER KASSETTEN-";
10445 PRINT"ANWENDER BESTEHT DIESE";
10450 PRINT"MOEGELICHKEIT NICHT!"
10455 PRINT"@ABSPEICHERN (J/N) ?"
10460 GETG$: IFG$="" THEN10460
10465 IFG$<>"J" THEN10485
10470 SYS57809 "AUTONUMBER.CODE",8,1
10475 POKE251,60:POKE252,3:POKE781,182
10480 POKE782,3:POKE780,251:SYS65496
10485 PRINT"AKTIVIERT WIRD DURCH:"
    
```

```

10490 PRINT"@SYSB28,SCHRITTWEITE"
10495 END
10499 :
10500 A$=LEFT$(A$,2)
10510 A1=ASC(LEFT$(A$,1)):
A2=ASC(RIGHT$(A$,1))
10520 A1=A1-48:IFA1>9THENA1=A1-7
10530 A2=A2-48:IFA2>9THENA2=A2-7
10540 A=16*A1+A2
10560 RETURN
    
```

„Mini-Karate“

Sie, als VC-20-Fan sind zum Karateka avanciert und stehen auf der Matte einem computergesteuerten Gegner gegenüber. Ziel des Spieles ist es, den Kontrahenten möglichst oft zu Boden zu zwingen. Zur Realisation dieses Vorhabens verfügen Sie über ein Schlagreporoire, das aus drei Varianten besteht: Karatekick (F3), Magenhieb (F5) und Haken (F7). Zudem kann Ihr Kämpfer nach rechts (Semikolon) und links (Doppelpunkt) ausweichen. Jeder Treffer, den Sie oder Ihr Gegenüber einstecken müssen, kostet

Lebenspunkte: Karatekick = 3 Lebenspunkte
Magenhieb = 2 Lebenspunkte
Haken = 1 Lebenspunkt

Am unteren Bildschirmrand befindet sich eine Lebenspunktanzeige: Links ist der eigene Punktstand angegeben, rechts der

Auch der VC-20 (Grundversion) liefert seinen Beitrag für die Renaissance der asiatischen Kampfsportarten



„Nach dem Kampf mit dem Meister ist er vollkommen verdreht!“

des Gegners.

Wenn Sie den asiatischen Karatemeister besiegen wollen, müssen Sie Ihr ganzes

Können aufbieten. Denn selbst unsere erfahrenen Programmierer waren des öfteren „am Boden zerstört“.

```

0 CLR:PRINT" {CLEAR WHITE}";:POKE36879,8:
POKE36878,15
1 PRINT" {RVSON SPACE8}MINI {SPACE}- {SPACE
30}";
2 PRINT" {RVSON SPACE4} K {SPACE}A {SPACE}R
{SPACE}A {SPACE}T {SPACE}E' {SPACE27}";
3 PRINT" {RVSON SPACE9}BY {SPACE}: {SPACE31
}";
4 PRINT" {RVSON SPACE6} T.KLEINERT {SPACE28
}";
5 PRINT" {RVSOFF RIGHT2}STEUERUNG {SPACE}:
":PRINT" {DOWN SPACE9 RVSON}F3 {RVSOFF}-KA
RATEKICK"
6 PRINT" {RVSON}I {RVSOFF}- {SHIFTSPACE}->
{RVSON}J {RVSOFF SPACE}-UP LEFT/ {DOWN R
VSON}F5 {RVSOFF}-MAGENBOX"
7 PRINT" {SPACE8} \ {DOWN RVSON}F7 {RVSOFF}-
HAKEN"
8 PRINT" {DOWN3 SPACE}PRESS {SPACE}A {SPACE
}KEY {SPACE}TD {SPACE}BEGIN":POKE198,0:WAI
T198,1
9 FORT=7680+8*22T08185:POKET,32:NEXT
10 S$(1)=" {RIGHT}o {DOWN LEFT2} \ {DOWN L
    
```

```

EFT3 SPACE}I | " :S$(2)=" {RIGHT}/ {DOWN LEFT
2}o \ {DOWN LEFT3} ' |"
11 S$(3)=" {RIGHT}o {DOWN LEFT2} / \ {DOWN L
EFT2} | | " :S$(4)=" {RIGHT}o {DOWN LEFT2} / \ {
DOWN LEFT2} | \ "
12 S$(5)=" {DOWN LEFT}o \ {DOWN LEFT2} | \ "
:S$(6)=" {DOWN RIGHT} / \ {DOWN LEFT3}o \ "
13 C$(1)=" {RIGHT}o {DOWN LEFT2} / \ {DOWN L
EFT3} | | " :C$(2)=" {RIGHT} \ {DOWN LEFT2} \ o \
DOWN LEFT3} | | " :C$(3)=" {RIGHT}o {DOWN LEFT
2} \ \ {DOWN LEFT3} | | "
14 C$(4)=" {RIGHT}o {DOWN LEFT2} \ \ {DOWN L
EFT3} / | " :C$(5)=" {RIGHT2} / {DOWN LEFT3} \ o
{DOWN LEFT3} / | " :C$(6)=" {DOWN} \ \ {DOWN LEF
T2} \ o "
15 B$=" {RIGHT22} " :A$=" {DOWN21} "
16 FORT=8054T08185:POKET,160:NEXT:PRINT"
{HOME}"LEFT$(A$,19);" {RIGHT4 RVSON}LEBEN
SPUNKTE {SPACE}: {HOME RVSOFF}";
17 X=3:Y=15:SP=25:CF=50:GDSUB295
18 S$(7)=" {SPACE3 DOWN LEFT3 SPACE3 DOWN
LEFT3 SPACE3} " :C$(7)=" {SPACE3 DOWN LEFT
3 SPACE3 DOWN LEFT3 SPACE3} "
    
```

```

19 FORT=7680+8*22T07680+16*22STEP22:POKE
T,160:POKET+21,160:NEXT <79>
20 PRINT"HOME";A$(RVSON);SP"LEFT SP
ACE14";CP;"LEFT SPACE HOME RVSOFF";:F
ORZ=1T0100:NEXTZ <84>
24 IFRND(1)<.5THEN60 <73>
25 GETU$:IFU$=""THEN60 <164>
30 IFU$="":THENGOSUB95:X=X-1:GOSUB105 <11>
35 IFU$="":THENGOSUB95:X=X+1:GOSUB105 <6>
40 IFU$="{F3}"THENSH=1:SA=2 <239>
45 IFU$="{ES}"THENSH=1:SA=3 <11>
50 IFU$="{F7}"THENSH=1:SA=4 <39>
55 IFSH=1THEN150 <150>
60 IFER>0THEN145 <185>
65 IFY-1=XORY-2=XTHENCH=1:CA=INT(RND(1)*
3)+2 <185>
70 IFCH=1THEN165 <186>
75 GOSUB120:Q=SGN((X+2)-Y):Y=Y+Q:GOSUB13
0 <123>
80 IFRND(1)<.5THENIFY-1=XORY-2=XTHENCH=1
:CA=INT(RND(1)*3)+2 <242>
85 IFCH=1THEN165 <201>
90 FORZ=1T0200:NEXT:GOTO20 <203>
95 PRINT"HOME"LEFT$(A$,14)LEFT$(B$,X);
S$(7);"HOME"; <167>
100 PRINTLEFT$(A$,14)LEFT$(B$,Y);C$(1)"(
HOME)":RETURN <206>
105 IFX<2THENX=2 <70>
110 IFX>15THENX=15 <54>
115 PRINT"HOME"LEFT$(A$,14)LEFT$(B$,X)
;S$(1):RETURN <157>
120 PRINT"HOME"LEFT$(A$,14)LEFT$(B$,Y)
;C$(7) <22>
125 PRINT"HOME";LEFT$(A$,14)LEFT$(B$,X
);S$(1):RETURN <220>
130 IFY<2THENY=2 <103>
135 IFY>15THENY=15 <88>
140 PRINT"HOME"LEFT$(A$,14)LEFT$(B$,Y)
;C$(1):RETURN <120>
145 GOSUB120:Y=Y+1:GOSUB130:ER=ER-1:GOTO
20 <36>
150 GOSUB95:PRINT"HOME";LEFT$(A$,14)LE

```

```

FT$(B$,X);S$(SA) <58>
155 IFX+1=YORX+2=YTHENGOSUB170:GOTO180 <243>
160 FORT=1T0200:NEXT:PRINT"HOME"LEFT$(
A$,14)LEFT$(B$,X)S$(7):SH=0:GOSUB295:GOT
060 <2>
165 GOSUB120:PRINT"HOME"LEFT$(A$,14)LE
FT$(B$,Y);C$(CA):GOSUB170:GOTO200 <104>
170 POKE36877,INT(RND(1)*40)+200 <184>
175 FORT=15T00STEP-.8:POKE36878,T:NEXT:P
OKE36877,0:RETURN <20>
180 FORT=1T02:PRINT"HOME"LEFT$(A$,14)L
EFT$(B$,Y);C$(7) <134>
185 PRINT"HOME"LEFT$(A$,14)LEFT$(B$,Y)
;C$(4+T) <131>
187 FORZ=1T0200:NEXTZ,T <74>
190 PRINT"HOME"LEFT$(A$,14)LEFT$(B$,X)
;S$(1) <10>
195 CP=CP-(5-SA):ER=10:GOTO225 <132>
200 FORT=1T02:PRINT"HOME"LEFT$(A$,14)L
EFT$(B$,X)S$(7) <9>
205 PRINT"HOME"LEFT$(A$,14)LEFT$(B$,X)
;S$(4+T) <213>
207 FORZ=1T0200:NEXTZ,T <94>
210 PRINT"HOME";LEFT$(A$,14);LEFT$(B$,
Y)C$(1) <165>
220 SP=SP-(5-CA) <219>
225 IFSP<1ORCP<1THEN235 <111>
230 SH=0:CH=0:FORT=1T01000:NEXT:GOSUB95:
GOSUB120:GOSUB295:POKE198,0:GOTO20 <206>
235 PRINT"HOME DOWN? RIGHT?DER(SPACE)SI
EGER(SPACE)"; <89>
240 IFSP<1THENPRINT"IST(SPACE)VC-20" <69>
245 IFCP<1THENPRINT"SIND(SPACE)SIE" <50>
250 PRINT"DOWN?";SPC(4);"PRESS(SPACE)A
(SPACE)KEY" <236>
260 POKE198,0:WAIT198,1 <147>
270 RUN <153>
295 PRINT"HOME"LEFT$(A$,14)LEFT$(B$,X)
S$(1) <223>
300 PRINT"HOME"LEFT$(A$,14)LEFT$(B$,Y)
C$(1):RETURN <204>

```

6502-Assembler-Kurs

Teil 10

LSR Shift Right One Bit (Memory or Akkumulator) (Verschiebe eine Speicherstelle oder den Akkumulator um 1 Bit nach rechts)

Der LSR-Befehl gehört zu den „Bit-Manipulations“-Befehlen und wird deshalb oft in Verbindung mit relativen Sprungbefehlen verwendet. Eine weitere Verwendung findet dieser Befehl bei der Durchführung mathematischer Funktionen. Durch die Ausführung dieses Befehles wird der Inhalt des Akkumulators oder einer adressierten Speicherstelle um ein Bit nach rechts verschoben. Dies bedeutet, daß nach der Ausführung Bit 7 zu Bit 6, Bit 6 zu Bit 5 usw. verschoben wird. Das letzte Bit (Bit 0) wird in das Carry-Flag geschoben. War dieses Bit also auf 1 gesetzt, wird das Carry-Flag ebenfalls gesetzt, ansonsten wird es gelöscht. Durch die Rechtsverschiebung wird von links immer ein gelöscht Bit nachgeschoben.

HEX	Adressierung	Bytes	Syntax	
46	Null-Seite (direkt)	2	LSR addr	7 6 5 4 3 2 1 0
4E	Absolut (direkt)	3	LSR addr	
56	Null-Seite indiziert mit X	2	LSR addr, X	NV BDIZC
5E	Absolut indiziert mit X	3	LSR addr, X	0 x x

Beispiel:
 100 PRINT = \$BDCD ; Routine zur Zahlen-
 ausgabe
 105 BSOUT = \$F1CA
 110 ZAHL = 832 ; Variable
 120 ;
 130 START LDA 832 ; lade Akku mit Zahl
 140 JSR ZPRINT ; Zahl ausgeben
 150 LDA 832 ; lade Akku nochmal
 160 LSR ; verschiebe um 1 BIT
 170 JSR ZPRINT ; ZAHL ausgeben
 180 LDA 832 ; lade Akku nochmal
 190 LSR
 200 LSR
 210 JSR ZPRINT ; Zahl ausgeben
 220 RTS
 230 ;
 240 ZPRINT TAX ; Akku nach X übertragen
 250 LDA #0, ; High Byte löschen
 260 JSR PRINT
 270 LDA #13 ; Zeichen für Zeilenvor-
 schub
 280 JSR BSOUT
 290 RTS

Erklärung des Beispiels:

Zum erstenmal benutzen wir in unserem Beispiel zwei „ROM-Routinen“ zur Abkürzung unseres Programmes. Als ROM-Routine bezeichnet man ein Maschinen-Unterprogramm, welches eine bestimmte Aufgabe erfüllt und danach zum Hauptprogramm zurückspringt. Wie die Be-

zeichnung „ROM-Routine“ schon sagt, liegt dieses Unterprogramm im ROM (Festwertspeicher) und ist damit sofort nach dem Einschalten des Rechners nutzbar. Die hier verwendeten Routinen werden normalerweise vom Basic-Interpreter des C64 genutzt, um Zeilennummern bzw. Zeichen auf dem Bildschirm auszugeben. Die Routine PRINT gibt eine gerade Zahl auf dem Bildschirm aus, dabei muß sich das High-Byte im Akku und das Low-Byte im X-Register befinden. Die Routine BSOUT gibt ein beliebiges Zeichen auf dem Schirm aus. Wir verwenden dies, um einen Zeilenvorschub zu erreichen. Wir werden später noch öfter auf diese und auch andere ROM-Routinen stoßen und diese dann etwas genauer betrachten.

Ebenfalls neu in unserem Beispiel ist der Befehl TAX. Dieser Befehl überträgt den Inhalt von dem Akkumulator nach dem X-Register. In unserer alphabetischen Auflistung wird noch näher darauf eingegangen.

Nun zur Funktion unseres Programmes: Als erstes wird der Akku mit dem Inhalt der Speicherstelle 832 geladen. Wir haben absichtlich diese Speicherstelle verwendet, da sich der Inhalt mittels Basic-POKE Befehl schnell ändern läßt und so mehrere Tests durchführbar sind.

Nachdem dieser Wert in den Akku gela-

den wurde, wird dieser auf dem Schirm ausgegeben. Nun wird der Akku nochmals mit unserer Zahl geladen, jedoch wird diesmal der Akkuinhalt durch den Befehl LSR manipuliert und das Ergebnis ausgegeben. Das Gleiche wird noch einmal mit zwei aufeinanderfolgenden LSR-Befehlen durchgeführt.

Nach der Eingabe unseres Beispiels können wir folgende Tests durchführen:

```
POKE 832,200:SYS 49152 <return>
200
100
50
```

```
POKE 832,60:SYS 49152 <return>
60
30
15
```

```
POKE 832,12:SYS 49152 <return>
12
6
3
```

Wie aus den Beispielen zu erkennen ist, läßt sich der LSR-Befehl auch sehr gut zum Teilen von Zahlen verwenden. Eine andere Anwendung wäre beispielsweise der Test von Bit 0. Durch die Befehlskombination LSR und BCS läßt sich diese sehr schnell bewerkstelligen.

NOP No Operation (keine Ausführung)

Einer der hilfreichsten Befehle ist der NOP-Befehl. Er macht nämlich garnichts und ist mit dem Basic-Befehl REM vergleichbar. Oft muß ein älteres Maschinenprogramm verändert werden, wobei einige Lücken

entstehen können. Die Herausnahme eines Sprungbefehles würde beispielsweise 3 Byte freilegen. Diese Lücken können nun mit drei NOP-Befehlen aufgefüllt werden. Geschieht dies nicht, kann der Micro-

prozessor in einen undefinierbaren Zustand geraten und nur durch Aus- und Einschalten wieder in Gang gebracht werden.

HEX	Adressierung	Bytes	Syntax
EA	Impliziert	1	NOP

7	6	5	4	3	2	1	0
N	V	B	D	I	Z	C	
0							x x

Literatur:

6502 Microcomputer-Programmierung, Peter Heuer, Hofacker Verlag
 6502 Programmierung in Assembler, Lance, A. Leventhal
 64 Intern, Angershausen, Becker, Englisch, Gerits, Data-Becker Buch
 6502/65c02, Christian Persson, Heinz Heise Verlag

Beispiele:
 100 LDA #1 ; Zeichencode für A
 110 NOP ; keine Operation
 120 NOP ; keine Operation
 130 NOP ; keine Operation
 140 STA 1024 ; Bildschirmanfänger
 150 RTS

Ein Labyrinthspiel frei nach dem Motto: „Fressen oder gefressen werden!“

„Smily“

oder in gut Deutsch

„Der Lächler“

auf dem CPC-464

Das Programm „Smily“ wird mit RUN gestartet. Nach dem Erscheinen des Titels beginnt die Musik zu spielen. Durch Drücken einer Taste können Sie eine

Spielstufe auswählen:

1. Anfänger (für Unerfahrene)
2. Könnler (für geübtere Spieler)
3. Profi (eine besondere Variante des Spiels)

Unterschiede innerhalb der 3 Spielstufen:

	Reservesmilies	Spieltempo	Herzchen	Labyrinth
Anfänger:	4	langsam	10	sichtbar
Könnler:	2	mittel	8	sichtbar
Profi:	2	schnell	6	unsichtbar

Wenn Sie die Profi-Spielstufe erreicht haben, ist das Labyrinth nur für kurze Zeit sichtbar. Wenn Smily gegen eine Wand läuft, blinkt es kurz auf.

Nach dem Aufbau des ersten Labyrinths erscheint Smily grinsend in der unteren Hälfte des Irrgartens und wartet nur darauf von Ihnen gesteuert zu werden. (Die Steuerung erfolgt mittels Joystick 1 oder der Cursortasten). Smily kann sich nach oben und unten bewegen, sowie nach rechts und links, eine diagonale Fortbewegungsweise ist ihm jedoch nicht mög-

lich. Je mehr Punkte Smily verschlingt, desto höher ist die Gutschrift, die Sie erhalten. Die Punktejagd gerät – wie nicht anders zu erwarten – zu einem gefährbringenden Unterfangen, denn unser Held wird vom Gespenst Leo auf das eifrigste verfolgt. Leo ist wesentlich schneller als Smily und so für dessen Leben eine ernste Bedrohung. Natürlich hat sich unser Autor einen Ausweg für Smily einfallen lassen. Dieser besteht aus im Labyrinth verstecktem Lebkuchen. Wenn Smily eine dieser Leckereien verschlingt, ist er für kurze Zeit

selbst in der Lage, seinem Feind den Gar aus zu machen. Solange Smily seine Wunderkraft besitzt, blinkt das Labyrinth. Gelingt es Smily, seinen Widersacher zu fressen, so bekommen Sie hohe Zusatzpunkte. Es bedarf aber einiges an Einfallsreichtum, um Leo zu erwischen. Sobald das Labyrinth aufhört zu blinken, kehrt sich das Ganze um und Smily ist wieder der Gejagte. Falls Smily erwischt wird, stellt er sich auf den Kopf und läuft vor Wut lila an. Nun muß ein Ersatz-Smily her. Am unteren Bildschirmrand befindet sich eine Anzeige für Ersatzsmilies und die erreichte Punktzahl. Sind die gesamten Smilies verbraucht, ist das Spiel beendet. Wenn Sie ein neues Abenteuer mit Smily und Leo erleben wollen, drücken Sie bitte die Joystick-Feuertaste oder COPY. Nach einer kurzen Pause erscheint dann wieder das Titelbild. Andernfalls drücken Sie bitte „E“ für Ende.

1. Teil:

```

10 ' Ladeprogramm fuer Smily
20 ' copyright 1985 by
30 ' M E D I - Software
40 ' M. Meier & C. Disch
50 SYMBOL AFTER 32
60 MODE 1:INK 0,1:INK 1,24:INK 2,20:INK
3,6:BORDER 1:PAPER 0:PEN 1:CLS
70 CLS:PRINT
80 PRINT" Schneider 20K Microcomputergam
e":PRINT:PRINT" $1985 M E D I - Software
Meier & Disch "STRING$(38,208):PRINT
90 PRINT" presents: Smily (and the ghost
)":PRINT
100 PRINT" Ready to load":PRINT
110 LOCATE 2,13:PRINT"loading SMILY":PRI
NT:PRINT:PRINT" Bitte warten !"
120 RUN"!smilypr"
130 END
    
```

CHECK V1:PRUEFSUMMEN:

ZEILENR.:	SUMMEN
10-	20: 49072
30-	40: 35338
50-	60: 35135
70-	80: 63041
90-	100: 74690
110-	120: 21537
130-	: 152

GESAMTCHECKSUMME: 16669

*Checksummen
für den
1. Teil:*

```

10 ' *****
20 ' *
30 ' *          S M I L Y
40 ' *          *
50 ' * von M E D I - Software *
60 ' *      copyright 1985
70 ' *      M. Meier & C. Disch
80 ' *
90 ' *****
    
```

programme

```
100 DEFINT a-o,q-z
110 FOR i=0 TO 79:KEY DEF i,0:NEXT i
120 RANDOMIZE TIME
130 DIM ta(320),tc(322),M(20,21),s1(8),s
2(8),s3(23),ton1(24),ton2(36)
140 REM *** Tonberechnung ***
150 FOR i=1 TO 24:ton1(i)=ROUND(125000/(
261.626*(2^(1/12))^i)):NEXT i
160 FOR i=0 TO 36:ton2(i)=ROUND(125000/(
130.813*(2^(1/12))^i)):NEXT i
170 RESTORE 4630
180 REM *** Sound aus DATAs lesen ***
190 FOR i=1 TO 320
200 da=da-1:IF da>0 THEN ta(i)=0 ELSE RE
AD da,ton:ta(i)=ton1(ton)
210 NEXT i
220 FOR i=3 TO 319 STEP 4
230 da=da-1:IF da=0 THEN READ da,dur
240 IF dur=1 THEN tc(i)=ton2(12):tc(i+1)
=ton2(16):tc(i+2)=ton2(19):tc(i+3)=ton2(
16):GOTO 260
250 IF dur=2 THEN tc(i)=ton2(7):tc(i+1)=
ton2(11):tc(i+2)=ton2(14):tc(i+3)=ton2(1
1)ELSE tc(i)=ton2(17):tc(i+1)=ton2(21):t
c(i+2)=ton2(24):tc(i+3)=ton2(21)
260 NEXT i
270 FOR i=1 TO 8:READ s1(i):NEXT i:FOR i
=1 TO 8:READ s2(i):NEXT i:FOR i=1 TO 23:
READ s3(i):NEXT i
280 REM *** Huelkurven fuer Sound ***
290 ENV 1,3,5,1,1,0,10,2,-1,9,13,-1,20:E
NV 2,10,0,100:ENV 3,7,2,1,1,0,2,1,-14,1:
ENV 4,15,1,5,1,-15,1:ENV 5,1,15,3,8,-1,5
,7,-1,1:ENV 6,3,5,1,1,0,10,1,-15,1
300 ENT-1,1,1,5,2,-1,5,1,1,5:ENT-2,1,-1,
4,2,1,3,1,-1,4:ENT-5,20,-1,1,20,1,1,1,0,
10:ENT-6,1,1,1,1,0,2,2,-1,1,1,0,2,1,1,1:
ENT-7,1,-1,3,2,1,5,1,-1,3
310 GOTO 590
320 REM *** Sound spielen ***
330 tonz=tonz+1:IF tonz>320 THEN tonz=1
340 ta=ta(tonz)
350 SOUND 129,tc(tonz),0,0,2,2
360 IF ta=0 THEN RETURN
370 SOUND 132,ta,0,0,1,1:RETURN
380 REM *** Titelbildsmily zeichnen ***
390 DEG
400 ORIGIN 320,208
410 FOR i=-127 TO 127 STEP 2
420 y=SQR(16384-i*i)
430 PLOT i*2,y,1:DRAW i*2,-y,1
440 NEXT i
450 FOR i=-192 TO 192 STEP 4
460 y=SQR(37249-i*i)
470 PLOT i,0,0:DRAW i,-y/2,0
480 NEXT i
490 FOR i=-23 TO 23 STEP 2
500 y=SQR(576-i*i)
510 PLOT i*2-95,y+48,0:DRAW i*2-95,-y+48
,0
520 PLOT i*2+96,y+48,0:DRAW i*2+96,-y+48
,0
530 NEXT i
```

```
540 PLOT 1000,1000,7:MOVE-72,-40:TAG:PRI
NT smily$;:TAGOFF
550 RETURN
560 tc(1)=tc(321):tc(2)=tc(322):ENV 2,2,
6,2,1,0,20,12,-1,25
570 RETURN
580 REM *** Zeichen umdefinieren ***
590 SYMBOL AFTER 32
600 SYMBOL 33,&EE,&FE,&FE,&6C,&FE,&FE,&E
E
610 SYMBOL 64,&3C,&42,&99,&A1,&A1,&99,&4
2,&3C
620 SYMBOL 88,&0,&0,&0,&18,&18
630 SYMBOL 96,&18,&3C,&7E,&DB,&FF,&FF,&2
4,&E7
640 SYMBOL 106,0,0,0,0,0,&66,&66
650 SYMBOL 112,&0,&FF,&0,&0,&0,&0,&FF
660 SYMBOL 113,&42,&42,&42,&42,&42,&42,&
42,&42
670 SYMBOL 116,&0,&7E,&42,&42,&42,&42,&4
2,&42
680 SYMBOL 117,&42,&42,&42,&42,&42,&42,&
7E
690 SYMBOL 118,&0,&FE,&2,&2,&2,&2,&C2,&4
2
700 SYMBOL 119,&0,&7F,&40,&40,&40,&40,&4
3,&42
710 SYMBOL 120,&42,&C2,&2,&2,&2,&2,&FE
720 SYMBOL 121,&42,&43,&40,&40,&40,&40,&
7F
730 SYMBOL 122,&42,&43,&40,&40,&40,&40,&
43,&42
740 SYMBOL 123,&42,&C2,&2,&2,&2,&2,&C2,&
42
750 SYMBOL 124,&42,&C3,&0,&0,&0,&0,&FF
760 SYMBOL 125,&0,&FF,&0,&0,&0,&0,&C3,&4
2
770 SYMBOL 126,&0,&7E,&42,&42,&42,&42,&7
E
780 SYMBOL 128,&3C,&7E,&C3,&81,&FF,&DB,&
7E,&3C
790 SYMBOL 129,&18,&A4,&7B,&21,&22,&A4,&
7B
800 SYMBOL 130,&0,&0,&0,&6C,&92,&92,&93
810 SYMBOL 131,&0,&8,&0,&10,&30,&50,&BF
820 SYMBOL 132,&20,&50,&50,&61,&42,&CC,&
30
830 SYMBOL 133,&0,&0,&0,&C9,&4A,&7C,&8,&
70
840 SYMBOL 143,&66,&99,&81,&81,&42,&24,&
18
850 REM *** Titelbild ***
860 smily$=CHR$(129)+CHR$(130)+CHR$(131)
+CHR$(132)+CHR$(133)
870 INK 0,0:INK 1,0:INK 3,0:INK 7,0:INK
12,0:INK 14,0:INK 15,0:PAPER 0:MODE 0:CL
S: BORDER 0
880 SPEED INK 48,48
890 PEN 3:PRINT:PRINT:PRINT" M E D I - S
OFTWARE":PRINT
900 GOSUB 390
910 PEN 14:LOCATE 1,22:PRINT" @ 19
85"
```



```

920 PEN 15:PRINT " M.Meier C.Disch"
930 PEN 12:LOCATE 1,1:PRINT"!!!!!!!!!!!!!!
!!!!!!!!!!"
940 FOR i=1 TO 24:LOCATE 1,i:PRINT"!":LO
CATE 20,i:PRINT"!":NEXT i
950 LOCATE 1,25:PRINT"!!!!!!!!!!!!!!
!!";:LOCATE 1,1
960 INK 1,24:INK 3,6:INK 7,17:INK 12,18:
INK 14,0,8:INK 15,26,0
970 AFTER 35,2 GOSUB 560
980 EVERY 12,3 GOSUB 330
990 uw$=INKEY$:IF uw$<>" "THEN 990
1000 IF INKEY$=""THEN 1000
1010 INK 13,26
1020 REM *** weiter Zeichen umdefinieren
***
1030 SYMBOL 104,&3C,&7E,&DB,&FF,&B1,&C3,
&66,&3C
1040 SYMBOL 105,&3C,&7E,&DB,&FF,&BD,&C3,
&7E,&3C
1050 SYMBOL 114,&0,&FE,&2,&2,&2,&2,&FE
1060 SYMBOL 115,&0,&7F,&40,&40,&40,&40,&
7F
1070 REM *** Schwierigkeitsgrad waehlen *
**
1080 MODE 1:INK 2,20:INK 3,0:PAPER 3:PEN
2:CLS
1090 PRINT:PRINT:PRINT:PRINT TAB(11)"S P
I E L S T U F E":PRINT TAB(11)STRING$(1
9,CHR$(154)):PRINT:PRINT:PRINT:PRINT TAB
(22)"j"
1100 PRINT TAB(14)"A -> ANFANGER":PRINT:
PRINT TAB(20)"j"
1110 PRINT TAB(14)"K -> KONNER":PRINT:PR
INT
1120 PRINT TAB(14)"P -> PROFI"
1130 I$=UPPER$(INKEY$):IF I$=""THEN 1130
1140 IF I$="A"THEN FA=11:F=0:punb=500:ag
=100:apz=10:GOTO 1180
1150 IF I$="K"THEN FA=11:F=2:punb=1000:a
g=50:apz=8:GOTO 1180
1160 IF I$="P"THEN FA=0:F=2:punb=2000:ag
=25:apz=6 ELSE 1130
1170 REM *** Musikbegleitung Abfrage ***
1180 CLS:LOCATE 4,13:PRINT"MIT MUSIKBEGL
EITUNG ? J ODER N"
1190 i$=UPPER$(INKEY$):IF i$=""THEN 1190
1200 IF i$="J"THEN ENV 1,2,4,1,1,0,10,1,
-1,9,7,-1,20:ENV 2,2,3,2,1,0,20,6,-1,25:
GOTO 1230
1210 IF i$="N"THEN LA=0:LC=0:ENV 1,1,0,1
00:ENV 2,1,0,100 ELSE 1190
1220 REM *** Labyrinth aufbauen ***
1230 MODE 0:PAPER 0:SPEED INK 10,10
1240 CLS:R=R+1:LB=LB+1
1250 IF LB>10 THEN APZ=APZ-1:LB=1
1260 IF APZ<1 THEN APZ=1
1270 ON lb GOTO 1280,1290,1300,1310,1320
,1330,1340,1350,1360,1370
1280 RESTORE 1440:GOTO 1380
1290 RESTORE 1640:GOTO 1380
1300 RESTORE 1840:GOTO 1380
1310 RESTORE 2040:GOTO 1380

```

```

1320 RESTORE 2240:GOTO 1380
1330 RESTORE 2440:GOTO 1380
1340 RESTORE 2640:GOTO 1380
1350 RESTORE 2840:GOTO 1380
1360 RESTORE 3040:GOTO 1380
1370 RESTORE 3240
1380 INK 11,fa:INK 12,18:LOCATE 8,1:PEN
7:PRINT smily$:PRINT
1390 fpu=0:FOR I=3 TO 21:FOR J=1 TO 19:R
EAD L$
1400 IF L$="X"THEN PEN 12:FPU=FPU+1:M(J,
I)=88 ELSE PEN 11:M(J,I)=1
1410 PRINT L$;:NEXT J:PRINT:NEXT I
1420 REM *** DATAs fuer Labyrinth ***
1430 REM *** 1. Labyrinth ***
1440 DATA " ", " ",w,p,p,p,p,p,p,p,p,p,
p,p,p,v, " ", " "
1450 DATA " ", " ",q,x,x,x,x,x,x,x,x,x,
x,x,x,q, " ", " "
1460 DATA " ", " ",q,x,t,x,s,p,p,p,p,p,r,
x,t,x,q, " ", " "
1470 DATA " ", " ",q,x,q,x,x,x,x,x,x,x,
x,q,x,q, " ", " "
1480 DATA " ", " ",q,x,q,x,w,p,r,x,s,p,v,
x,q,x,q, " ", " "
1490 DATA " ", " ",q,x,q,x,q,x,x,x,x,q,
x,q,x,q, " ", " "
1500 DATA " ", " ",q,x,q,x,q,x,s,p,r,x,q,
x,q,x,q, " ", " "
1510 DATA " ", " ",q,x,q,x,q,x,x,x,x,q,
x,q,x,q, " ", " "
1520 DATA s,p,x,x,q,x,u,x,w,p,v,x,u,x,q,
x,y,p,r
1530 DATA x,x,x,x,q,x,x,x,q, " ",q,x,x,x,
q,x,x,x,x
1540 DATA s,p,v,x,q,x,t,x,y,p,x,x,t,x,q,
x,w,p,r
1550 DATA " ", " ",q,x,q,x,q,x,x,x,x,q,
x,q,x,q, " ", " "
1560 DATA " ", " ",q,x,q,x,q,x,s,p,r,x,q,
x,q,x,q, " ", " "
1570 DATA " ", " ",q,x,q,x,q,x,x,x,x,q,
x,q,x,q, " ", " "
1580 DATA " ", " ",q,x,q,x,y,p,r,x,s,p,x,
x,q,x,q, " ", " "
1590 DATA " ", " ",q,x,q,x,x,x,x,x,x,x,
x,q,x,q, " ", " "
1600 DATA " ", " ",q,x,u,x,s,p,p,p,p,p,r,
x,u,x,q, " ", " "
1610 DATA " ", " ",q,x,x,x,x,x,x,x,x,x,
x,x,x,q, " ", " "
1620 DATA " ", " ",y,p,p,p,p,p,p,p,p,p,
p,p,p,x, " ", " "
1630 REM *** 2. Labyrinth ***
1640 DATA " ", " ",w,p,p,p,p,p,p,p,p,p,
p,p,p,v, " ", " "
1650 DATA " ", " ",q,x,x,x,u,x,x,x,x,u,
x,x,x,q, " ", " "
1660 DATA " ", " ",q,x,t,x,x,x,s,p,r,x,x,
x,t,x,q, " ", " "
1670 DATA " ", " ",q,x,z,p,r,x,x,x,x,s,
p,c,x,q, " ", " "
1680 DATA " ", " ",q,x,q,x,x,x,t,x,t,x,x,

```

programme

```

X,q,X,q," ",""
1690 DATA " "," " ,q,X,q,X,s,p,ç,X,z,p,r,
X,q,X,q," ",""
1700 DATA " "," " ,q,X,q,X,X,X,u,X,u,X,X,
X,q,X,q," ",""
1710 DATA " "," " ,q,X,q,X,t,X,X,X,X,X,t,
X,q,X,q," ",""
1720 DATA " "," " ,q,X,q,X,q,X,w,p,v,X,q,
X,q,X,q," ",""
1730 DATA " "," " ,q,X,q,X,q,X,q," " ,q,X,
q,X,q,X,q," ",""
1740 DATA " "," " ,q,X,q,X,q,X,y,p,x,X,q,
X,q,X,q," ",""
1750 DATA " "," " ,q,X,q,X,u,X,X,X,X,X,u,
X,q,X,q," ",""
1760 DATA " "," " ,q,X,q,X,X,X,t,X,t,X,X,
X,q,X,q," ",""
1770 DATA " "," " ,q,X,q,X,s,p,ç,X,z,p,r,
X,q,X,q," ",""
1780 DATA " "," " ,q,X,q,X,X,X,u,X,u,X,X,
X,q,X,q," ",""
1790 DATA " "," " ,q,X,z,p,r,X,X,X,X,X,s,
p,ç,X,q," ",""
1800 DATA s,p,x,X,u,X,X,X,s,p,r,X,X,X,u,
X,y,p,r
1810 DATA X,X,X,X,X,X,t,X,X,X,X,X,t,X,X,
X,X,X,X
1820 DATA s,p,p,p,p,p,"!" ,p,p,p,p,p,"!" ,
p,p,p,p,p,r
1830 REM *** 3. Labyrinth ***
1840 DATA s,p,p,p,p,p,p,p,p,p,p,p,p,p,
p,p,p,r
1850 DATA X,X,X,X,X,X,X,X,X,X,X,X,X,X,
X,X,X,X
1860 DATA w,p,p,p,ç,p,p,p,r,X,s,p,p,p,ç,
p,p,p,v
1870 DATA q,X,X,X,q,X,X,X,X,X,X,X,q,
X,X,X,q
1880 DATA q,X,t,X,q,X,w,p,r,X,s,p,v,X,q,
X,t,X,q
1890 DATA q,X,q,X,u,X,q,X,X,X,X,q,X,u,
X,q,X,q
1900 DATA q,X,q,X,X,X,q,X,s,p,r,X,q,X,X,
X,q,X,q
1910 DATA q,X,y,r,X,s,x,X,X,X,X,y,r,X,
s,x,X,q
1920 DATA q,X,X,X,X,X,X,w,p,v,X,X,X,X,
X,X,X,q
1930 DATA z,p,r,X,s,p,r,X,q," " ,q,X,s,p,
r,X,s,p,ç
1940 DATA q,X,X,X,X,X,X,y,p,x,X,X,X,X,
X,X,X,q
1950 DATA q,X,w,r,X,s,v,X,X,X,X,w,r,X,
s,v,X,q
1960 DATA q,X,q,X,X,X,q,X,s,p,r,X,q,X,X,
X,q,X,q
1970 DATA q,X,q,X,t,X,q,X,X,X,X,q,X,t,
X,q,X,q
1980 DATA q,X,u,X,q,X,y,p,r,X,s,p,x,X,q,
X,u,X,q
1990 DATA q,X,X,X,q,X,X,X,X,X,X,X,q,
X,X,X,q
2000 DATA y,p,p,p,"!" ,p,p,p,r,X,s,p,p,p,

```

```

"!" ,p,p,p,x
2010 DATA X,X,X,X,X,X,X,X,X,X,X,X,X,
X,X,X,X
2020 DATA s,p,p,p,p,p,p,p,p,p,p,p,p,
p,p,p,r
2030 REM *** 4. Labyrinth ***
2040 DATA s,p,p,p,p,p,ç,p,p,p,p,p,ç,p,p,
p,ç,p,r
2050 DATA X,X,X,X,X,X,q,X,X,X,X,q,X,X,
X,q,X,X
2060 DATA s,p,v,X,t,X,u,X,s,p,r,X,u,X,t,
X,q,X,t
2070 DATA " "," " ,q,X,u,X,X,X,X,X,X,
X,u,X,q,X,q
2080 DATA " "," " ,q,X,X,X,~X,~X,~X,~X,
X,X,X,q,X,q
2090 DATA " "," " ,z,p,r,X,X,X,X,X,X,
X,s,p,ç,X,q
2100 DATA " "," " ,q,X,X,X,w,r,X,~X,s,v,
X,X,X,q,X,q
2110 DATA " "," " ,q,X,s,p,x,X,X,X,X,y,
p,r,X,q,X,q
2120 DATA " "," " ,q,X,X,X,X,w,p,v,X,X,
X,X,X,q,X,q
2130 DATA " "," " ,z,p,r,X,~X,q," " ,q,X,
~X,s,p,ç,X,q
2140 DATA " "," " ,q,X,X,X,X,y,p,x,X,X,
X,X,X,q,X,q
2150 DATA " "," " ,q,X,s,p,v,X,X,X,X,w,
p,r,X,q,X,q
2160 DATA " "," " ,q,X,X,X,y,r,X,~X,s,x,
X,X,X,q,X,q
2170 DATA " "," " ,z,p,r,X,X,X,X,X,X,
X,s,p,ç,X,q
2180 DATA " "," " ,q,X,X,X,~X,~X,~X,~X,
X,X,X,q,X,q
2190 DATA " "," " ,q,X,t,X,X,X,X,X,X,
X,t,X,q,X,q
2200 DATA " "," " ,q,X,u,X,t,X,s,p,r,X,t,
X,u,X,u,X,q
2210 DATA " "," " ,q,X,X,X,q,X,X,X,X,q,
X,X,X,X,q
2220 DATA " "," " ,y,p,p,p,"!" ,p,p,p,p,
"!" ,p,p,p,p,p,x
2230 REM *** 5. Labyrinth ***
2240 DATA w,p,p,p,p,p,p,p,p,p,v," " ,
" " " " " " " " " "
2250 DATA q,X,X,X,X,X,X,X,q," " ,
" " " " " " " " " "
2260 DATA q,X,w,p,p,p,p,v,X,z,p,p,p,
p,v," " ,
" " "
2270 DATA q,X,q,X,X,X,X,u,X,u,X,X,X,
X,q," " ,
" " "
2280 DATA q,X,q,X,t,X,~X,X,X,X,~X,t,
X,q," " ,
" " "
2290 DATA q,X,q,X,q,X,X,t,X,t,X,X,q,
X,q," " ,
" " "
2300 DATA q,X,q,X,z,r,X,s,x,X,y,r,X,s,ç,
X,q," " ,
" " "
2310 DATA q,X,q,X,u,X,X,X,X,X,X,u,
X,q," " ,
" " "
2320 DATA u,X,q,X,X,X,~X,w,p,v,X,~X,X,
X,z,p,r

```

```

2330 DATA X,X,z,p,r,X,X,X,q," ",q,X,X,X,
s,p,ç,X,X
2340 DATA s,p,ç,X,X,X,~,X,y,p,x,X,~,X,X,
X,q,X,t
2350 DATA " "," " ,q,X,t,X,X,X,X,X,X,X,
X,t,X,q,X,q
2360 DATA " "," " ,q,X,z,r,X,s,v,X,w,r,X,
s,ç,X,q,X,q
2370 DATA " "," " ,q,X,q,X,X,X,u,X,u,X,X,
X,q,X,q,X,q
2380 DATA " "," " ,q,X,u,X,~,X,X,X,X,X,~,
X,u,X,q,X,q
2390 DATA " "," " ,q,X,X,X,X,X,t,X,t,X,X,
X,X,X,q,X,q
2400 DATA " "," " ,y,p,p,p,p,p,ç,X,y,p,p,
p,p,p,x,X,q
2410 DATA " "," " " " " " " " " " " " " "
",q,X,X,X,X,X,X,X,X,X,q
2420 DATA " "," " " " " " " " " " " " " "
",y,p,p,p,p,p,p,p,p,p,x
2430 REM *** 6. Labyrinth ***
2440 DATA s,p,ç,p,p,p,p,p,p,ç,p,p,p,p,p,
p,ç,p,r
2450 DATA X,X,q,X,X,X,X,X,q,X,X,X,X,X,
X,q,X,X
2460 DATA t,X,q,X,w,r,X,~,X,u,X,~,X,s,v,
X,q,X,t
2470 DATA q,X,q,X,u,X,X,X,X,X,X,X,X,u,
X,q,X,q
2480 DATA q,X,q,X,X,X,w,r,X,~,X,s,v,X,X,
X,q,X,q
2490 DATA q,X,q,X,s,p,x,X,X,X,X,y,p,r,
X,q,X,q
2500 DATA q,X,q,X,X,X,X,~,X,~,X,X,X,X,
X,q,X,q
2510 DATA q,X,z,p,r,X,~,X,X,X,X,X,~,X,s,
p,ç,X,q
2520 DATA q,X,u,X,X,X,X,X,w,p,v,X,X,X,X,
X,u,X,q
2530 DATA q,X,X,X,~,X,s,p,ç," ",z,p,r,X,
~,X,X,X,q
2540 DATA q,X,t,X,X,X,X,X,y,p,x,X,X,X,
X,t,X,q
2550 DATA q,X,z,p,r,X,~,X,X,X,X,X,~,X,s,
p,ç,X,q
2560 DATA q,X,q,X,X,X,X,X,~,X,~,X,X,X,X,
X,q,X,q
2570 DATA q,X,q,X,s,p,v,X,X,X,X,X,w,p,r,
X,q,X,q
2580 DATA q,X,q,X,X,X,y,r,X,~,X,s,x,X,X,
X,q,X,q
2590 DATA q,X,q,X,t,X,X,X,X,X,X,X,t,
X,q,X,q
2600 DATA u,X,q,X,y,r,X,~,X,t,X,~,X,s,x,
X,q,X,u
2610 DATA X,X,q,X,X,X,X,X,q,X,X,X,X,X,
X,q,X,X
2620 DATA s,p,"!",p,p,p,p,p,p,"!",p,p,p,
p,p,p,"!",p,r
2630 REM *** 7. Labyrinth ***
2640 DATA " ",w,p,p,p,p,p,p,p,p,p,p,p,
p,p,p,v," "
2650 DATA " ",q,X,X,X,X,X,X,X,X,X,X,X,

```

```

X,X,X,q," "
2660 DATA " ",q,X,s,p,ç,p,p,p,p,p,p,p,ç,
p,r,X,q," "
2670 DATA " ",q,X,X,X,u,X,X,X,X,X,X,u,
X,X,X,q," "
2680 DATA " ",q,X,t,X,X,X,t,X,~,X,t,X,X,
X,t,X,q," "
2690 DATA " ",q,X,z,r,X,s,ç,X,X,X,z,r,X,
s,ç,X,q," "
2700 DATA " ",q,X,q,X,X,X,u,X,~,X,u,X,X,
X,q,X,q," "
2710 DATA " ",q,X,q,X,t,X,X,X,X,X,X,t,
X,q,X,q," "
2720 DATA s,x,X,q,X,y,r,X,w,p,v,X,s,x,X,
q,X,y,r
2730 DATA X,X,X,q,X,X,X,X,q," ",q,X,X,X,
X,q,X,X,X
2740 DATA s,v,X,q,X,w,r,X,y,p,x,X,s,v,X,
q,X,w,r
2750 DATA " ",q,X,q,X,u,X,X,X,X,X,X,u,
X,q,X,q," "
2760 DATA " ",q,X,q,X,X,X,t,X,~,X,t,X,X,
X,q,X,q," "
2770 DATA " ",q,X,z,r,X,s,ç,X,X,X,z,r,X,
s,ç,X,q," "
2780 DATA " ",q,X,u,X,X,X,u,X,~,X,u,X,X,
X,u,X,q," "
2790 DATA " ",q,X,X,X,t,X,X,X,X,X,X,t,
X,X,X,q," "
2800 DATA " ",q,X,s,p,"!",p,p,p,p,p,p,p,
"!",p,r,X,q," "
2810 DATA " ",q,X,X,X,X,X,X,X,X,X,X,X,
X,X,X,q," "
2820 DATA " ",y,p,p,p,p,p,p,p,p,p,p,p,
p,p,p,x," "
2830 REM *** 8. Labyrinth ***
2840 DATA w,p,p,p,p,p,ç,p,p,p,ç,p,p,p,ç,p,
p,p,p,v
2850 DATA q,X,X,X,X,u,X,X,X,q,X,X,X,u,X,
X,X,X,q
2860 DATA q,X,w,r,X,X,X,~,X,u,X,~,X,X,X,
s,v,X,q
2870 DATA q,X,u,X,X,t,X,X,X,X,X,X,t,X,
X,u,X,q
2880 DATA q,X,X,X,w,x,X,s,r,X,s,r,X,y,v,
X,X,X,q
2890 DATA q,X,~,X,u,X,X,X,X,X,X,X,u,
X,~,X,q
2900 DATA q,X,X,X,X,X,~,X,~,X,~,X,~,X,X,
X,X,X,q
2910 DATA y,p,p,p,r,X,X,X,X,X,X,X,s,
p,p,p,x
2920 DATA X,X,X,X,X,X,t,X,w,p,v,X,t,X,X,
X,X,X,X
2930 DATA s,r,X,~,X,s,ç,X,q," ",q,X,z,r,
X,~,X,s,r
2940 DATA X,X,X,X,X,X,u,X,y,p,x,X,u,X,X,
X,X,X,X
2950 DATA w,p,p,p,r,X,X,X,X,X,X,X,s,
p,p,p,v
2960 DATA q,X,X,X,X,X,~,X,~,X,~,X,~,X,X,
X,X,X,q
2970 DATA q,X,~,X,t,X,X,X,X,X,X,t,

```

programme

```

X,~,X,q
2980 DATA q,X,X,X,y,v,X,s,r,X,s,r,X,w,x,
X,X,X,q
2990 DATA q,X,t,X,X,u,X,X,X,X,X,X,u,X,
X,t,X,q
3000 DATA q,X,y,r,X,X,X,~,X,t,X,~,X,X,X,
s,x,X,q
3010 DATA q,X,X,X,X,t,X,X,X,q,X,X,X,t,X,
X,X,X,q
3020 DATA y,p,p,p,p,"i",p,p,p,"i",p,p,p,
"i",p,p,p,p,x
3030 REM *** 9. Labyrinth ***
3040 DATA w,p,p,p,p,p,p,p,p,p,p,p,p,p,
p,p,p,v
3050 DATA u,X,X,X,X,X,X,X,X,X,X,X,X,X,
X,X,X,u
3060 DATA X,X,s,p,p,p,p,p,p,p,p,p,p,p,
p,r,X,X
3070 DATA t,X,X,X,X,X,X,X,X,X,X,X,X,X,
X,X,X,t
3080 DATA z,p,r,X,w,p,p,p,p,p,p,p,p,v,
X,s,p,c
3090 DATA q,X,X,X,u,X,X,X,X,X,X,X,X,u,
X,X,X,q
3100 DATA q,X,t,X,X,X,t,X,~,X,~,X,t,X,X,
X,t,X,q
3110 DATA q,X,y,v,X,s,x,X,X,X,X,y,r,X,
w,x,X,q
3120 DATA q,X,X,u,X,X,X,X,w,p,v,X,X,X,
u,X,X,q
3130 DATA z,r,X,X,X,~,X,s,c," ",z,r,X,~,
X,X,X,s,c
3140 DATA q,X,X,t,X,X,X,X,y,p,x,X,X,X,
t,X,X,q
3150 DATA q,X,w,x,X,s,v,X,X,X,X,w,r,X,
y,v,X,q
3160 DATA q,X,u,X,X,X,u,X,~,X,~,X,u,X,X,
X,u,X,q
3170 DATA q,X,X,X,t,X,X,X,X,X,X,X,X,t,
X,X,X,q
3180 DATA z,p,r,X,y,p,p,p,p,p,p,p,p,x,
X,s,p,c
3190 DATA u,X,X,X,X,X,X,X,X,X,X,X,X,X,
X,X,X,u
3200 DATA X,X,s,p,p,p,p,p,p,p,p,p,p,p,
p,r,X,X
3210 DATA t,X,X,X,X,X,X,X,X,X,X,X,X,X,
X,X,X,t
3220 DATA y,p,p,p,p,p,p,p,p,p,p,p,p,p,
p,p,p,x
3230 REM *** 10. Labyrinth ***
3240 DATA " ", " ", " ", " ", " ", " ", " ", "w,p,p,p,},
p,p,p,v," ", " ", " ", " ", " ", " "
3250 DATA " ", " ", " ", " ", " ", " ", " ",q,X,X,X,u,
X,X,X,q," ", " ", " ", " ", " ", " "
3260 DATA " ", " ", " ", " ", " ", " ", "w,x,X,~,X,X,X,
~,X,y,v," ", " ", " ", " ", " ", " "
3270 DATA " ", " ", " ", " ", " ", " ", "q,X,X,X,X,~,X,
X,X,X,q," ", " ", " ", " ", " ", " "
3280 DATA " ", " ", " ", " ", " ", " ", "w,x,X,s,r,X,X,X,s,
r,X,y,v," ", " ", " ", " ", " ", " "
3290 DATA " ", " ", " ", " ", " ", " ", "q,X,X,X,X,X,t,X,X,
X,X,X,q," ", " ", " ", " ", " ", " "

```

```

3300 DATA " ", " ", "w,x,X,~,X,s,p,"i",p,r,
X,~,X,y,v," ", " ", " "
3310 DATA " ", " ", "q,X,X,X,X,X,X,X,X,X,
X,X,X,q," ", " ", " "
3320 DATA " ", "w,x,X,~,X,~,X,w,p,v,X,~,X,
~,X,y,v," ", " ", " "
3330 DATA " ", "q,X,X,X,X,X,X,q," ", "q,X,X,
X,X,X,q," ", " ", " "
3340 DATA w,x,X,~,X,s,r,X,y,p,x,X,s,r,X,
~,X,y,v
3350 DATA q,X,X,X,X,X,X,X,X,X,X,X,X,X,
X,X,X,q
3360 DATA q,X,~,X,~,X,~,X,~,X,~,X,~,
X,~,X,q
3370 DATA q,X,X,X,X,X,X,X,X,X,X,X,X,X,
X,X,X,q
3380 DATA y,p,p,p,p,v,X,~,X,~,X,~,X,w,p,
p,p,p,x
3390 DATA " ", " ", " ", " ", " ", " ", "q,X,X,X,X,
X,X,X,q," ", " ", " ", " ", " ", " "
3400 DATA s,p,p,p,p,x,X,w,p,p,p,v,X,y,p,
p,p,p,r
3410 DATA X,X,X,X,X,X,X,q," ", " ", " ", "q,
X,X,X,X,X,X,X
3420 DATA s,p,p,p,p,p,p,x," ", " ", " ", "y,
p,p,p,p,p,p,r
3430 REM *** Spielablauf ***
3440 FOR i=3 TO 9-f*2 STEP 2:LOCATE I,23
:PEN 1:PRINT"h":NEXT
3450 INK 6,15:PEN 6
3460 uw=REMAIN(3):FOR I=1 TO APZ
3470 SFA=INT(RND(1)*17)+4:SFB=INT(RND(1)
*13)+4:IF M(SFB,SFA)=88 THEN LOCATE SFB,
SFA:PRINT CHR$(143);:M(SFB,SFA)=143 ELSE
3470
3480 NEXT i
3490 SOUND 2,0,0,0,4,0,1:WHILE SQ(2)>127
:WEND:SOUND 2,119,0,0,5,5:WHILE SQ(2)>12
7:WEND
3500 ptime=0
3510 FPU=FPU+APZ*9
3520 GOSUB 4560
3530 SIG=1
3540 AN=18:A=18:BN=10:B=10
3550 GOTO 3770
3560 FOR i=1 TO ag-r*2:NEXT i:X=0:Y=0:JO
=JOY(0)
3570 IF JO=1 OR INKEY(0)>-1 THEN Y=-1:GO
TO 3610
3580 IF JO=2 OR INKEY(2)>-1 THEN Y=1:GOT
O 3610
3590 IF JO=4 OR JO=5 OR JO=6 OR INKEY(8)
>-1 THEN X=-1:GOTO 3610
3600 IF JO=8 OR JO=9 OR JO=10 OR INKEY(1
)>-1 THEN X=1
3610 BN=B+X
3620 AN=A+Y
3630 IF BN<1 THEN BN=19
3640 IF BN>19 THEN BN=1
3650 IF AN<>ZN OR BN<>SN THEN 3770
3660 IF SIG<0 THEN 3740
3670 PEN 7:F=F+1:LOCATE B,A:PRINT" ";:LO
CATE BN,AN:PRINT CHR$(128);

```

```

3680 GOSUB 5110
3690 IF F>4 THEN 4410
3700 PEN 1:LOCATE 1,23:PRINT"
      ":FOR I=3 TO 9-F*2 STEP 2:LOCATE
      I,23:PRINT"h":NEXT
3710 LOCATE BN,AN:IF M(BN,AN)=88 THEN PE
N 12 ELSE PEN 6
3720 PRINT CHR$(M(BN,AN));
3730 GOSUB 4560:GOTO 3540
3740 PEN 1:LOCATE B,A:PRINT" ";:LOCATE B
N,AN:PRINT"h";
3750 GOSUB 5130
3760 pun=PUN+R*50:PEN 6:LOCATE 4,25:PRIN
T USING"##### PUNKTE";PU*r+pun:GOSUB 4
560
3770 IF M(BN,AN)=88 THEN SOUND 130,0,0,5
,3,0,25:LOCATE b,a:PRINT" ":LOCATE bn,an
:PEN 1:PRINT"i":PU=PU+1:PEN 6:LOCATE 4,2
5:PRINT USING"##### PUNKTE";PU*r+pun:M
(BN,AN)=32:GOTO 3830
3780 IF M(BN,AN)<>143 THEN 3830
3790 LOCATE b,a:PRINT" ";:LOCATE bn,an:P
EN 1:PRINT"i";:SOUND 2,239,0,0,5,2:PU=PU
+10:PEN 6:LOCATE 4,25:PRINT USING"#####
# PUNKTE";PU*r+pun:M(BN,AN)=32
3800 INK 11,11,17:SIG=-1:SR=0
3810 ZR=SGN(AN-Z)*(-1):IF ZR=0 THEN SR=S
GN(BN-S)*SIG
3820 IF TIME<ptime THEN ptime=ptime+1800
+300*(fa=0)ELSE ptime=TIME+1800+300*(fa=
0)
3830 IF X=0 AND Y=0 THEN 3880
3840 IF M(BN,AN)<>1 THEN 3870
3850 AN=A:BN=B:IF FA=0 AND sig>0 THEN IN
K 11,11:FOR I=1 TO 400:NEXT:INK 11,0
3860 GOTO 3880
3870 LOCATE B,A:PRINT" ";
3880 LOCATE BN,AN:PEN 1:PRINT"h";:A=AN:B
=BN
3890 IF SIG>0 THEN 3940
3900 IF TIME<ptime THEN 3940
3910 sig=1:INK 11,fa:SR=0
3920 ZR=SGN(an-z):IF ZR<>0 THEN 3940
3930 SR=SGN(bn-s)
3940 IF pun+pu*r<punb THEN 4010
3950 LOCATE 6,2:PRINT"! BONUS !"
3960 GOSUB 5150
3970 IF f>-4 THEN f=f-1 ELSE pun=pun+300
:LOCATE 4,25:PEN 6:PRINT USING"##### P
UNKTE";PU*r+pun
3980 punb=punb*2
3990 LOCATE 6,2:PRINT"      "
4000 PEN 1:LOCATE 3,23:PRINT"
      ":FOR i=3 TO 9-f*2 STEP 2:LOCATE
      i,23:PRINT"h":NEXT i
4010 IF pu>=fpu THEN pun=pun+(pu+10)*r:p
u=0:LOCATE 4,25:PEN 6:PRINT USING"#####
# PUNKTE";pun:GOTO 1240
4020 IF ZR<>0 THEN 4080
4030 ZRP=SGN(an-z)*sig:IF ZRP=0 THEN 413
0
4040 IF m(s,z+zrp)<>1 THEN 4070
4050 IF m(s+sr,z)<>1 THEN 4210

```

```

4060 ZR=-ZRP:SR=0:GOTO 4210
4070 ZR=ZRP:SR=0:GOTO 4210
4080 SRP=SGN(bn-s)*sig:IF SRP=0 THEN 417
0
4090 IF m(s+srp,z)<>1 THEN 4120
4100 IF m(s,z+zr)<>1 THEN 4210
4110 SR=-SRP:ZR=0:GOTO 4210
4120 SR=SRP:ZR=0:GOTO 4210
4130 IF m(s+sr,z)<>1 THEN 4210
4140 ZR=SGN(INT(2*RND(1))-0.5):SR=0
4150 IF m(s,z+zr)<>1 THEN 4210
4160 ZR=-ZR:GOTO 4210
4170 IF m(s,z+zr)<>1 THEN 4210
4180 SR=SGN(INT(2*RND(1))-0.5):ZR=0
4190 IF m(s+sr,z)<>1 THEN 4210
4200 SR=-SR
4210 ZN=Z+ZR:SN=S+SR
4220 IF SN<1 THEN SN=19
4230 IF SN>19 THEN SN=1
4240 IF M(S,Z)=88 THEN PEN 12 ELSE PEN 6
4250 LOCATE S,Z:PRINT CHR$(M(S,Z));
4260 LOCATE SN,ZN:PEN 13:PRINT" ";
4270 IF ZN<>AN OR SN<>BN THEN 4390
4280 IF SIG<0 THEN 4360
4290 F=F+1:LOCATE SN,ZN:PEN 7:PRINT CHR$
(12B);
4300 GOSUB 5110
4310 IF F>4 THEN 4410
4320 PEN 1:LOCATE 1,23:PRINT"
      ":FOR I=3 TO 9-F*2 STEP 2:LOCATE
      I,23:PRINT"h":NEXT
4330 LOCATE SN,ZN:PRINT" ";
4340 GOSUB 4560
4350 GOTO 3540
4360 LOCATE BN,AN:PEN 1:PRINT"h";:GOSUB
5130:pun=PUN+R*50:PEN 6:LOCATE 4,25:PRIN
T USING"##### PUNKTE";PU*r+pun
4370 GOSUB 4560
4380 GOTO 3560
4390 Z=ZN:S=SN:GOTO 3560
4400 REM *** Spielende ***
4410 PUN=PUN+PU*r:LOCATE 4,25:PEN 6:PRIN
T USING"##### PUNKTE";PUN
4420 INK 14,26,8:INK 15,8,26:SPEED INK 2
4,24:PEN 14:LOCATE 2,1:PRINT"***
      ***":PEN 15:LOCATE 6,1:PRINT"GAME OVE
R"
4430 ENV 1,3,5,1,1,0,10,2,-1,9,13,-1,20:
ENV 2,2,6,2,1,0,20,12,-1,25:EVERY 12,3 G
OSUB 330
4440 INK 9,16:PEN 9:LOCATE 1,23:PRINT"@
1985 MEDI-SOFTWARE"
4450 uw$=INKEY$:IF uw$<>""THEN 4450
4460 i$=INKEY$:IF i$="E"OR i$="e"THEN 46
00
4470 IF JOY(0)<>16 AND INKEY(9)<0 THEN 4
460
4480 PEN 1:FOR i=1 TO 17:LOCATE i,1:PRIN
T" h":FOR j=1 TO 150:NEXT j:SOUND 130,0,
0,5,3,0,25:LOCATE i,1:PRINT" i":FOR j=1
TO 50:NEXT j:NEXT i
4490 SYMBOL 104,&E0,&60,&6C,&76,&66,&66,
&E6

```


programme

```

4500 SYMBOL 105,&1B,&0,&3B,&1B,&1B,&1B,&
3C
4510 SYMBOL 114,&0,&0,&DC,&76,&60,&60,&F
0
4520 SYMBOL 115,&0,&0,&3C,&60,&3C,&6,&7C
4530 PUN=0:PU=0:R=0:LB=0
4540 GOTO 870
4550 REM *** Gespenst kommt heraus ***
4560 Z=10:ZN=10:S=10:SN=10:ZR=0:SR=SGN(I
NT(RND(1)*2)-0.5)
4570 PEN 13:LOCATE 10,12:PRINT"";
4580 LOCATE 10,12:PRINT" ";LOCATE 10,10
:PRINT"";:EVERY 12,3 GOSUB 330
4590 RETURN
4600 uw=REMAIN(3):SOUND 133,0:SYMBOL AFT
ER 32:FOR i=0 TO 79:KEY DEF i,1:NEXT:PAP
ER 0:PEN 1:MODE 2
4610 END
4620 REM *** DATAs fuer Sound ***
4630 DATA 2,7,2,12,1,12,1,14,2,16,2,12
4640 DATA 2,14,4,19,2,7
4650 DATA 2,11,1,11,1,12,2,14,2,11
4660 DATA 2,12,4,7,2,7
4670 DATA 2,12,1,12,1,14,2,16,2,12
4680 DATA 2,17,2,19,4,21
4690 DATA 4,19,4,11
4700 DATA 7,12,1,19
4710 DATA 2,24,2,23,3,21,1,19
4720 DATA 2,24,2,23,3,21,1,19
4730 DATA 2,24,2,23,2,21,2,19
4740 DATA 7,17,1,19
4750 DATA 2,23,2,19,3,21,1,17
4760 DATA 2,19,2,16,3,17,1,14
4770 DATA 2,16,2,19,2,16,2,14
4780 DATA 7,12,1,19
4790 DATA 2,24,2,23,3,21,1,19
4800 DATA 2,24,2,23,3,21,1,19
4810 DATA 2,24,2,23,2,21,2,19
4820 DATA 7,17,1,19
4830 DATA 2,23,2,19,3,21,1,17
4840 DATA 2,19,2,16,3,17,1,14
4850 DATA 2,16,2,19,2,16,2,14
4860 DATA 6,12,2,7
4870 DATA 2,12,1,12,1,14,2,16,2,12
4880 DATA 2,14,4,19,2,7
4890 DATA 2,11,1,11,1,12,2,14,2,11
4900 DATA 2,12,4,7,2,7
4910 DATA 2,12,1,12,1,14,2,16,2,12
4920 DATA 2,17,2,19,3,21,1,17
4930 DATA 3,19,1,16,3,17,1,14
4940 DATA 8,12
4950 DATA 3,12,1,14,1,16,1,14,3,12
4960 DATA 2,17,1,12,1,17,1,16,1,14,1,12
4970 DATA 6,19,1,14,1,16
4980 DATA 8,12
4990 DATA 3,12,1,14,1,16,1,14,3,12
5000 DATA 2,17,1,12,1,17,1,16,1,14,1,12
5010 DATA 6,19,1,14,1,16
5020 DATA 6,12
5030 DATA 2,1,4,2,4,1,2,3,1,1,1,2,2,1
5040 DATA 6,1,8,2,8,1,8,2,2,1
5050 DATA 2,1,4,2,4,1,2,3,1,1,1,2,2,1
5060 DATA 2,1,2,3,2,2,4,1,2,3,2,2,2,1

```

```

5070 DATA 379,319,426,358,478,379,426,63
B
5080 DATA 638,426,379,478,358,426,319,37
9
5090 DATA 478,379,319,239,190,239,253,21
3,253,284,239,284,319,253,319,284,239,28
4,253,213,253,239,478
5100 REM *** Fehlersound ***
5110 uw=REMAIN(3):SOUND 133,0:FOR so=1 T
O 8:SOUND 2,s1(so),0,0,6,6:NEXT so:WHILE
SQ(2)>127:WEND:RETURN
5120 REM *** Sound fuer gefangenes Gespe
nst ***
5130 uw=REMAIN(3):SOUND 133,0:FOR so=1 T
O 8:SOUND 2,s2(so),0,0,6,6:NEXT so:WHILE
SQ(2)>127:WEND:ptime=ptime+900:RETURN
5140 REM *** Bonusound ***
5150 uw=REMAIN(3):SOUND 133,0:FOR so=1 T
O 23:SOUND 130,s3(so),0,0,5,7:FOR j=1 TO
300:NEXT j,so:WHILE SQ(2)>127:WEND:EVER
Y 12,3 GOSUB 330:ptime=ptime-1500*(sig=-
1):RETURN

```

CHECK V1:PRUEFSUMMEN:

ZEILENNR.:	SUMMEN	1410- 1420:	53291	2850- 2860:	112208	4290- 4300:	51816
10- 20:	20174	1430- 1440:	20924	2870- 2880:	113130	4310- 4320:	17190
30- 40:	16412	1450- 1460:	62163	2890- 2900:	111874	4330- 4340:	11448
50- 60:	48741	1470- 1480:	62990	2910- 2920:	112221	4350- 4360:	11784
70- 80:	25809	1490- 1500:	128903	2930- 2940:	116647	4370- 4380:	1879
90- 100:	22084	1510- 1520:	122907	2950- 2960:	113420	4390- 4400:	42025
110- 120:	25462	1530- 1540:	115483	2970- 2980:	113508	4410- 4420:	63425
130- 140:	25026	1550- 1560:	128903	2990- 3000:	112922	4430- 4440:	82473
150- 160:	111439	1570- 1580:	64340	3010- 3020:	65466	4450- 4460:	73526
170- 180:	35240	1590- 1600:	63463	3030- 3040:	79180	4470- 4480:	27023
190- 200:	56066	1610- 1620:	64605	3050- 3060:	112637	4490- 4500:	59709
210- 220:	10018	1630- 1640:	21581	3070- 3080:	114353	4510- 4520:	42497
230- 240:	69122	1650- 1660:	126753	3090- 3100:	111034	4530- 4540:	38149
250- 260:	41217	1670- 1680:	128896	3110- 3120:	113989	4550- 4560:	96776
270- 280:	58947	1690- 1700:	64636	3130- 3140:	116953	4570- 4580:	40499
290- 300:	46148	1710- 1720:	63717	3150- 3160:	114936	4590- 4600:	42445
310- 320:	20655	1730- 1740:	1445	3170- 3180:	115487	4610- 4620:	24426
330- 340:	36964	1750- 1760:	127330	3190- 3200:	112637	4630- 4640:	28116
350- 360:	27684	1770- 1780:	64636	3210- 3220:	114859	4650- 4660:	21782
370- 380:	57480	1790- 1800:	123455	3230- 3240:	31188	4670- 4680:	23408
390- 400:	2403	1810- 1820:	58599	3250- 3260:	14513	4690- 4700:	6447
410- 420:	35354	1830- 1840:	78970	3270- 3280:	8124	4710- 4720:	22354
430- 440:	37160	1850- 1860:	113748	3290- 3300:	72697	4730- 4740:	14451
450- 460:	42933	1870- 1880:	112542	3310- 3320:	125227	4750- 4760:	23386
470- 480:	26347	1890- 1900:	112901	3330- 3340:	120451	4770- 4780:	14444
490- 500:	33128	1910- 1920:	112603	3350- 3360:	112252	4790- 4800:	22354
510- 520:	24682	1930- 1940:	116967	3370- 3380:	113880	4810- 4820:	14451
530- 540:	55797	1950- 1960:	114567	3390- 3400:	67511	4830- 4840:	22386
550- 560:	38082	1970- 1980:	114894	3410- 3420:	41066	4850- 4860:	13882
570- 580:	37834	1990- 2000:	59364	3430- 3440:	71045	4870- 4880:	22671
590- 600:	37954	2010- 2020:	113431	3450- 3460:	32453	4890- 4900:	21782
610- 620:	34582	2030- 2040:	79979	3470- 3480:	64883	4910- 4920:	27996
630- 640:	47758	2050- 2060:	112080	3490- 3500:	45229	4930- 4940:	12225
650- 660:	40801	2070- 2080:	66484	3510- 3520:	25443	4950- 4960:	40549
670- 680:	40945	2090- 2100:	2820	3530- 3540:	40548	4970- 4980:	7632
690- 700:	43817	2110- 2120:	67503	3550- 3560:	43483	4990- 5000:	40549
710- 720:	37869	2130- 2140:	6387	3570- 3580:	83552	5010- 5020:	7626
730- 740:	43433	2150- 2160:	3642	3590- 3600:	68311	5030- 5040:	31498
750- 760:	40753	2170- 2180:	2522	3610- 3620:	21625	5050- 5060:	40936
770- 780:	54244	2190- 2200:	67543	3630- 3640:	28181	5070- 5080:	57478
790- 800:	43943	2210- 2220:	80835	3650- 3660:	48205	5090- 5100:	40087
810- 820:	37479	2230- 2240:	74608	3670- 3680:	32914	5110- 5120:	16271
830- 840:	42090	2250- 2260:	112688	3690- 3700:	17190	5130- 5140:	72465
850- 860:	43942	2270- 2280:	107688	3710- 3720:	22567	5150- 5160:	10539
870- 880:	60838	2290- 2300:	110980	3730- 3740:	50454	GESAMTCHECKSUMME:	
890- 900:	31520	2310- 2320:	116584	3750- 3760:	10386	19868	
910- 920:	34609	2330- 2340:	116534	3770- 3780:	80916		
930- 940:	79688	2350- 2360:	68550	3790- 3800:	50048		
950- 960:	57803	2370- 2380:	1748	3810- 3820:	76256		
970- 980:	4374	2390- 2400:	70691	3830- 3840:	37701		
990- 1000:	33516	2410- 2420:	44477	3850- 3860:	42232		
1010- 1020:	59343	2430- 2440:	79810	3870- 3880:	72984		
1030- 1040:	68933	2450- 2460:	112166	3890- 3900:	22596		
1050- 1060:	38241	2470- 2480:	112376	3910- 3920:	79703		
1070- 1080:	66188	2490- 2500:	113046	3930- 3940:	68570		
1090- 1100:	79760	2510- 2520:	114059	3950- 3960:	8285		
1110- 1120:	28956	2530- 2540:	116387	3970- 3980:	43686		
1130- 1140:	48353	2550- 2560:	113704	3990- 4000:	35423		
1150- 1160:	26727	2570- 2580:	114640	4010- 4020:	63363		
1170- 1180:	96173	2590- 2600:	113972	4030- 4040:	61632		
1190- 1200:	88522	2610- 2620:	64002	4050- 4060:	66118		
1210- 1220:	42681	2630- 2640:	17474	4070- 4080:	66513		
1230- 1240:	37288	2650- 2660:	121961	4090- 4100:	73346		
1250- 1260:	64280	2670- 2680:	118868	4110- 4120:	57657		
1270- 1280:	41337	2690- 2700:	123066	4130- 4140:	87155		
1290- 1300:	5066	2710- 2720:	119149	4150- 4160:	51094		
1310- 1320:	5938	2730- 2740:	115693	4170- 4180:	88728		
1330- 1340:	5282	2750- 2760:	119980	4190- 4200:	43796		
1350- 1360:	5390	2770- 2780:	125250	4210- 4220:	61440		
1370- 1380:	51356	2790- 2800:	66695	4230- 4240:	47689		
1390- 1400:	92775	2810- 2820:	57157	4250- 4260:	49237		
		2830- 2840:	79992	4270- 4280:	42304		

Sind Aufwendungen für Home-Computer Werbungskosten?

Haben Sie schon mal darüber nachgedacht, ob Sie vom Finanzamt einen Teil des Kaufpreises Ihres Home- oder Personalcomputers zurückholen können? Für alle, die sich 1985 einen Homecomputer oder Personalcomputer zulegte, und für alle, die entsprechende Kaufabsichten haben, ist es wichtig zu wissen, unter welchen Voraussetzungen das Finanzamt derartige Aufwendungen als Werbungskosten anerkennt.

Die Finanzämter tun sich zur Zeit noch schwer mit der steuerlichen Behandlung der Aufwendungen für Heimcomputer als Werbungskosten. Eine gewisse Klarheit hat nun eine Verfügung der Oberfinanzdirektion Köln vom 26. 7. 1985 (§ 2354/27/ST 121) gebracht. Danach ist zunächst zu prüfen, wie der Computer beschaffen ist und welche Einsatzmöglichkeiten er eröffnet.

Handelt es sich um einen sogenannten Spiele-Computer, so spreche schon der objektive Charakter des Wirtschaftsgutes dafür, daß es nur privat verwendet wird. Aktentaschen- und Pocket-Computer dagegen seien aufgrund ihrer technischen Gestaltung regelmäßig für eine berufliche Verwendung gedacht und nicht geeignet für eine private Nutzung, insbesondere für Heim- und Videospiele.

Bei Computern, die wegen ihrer Beschaffenheit eine berufliche und eine private Nutzung zuließen (z. B. solche mit Cassette- oder Diskettenlaufwerken, Schreibmaschinentastatur, Bildschirm, Drucker), lasse der objektive Charakter des Wirtschaftsgutes nicht ohne weiteres auf den tatsächlichen Verwendungszweck schließen.

Es könne grundsätzlich davon ausgegangen werden, daß die Kosten für solche Computer, z. B. Commodore 64, Atari 600 XL, Dragon 32, Aufwendungen für ein gemischt genutztes Wirtschaftsgut seien.

Die Anschaffungskosten könnten deshalb bei diesen Computertypen regelmäßig nicht einfach und nach objektiven Maßstäben leicht nachprüfbar in einen beruflichen und privaten Teil aufgeteilt werden. Diese Kosten könnten vielmehr steuerlich nur dann berücksichtigt werden, wenn der Steuerpflichtige nachweise oder glaubhaft mache, daß er den Computer so gut wie ausschließlich beruflich nutze.

Um ein unvertretbares Eindringen in die

Privatsphäre des Steuerpflichtigen zu vermeiden, müßten die Kriterien einer so gut wie ausschließlich beruflichen Nutzung typisierend geprüft werden. Hierbei kämen insbesondere folgende Gesichtspunkte in Betracht:

– Arbeitet der Arbeitnehmer mit käuflich erworbenen berufsspezifischen Programmen, so spricht dies für die Anerkennung als Arbeitsmittel.

– Erstellt der Arbeitnehmer eigene Programme, so muß er hierfür über Grundkenntnisse verfügen. Will er mit Hilfe des Computers diese Grundkenntnisse erst gewinnen, spricht dies zunächst für die private Mitveranlassung. Wenn aber das Erwerben der Grundkenntnisse beruflich notwendig oder sinnvoll ist, z. B. um selbst berufsspezifische Programme zu entwickeln, können die Aufwendungen als Werbungskosten angesehen werden.

– Beruft der Arbeitnehmer sich darauf, daß in seinem Betrieb mit Computern gearbeitet wird und er sich deshalb zu Hause auf seine Berufstätigkeit vorbereiten müsse, so ist darauf abzustellen, ob der Arbeitnehmer zu Hause ein Gerät hat, dessen Ergebnisse im Betrieb technisch weiterverarbeitet werden können. Anderenfalls dürfte der Erwerb des Gerätes privat mitveranlaßt werden.

– Auch Arbeitnehmer, die ihre Geräte im Rahmen der allgemeinen Lebensführung erworben haben, verarbeiten mitunter nach dem Erwerb der Grundkenntnisse Probleme aus ihrem Berufsbereich. Hier können die Absetzungen für Abnutzung Werbungskosten sein, wenn der Computer jetzt so gut wie ausschließlich beruflich genutzt wird.

– Ein hoher finanzieller Aufwand (etwa ab 2.000 DM) kann für die fast ausschließliche berufliche Veranlassung sprechen.

– Das gleiche gilt, wenn der selbst angeschaffte Computer am Arbeitsplatz des Arbeitnehmers steht.

– Das Fehlen eines Druckers wird besonders bei Lehrern gegen eine so gut wie ausschließlich berufliche Nutzung sprechen, es sei denn, der Steuerpflichtige könne glaubhaft machen, daß er Ergebnisausdrucke nicht benötige oder auf andere Weise erhalte.

– Hat der Arbeitnehmer in Ausbildung befindliche Kinder, so kann widerlegbar eine private Mitbenutzung des Gerätes vermutet werden.

Die Oberfinanzdirektion Köln weist darauf hin, daß Aufwendungen für einen Computer nicht objektiv für den Beruf notwendig, üblich oder zweckmäßig seien. Es genüge regelmäßig, daß der Arbeitnehmer die Aufwendungen für die Berufstätigkeit für geeignet halten konnte.

Sei aber zweifelhaft, ob es sich bei Aufwendungen um Werbungskosten oder um nicht abziehbare Kosten der Lebensführung handele, so könne die Unüblichkeit, Nichtnotwendigkeit oder Unzweckmäßigkeit der Aufwendungen ein Indiz, gegebenenfalls sogar eine unwiderlegbare Vermutung für den privaten Charakter der Ausgabe sein.

Seien die Aufwendungen für die Anschaffung eines Computers ausnahmsweise beruflich veranlaßt, so sei regelmäßig von einer betriebsgewöhnlichen Nutzungsdauer von fünf Jahren auszugehen.

Auch die Oberfinanzdirektion Düsseldorf weist in ihrer Verfügung vom 14. 10. 1985 darauf hin, daß Aktentaschen-Computer unstrittig nur beruflich genutzt würden. Hier bestünden also keine Bedenken, den Anschaffungsaufwand als Werbungskosten abzusetzen.



Doppelveröffentlichungen

Als Abonnent der Zeitschrift „Compute mit“ muß ich mit Bedauern feststellen, daß der „Geist“ sehr schwach ist! Innerhalb der letzten Ausgaben wird kostbarer (von uns bezahlter) Platz mit Duplikationen voll gedruckt. Nur weiter so! Den Checksummer CPC kann man noch öfter bringen. **Joh. Mertig, Stephanskirchen**

Redaktion: Natürlich ist der „Geist“ in unserer Zeitschrift – so glauben wir jedenfalls – noch nicht schwach. Bisher haben wir so gut wie möglich versucht, Doppelveröffentlichungen auszuschließen. Die Checksummerprogramme für die einzelnen Systeme müssen leider mehrmals abgedruckt werden, da es ein neues Verfahren ist und auch neue Käufer dieser Zeitschrift nicht auf diesen Service verzichten sollen. Aber wie Sie sicherlich in dieser Ausgabe feststellen werden, wird der Schneider-Checksummer nicht mehr veröffentlicht. Allen neuen Lesern können sich jedoch jederzeit an den Tronic-Verlag wenden und dort die erforderlichen Unterlagen nachfordern.

Adventure- und Actionspiele für C-16/116

Ich besitze einen C-16. Bisher bin ich mit Ihrer Zeitschrift zufrieden. Aber könnten Sie nicht einmal ein Adventure oder ein Action-Spiel abdrucken? Außerdem möchte ich Sie bitten eine Liste verschiedener POKES abzudrucken. Dann möchte ich Sie fragen, ob das Listing „Helmut“ aus Heft 1 korrekt ist. Bei mir sind keine Zeitschwerter auf dem Bildschirm zu sehen. **Carsten Blunck, Oldenburg**

Redaktion: Innerhalb der Redaktion und unserer Programmierabteilung machen wir uns laufend Gedanken über die Programmangebote für die einzelnen Systeme. Unter der Rubrik Tips & Tricks werden daher auch ständig nützliche Informationen weitergegeben. Bei der Auswahl der Spiel- und Anwendungsprogramme muß man natürlich auch nach der Programmlänge entscheiden. Adventurespiele sind aufgrund Ihres Charakters meist jedoch sehr umfangreich und somit auch bei einem Abdruck in der „Compute mit“ äußerst platzraubend. Wir empfehlen Ihnen jedoch

darauf zu achten, ob in der von unserer Redaktion geplanten „Compute mit C-16/116 Sonderausgabe“ – die für die Osterzeit geplant ist – solche Programme veröffentlicht wurden. Das Listing „Helmut“ war korrekt. Bitte überprüfen Sie Ihr Programm nochmals sehr sorgfältig auf evtl. Eingabefehler.

Zeichenprobleme beim C-64 und VC-20

Wie bekomme ich dieses Zeichen „^“ auf den Bildschirm, welches im Grafikteil verschiedener Programme vorkommt? Nennen Sie mir bitte die entsprechenden Tasten. **Ralf Davids, Ruethen 3**

Albert Meyer, CH-Geroldswil

Redaktion: Das von Ihnen gesuchte Zeichen „^“ erhalten Sie mit der „Pfeil nach oben Taste“. Diese Taste enthält das Potenzierungszeichen. Sie finden Sie auf der Tastatur links neben der Restore-Taste.

Gibt es einen C-16 Checksummer?

In Ihrer Zeitschrift werden die Listings für den C-64 und Schneider-Computer mit „Checksummer“ aufgeführt. Ich möchte Sie fragen, ob dies nicht auch bei den Listings für den Commodore 16/116 möglich ist? **H.J. Gertlowski, Kerken**

Redaktion: Nachdem ab dieser Ausgabe auch ein Checksummerprogramm für den VC-20 veröffentlicht wird, fehlt nur noch der C-16/116 im Angebot dieses Eingabeservice. Unsere Programmierabteilung ist derzeit mit der Entwicklung eines solchen Programmes beschäftigt, so daß auch die Anwender des C-16/116 bald diesen Service in Anspruch nehmen können.

„Eigentor“

In Heft 1/86 schreiben Sie noch von fehlerfreien Programmen und in der gleichen Ausgabe haben Sie ein Programm mit einer fehlerhaften Zeile abgedruckt. Es ist das Programm „Mehr Spaß in den Hackerbuden“ für den Commodore 64. Dort heißt es in Zeile 140: `POKE 48980+i,asc (mid$(a$,i))`. Wenn diese

Änderung vorgenommen wurde, finde ich das Programm ganz gut. Ihre Zeitschrift ist spitze! **Michael Klanert, Dortmund**

Redaktion: Vielen Dank für den Hinweis auf das fehlerhafte Programm. Nach der Korrektur funktioniert es tatsächlich. Hier hat man wieder einen Beweis dafür, daß auch wir nicht frei von Fehlern sind und uns hin und wieder auch kleine Unzulänglichkeiten unterlaufen können. Wir bitten um Entschuldigung!

Mehr Programme für den C-16/116

Zum zweiten Male habe ich mir die „Compute mit“ gekauft, weil ich seit kurzem einen C-16 besitze und Ihre wenigen Seiten in Ihrem Magazin so ziemlich die einzigen sind, die es auf dem gesamten Zeitschriftenmarkt gibt. Wenn man bedenkt, daß die anderen Computer C-64, Schneider, Atari, MSX usw. Monat für Monat mehrere Magazine füllen, verstehe ich nicht ganz, warum auch Ihre Zeitschrift sich mit diesen Systemen beschäftigt. Warum ist es nicht möglich, einmal im Monat mehr Informationen für den C-16/116 zu bringen. Vielleicht starte ich mit diesem Brief eine interessante Leserumfrage, bin ich doch überzeugt, daß ich nicht allein mit meiner Meinung dastehe. **H.-J. Schwandt, Aachen**

Redaktion: Mit Ihrer Meinung stehen Sie sicherlich nicht allein. Die Überlegungen über dieses Problem sind in unserem Hause auch schon angestellt worden. Sicherlich ist ein Magazin für den C-16/116 für alle Besitzer dieses Systems recht interessant. Leider ist im internationalen Software-Bereich diesem System noch wenig Beachtung geschenkt worden, so daß kaum Neuerscheinungen auf dem Markt vertrieben werden. Auch der Hardware-Bereich ist bisher sträflich vernachlässigt worden. Wir haben uns jedoch entschlossen, eine Sonderausgabe für dieses System zu veröffentlichen, die sich auch bereits in der Vorbereitungsphase befindet. Dort werden Sie dann Infos, Programme, Tips & Tricks und vieles mehr über den C-16/116 nachlesen können. Im übrigen sind wir auf die Meinung der Leser zu diesem Thema wirklich sehr gespannt und bitten deshalb um entsprechende Zuschriften.

Bildschirmmasken in Sekundenschnelle erstellt mit

Screpri für den CPC 464

Wer hat sich nicht schon oft geärgert, wenn er eine Bildschirmmaske erstellen wollte und die PRINTs waren nicht richtig positioniert, der falsche Schreibstift verwendet oder eine Ink falsch gesetzt, da man die optimale Belegung erst im Zusammenspiel mit den anderen Bildschirmteilen erkennt? Da dauert es nicht selten fast eine halbe Stunde, bis eine Bildschirmmaske richtig steht.

Aber SCREPRI schafft hier Abhilfe! In Sekundenschnelle sind nun Bildschirmmasken für Adventures, Dateiverwaltungen, Programmierläuterungen etc. erstellt und das Ganze steht dann als Basicprogramm im Speicher des CPC 464.

Doch nun etwas konkreter. Bei SCREPRI handelt es sich um ein Maschinencodeprogramm, das es erlaubt, auf bequemste Art und Weise Bildschirmmasken zu erstellen. Sie müssen entweder das Assemblerlisting oder – was eine schwierige Aufgabe ist, wenn Sie keinen Assembler besitzen – den Basiclader eintippen und starten. Der Basiclader erzeugt das Binärfile SCREPRI.BIN automatisch, beim Assembler müssen Sie dieses mit 'SAVE"SCREPRI.BIN",B,&9900,&55F' von Hand kreieren. Wenn Sie nun SCREPRI benötigen, geben Sie MEMORY &98FF ein, laden SCREPRI.BIN und rufen es mit Call &9900 auf. Es belegt den Speicherplatz zwischen &9900 und &A570 und ist somit auch unter AMSDOS lauffähig. SCREPRI besitzt folgende Features: *Ink betrachten, Ink setzen, Mode setzen, Schreibstift setzen, Drucken eines beliebigen Charakters, Abbrechen, Delete, Cursorsteuerung, Ausgabe der aktuellen Cursorposition und Erstellen eines Basicprogrammes.* Die meisten Features werden mit Controllcodes, d.h. durch gleichzeitiges Drücken der CTRL und einer anderen Taste aufgerufen. Folgende Controllcodes sind belegt:

CTRL-P: Änderung des Schreibstiftes; nach CTRL-P muß eine Zahl zwischen 0 und 15 (beides inclusive) angegeben werden; Zahl größer als 15 bricht das Kommando ab;

CTRL-W:

Augenblickliche Cursorposition wird in physikalischen Koordinaten im unteren linken Eck des Bildschirms angezeigt. Erneuter Tastendruck stellt den alten Inhalt der überschriebenen Positionen wieder her. Diese Funktion ist vor allem zum Erstellen von Tabellen etc. geeignet, wo es nötig ist, z. B. bestimmte Abstände einzuhalten.

CTRL-B:

Bricht das Maschinenprogramm ab und kehrt zu Basic zurück. Um einen unbeabsichtigten Abbruch zu vermeiden, muß die Taste zweimal betätigt werden. Bei Eingabe einer anderen Taste nach CTRL-B wird das Kommando abgebrochen;

CTRL-C:

Ändern des Modes; gewünschten Mode eingeben; Zahl größer als 2 bricht das Kommando ab;

CTRL-E:

Legt die Bildschirmmaske als Basicprogramm im Speicher ab;

CTRL-I:

Diese Funktion ist in Ink setzen „S“ und Ink betrachten „G“ unterteilt. CTRL-I+„S“: (groß S!) = setzen einer Ink. Es müssen nacheinander die Inknummer, die 1. Farbe und die 2. Farbe eingegeben werden; CTRL-I+„G“ = Ink betrachten. Inknummer angeben; erste und zweite Farbe stehen im linken unteren Eck. Erneuter Tastendruck stellt den alten Zustand der Position wieder her;

CTRL-G:

Drucken eines beliebigen Charakters zwischen 0 und 255 (beides inclusive). ASCII-Code des Zeichens eingeben.

Die Steuerung des Cursor erfolgt mit den Cursorstasten. Das Kommando Delete (DEL-Taste) löscht das Zeichen links vom Cursor und zieht den Rest der Zeile um ein Zeichen nach links.

Die Zahleneingaben müssen in dezimaler Form erfolgen und mit Enter abgeschlossen werden.

Das sich, nach der Beendigung des Arbeitens mit SCREPRI, im Speicher befindliche Basicprogramm (das alte wurde gelöscht!) können Sie nun zu dem Programm, für welches Sie die Bildschirmmaske benötigen, mit 'MERGE'(Filename); hinzufügen (evtl. Renumerieren nicht vergessen!).

Sollte einmal der Cursor verschwunden sein, liegt das daran, daß Sie einen falschen Schreibstift gewählt haben. Wenn SCREPRI scheinbar nicht mehr auf Ihren Tastendruck reagiert, dann haben Sie unbeabsichtigt eine Funktion aufgerufen, welche nun auf die Eingabe einer Zahl wartet. In diesem Falle ist es am günstigsten 16 einzugeben.

Normalerweise bleibt SCREPRI im Speicher erhalten und Sie können es mehrmals hintereinander aufrufen, es sei denn, der HIMEM wird erhöht oder der Basicinterpreter wählt seinen 2K-Block für Kassettenoperationen aufgrund akuter Speicherplatznot oberhalb &9900.

programme

```
10 .....
.....
20 '      SCREPRI (Basic-Lader)
      Version 1.0
30 .....
.....
40 '      (c)1985 by      DUCKsoft
.....
50 '      Thomas Gmach
.....
60 '      Offersdorf 5
.....
70 '      8491 Rimbach
.....
80 .....
.....
100 MEMORY &9BFF : BORDER &0 : INK &0,&0
: INK &1,&F : MODE &1
110 FOR adr=&9900 TO &9E5F
120 READ byte: check=check+byte : POKE
adr,byte
130 NEXT adr
140 IF check<>172119 THEN PRINT "DATA-er
ror!!" : END
150 LOCATE 17,1 : PRINT "SCREPRI"
160 LOCATE 1,3 : PRINT "Abspeicherung de
s Objektcodes" : PRINT
170 SAVE "screpri.bin",b,&9900,&55F
180 END
190 ' der Maschinencode
200 DATA &21,&60,&9E,&01,&00,&00,&36,&14
,&03,&23,&CB,&58,&28,&F8,&CD
210 DATA &4E,&BB,&CD,&6C,&BB,&CD,&81,&BB
,&CD,&17,&BC,&04,&78,&32,&56
220 DATA &A6,&21,&01,&01,&CD,&75,&BB,&CD
,&06,&BB,&21,&50,&9E,&06,&11
230 DATA &BE,&28,&F5,&23,&10,&FA,&21,&43
,&9E,&06,&0D,&BE,&28,&09,&23
240 DATA &10,&FA,&CD,&20,&9B,&C3,&25,&99
,&FE,&F1,&2B,&44,&FE,&F2,&28
250 DATA &50,&FE,&F3,&28,&5C,&FE,&7F,&28
,&69,&FE,&10,&CA,&14,&9A,&FE
260 DATA &17,&CA,&2A,&9A,&FE,&09,&CA,&5D
,&9A,&FE,&02,&CA,&F4,&9A,&FE
270 DATA &05,&CA,&65,&9B,&FE,&03,&CA,&FD
,&9A,&FE,&0D,&CA,&0E,&9B,&FE
280 DATA &07,&CA,&44,&9B,&CD,&78,&BB,&7D
,&FE,&01,&CA,&25,&99,&2D,&CD
290 DATA &75,&BB,&C3,&25,&99,&CD,&78,&BB
,&7D,&FE,&19,&CA,&25,&99,&2C
300 DATA &CD,&75,&BB,&C3,&25,&99,&CD,&78
,&BB,&7C,&FE,&01,&CA,&25,&99
310 DATA &25,&CD,&75,&BB,&C3,&25,&99,&CD
,&78,&BB,&3A,&56,&A6,&BC,&CA
320 DATA &25,&99,&24,&CD,&75,&BB,&C3,&25
,&99,&CD,&78,&BB,&3E,&01,&BC
330 DATA &CA,&25,&99,&CD,&93,&BB,&F5,&E5
,&25,&E5,&CD,&75,&BB,&E1,&3A
340 DATA &56,&A6,&94,&47,&24,&C5,&E5,&CD
,&75,&BB,&E1,&E5,&CD,&16,&9D
350 DATA &7E,&FE,&14,&28,&03,&CD,&90,&BB
,&CD,&3A,&9D,&E1,&25,&E5,&F5
360 DATA &CD,&75,&BB,&F1,&CD,&20,&9B,&E1
```

```
,&C1,&24,&10,&DA,&E1,&E5,&7D
370 DATA &FE,&19,&28,&05,&3E,&20,&CD,&20
,&9B,&E1,&25,&CD,&75,&BB,&F1
380 DATA &CD,&90,&BB,&C3,&25,&99,&CD,&59
,&9D,&06,&10,&CD,&D2,&9D,&DA
390 DATA &25,&99,&FE,&00,&CA,&25,&99,&CD
,&90,&BB,&C3,&25,&99,&CD,&93
400 DATA &BB,&F5,&CD,&78,&BB,&E5,&CD,&BC
,&9C,&21,&19,&01,&CD,&75,&BB
410 DATA &E1,&E5,&7C,&CD,&D5,&9C,&3E,&2C
,&CD,&5A,&BB,&E1,&E5,&7D,&CD
420 DATA &D5,&9C,&CD,&06,&BB,&CD,&FA,&9C
,&E1,&CD,&75,&BB,&F1,&CD,&90
430 DATA &BB,&C3,&25,&99,&CD,&06,&BB,&47
,&FE,&47,&2B,&05,&FE,&53,&C2
440 DATA &25,&99,&CD,&93,&BB,&F5,&CD,&78
,&BB,&E5,&C5,&CD,&BC,&9C,&CD
450 DATA &59,&9D,&06,&0F,&CD,&D2,&9D,&C1
,&3B,&3E,&C5,&47,&F1,&FE,&47
460 DATA &28,&42,&C5,&CD,&59,&9D,&06,&1B
,&CD,&D2,&9D,&38,&F6,&47,&21
470 DATA &19,&01,&CD,&75,&BB,&C5,&78,&CD
,&D5,&9C,&3E,&2C,&CD,&5A,&BB
480 DATA &CD,&59,&9D,&06,&1B,&CD,&D2,&9D
,&3B,&F6,&C1,&4F,&C5,&CD,&D5
490 DATA &9C,&C1,&F1,&CD,&32,&BC,&CD,&06
,&BB,&CD,&FA,&9C,&E1,&CD,&75
500 DATA &BB,&F1,&CD,&90,&BB,&C3,&25,&99
,&78,&CD,&35,&BC,&C5,&21,&19
510 DATA &01,&CD,&75,&BB,&78,&CD,&D5,&9C
,&3E,&2C,&CD,&5A,&BB,&C1,&79
520 DATA &CD,&D5,&9C,&CD,&06,&BB,&CD,&FA
,&9C,&E1,&CD,&75,&BB,&F1,&CD
530 DATA &90,&BB,&C3,&25,&99,&CD,&06,&BB
,&FE,&02,&CB,&C3,&25,&99,&CD
540 DATA &59,&9D,&06,&02,&CD,&D2,&9D,&DA
,&25,&99,&CD,&0E,&BC,&C3,&00
550 DATA &99,&CD,&78,&BB,&3E,&19,&BD,&CA
,&25,&99,&26,&01,&2C,&CD,&75
560 DATA &BB,&C3,&25,&99,&F5,&CD,&78,&BB
,&3E,&19,&BD,&20,&06,&3A,&56
570 DATA &A6,&BC,&28,&0E,&CD,&16,&9D,&E5
,&CD,&93,&BB,&E1,&77,&F1,&CD
580 DATA &5A,&BB,&C9,&F1,&3E,&07,&CD,&5A
,&BB,&C9,&CD,&78,&BB,&3E,&19
590 DATA &BD,&20,&07,&3A,&56,&A6,&BC,&CA
,&25,&99,&CD,&16,&9D,&E5,&CD
600 DATA &93,&BB,&E1,&77,&CD,&59,&9D,&CD
,&5D,&BB,&C3,&25,&99,&DD,&2A
610 DATA &81,&AE,&DD,&23,&DD,&22,&5D,&A6
,&DD,&23,&DD,&23,&DD,&36,&00
620 DATA &0A,&DD,&23,&DD,&36,&00,&00,&DD
,&23,&01,&04,&10,&C5,&05,&DD
630 DATA &36,&00,&A2,&DD,&23,&0C,&DD,&36
,&00,&20,&DD,&23,&0C,&78,&CD
640 DATA &E0,&9D,&DD,&36,&00,&2C,&DD,&23
,&0C,&C5,&78,&CD,&35,&BC,&59
650 DATA &78,&C1,&CD,&E0,&9D,&DD,&36,&00
,&2C,&DD,&23,&0C,&78,&CD,&E0
660 DATA &9D,&DD,&36,&00,&01,&DD,&23,&0C
,&79,&C1,&4F,&10,&C3,&CD,&F8
670 DATA &9D,&0E,&00,&DD,&22,&5D,&A6,&DD
,&23,&DD,&23,&DD,&36,&00,&14
```

```

680 DATA &DD,&23,&DD,&36,&00,&00,&DD,&23
,&DD,&36,&00,&AD,&DD,&23,&DD
690 DATA &36,&00,&20,&DD,&23,&0E,&06,&CD
,&11,&BC,&CD,&E0,&9D,&DD,&36
700 DATA &00,&01,&DD,&23,&0C,&21,&60,&9E
,&7E,&FE,&14,&23,&2B,&FA,&32
710 DATA &5F,&A6,&DD,&36,&00,&BB,&DD,&23
,&DD,&36,&00,&20,&DD,&23,&CD
720 DATA &E0,&9D,&0C,&0C,&CD,&FB,&9D,&21
,&01,&01,&11,&2B,&00,&06,&19
730 DATA &C5,&3A,&56,&A6,&47,&CD,&0B,&9E
,&E5,&CD,&75,&BB,&C5,&CD,&16
740 DATA &9D,&C1,&7E,&FE,&14,&2B,&07,&3A
,&5F,&A6,&BE,&C2,&50,&9C,&D5
750 DATA &CD,&3A,&9D,&D1,&DD,&77,&00,&DD
,&23,&0C,&E1,&24,&10,&DC,&CD
760 DATA &34,&9E,&CD,&FB,&9D,&C3,&79,&9C
,&CD,&34,&9E,&DD,&36,&00,&01
770 DATA &DD,&23,&0C,&DD,&36,&00,&BB,&DD
,&23,&0C,&DD,&36,&00,&20,&DD
780 DATA &23,&0C,&7E,&32,&5F,&A6,&CD,&E0
,&9D,&CD,&FB,&9D,&CD,&0B,&9E
790 DATA &E1,&C3,&23,&9C,&2C,&26,&01,&C1
,&10,&9C,&DD,&36,&00,&00,&DD
800 DATA &23,&DD,&36,&00,&00,&DD,&23,&DD
,&36,&00,&00,&DD,&22,&5D,&A6
810 DATA &2A,&5D,&A6,&ED,&5B,&B1,&AE,&ED
,&52,&44,&4D,&CD,&00,&B9,&F5
820 DATA &CD,&2C,&F5,&F1,&CD,&0C,&B9,&C9
,&57,&06,&0A,&7B,&C6,&2F,&BA
830 DATA &2B,&04,&10,&FB,&37,&C9,&7B,&3D
,&37,&3F,&C9,&06,&06,&11,&57
840 DATA &A6,&2E,&19,&60,&E5,&C5,&D5,&CD
,&75,&BB,&CD,&3A,&9D,&D1,&12
850 DATA &13,&C1,&E1,&10,&EF,&C9,&57,&06
,&00,&04,&D6,&0A,&F2,&DB,&9C
860 DATA &7B,&C6,&2F,&CD,&5A,&BB,&FE,&2F
,&2B,&0F,&05,&97,&C6,&0A,&10
870 DATA &FC,&47,&7A,&90,&C6,&30,&CD,&5A
,&BB,&C9,&97,&1B,&FA,&11,&57
880 DATA &A6,&06,&06,&2E,&19,&60,&CD,&75
,&BB,&C5,&CD,&16,&9D,&C1,&7E
890 DATA &CD,&90,&BB,&1A,&13,&CD,&5A,&BB
,&10,&EA,&C9,&CD,&7B,&BB,&D5
900 DATA &E5,&45,&21,&00,&00,&05,&7B,&FE
,&00,&2B,&09,&3A,&56,&A6,&5F
910 DATA &16,&00,&19,&10,&FD,&D1,&5A,&1D
,&16,&00,&19,&11,&60,&9E,&19
920 DATA &D1,&C9,&C5,&CD,&93,&BB,&F5,&CD
,&16,&9D,&7E,&FE,&14,&2B,&0E
930 DATA &CD,&90,&BB,&CD,&60,&BB,&57,&F1
,&CD,&90,&BB,&7A,&C1,&C9,&3E
940 DATA &20,&1B,&F4,&CD,&06,&BB,&CD,&AA
,&9C,&3B,&FB,&32,&53,&A6,&CD
950 DATA &06,&BB,&57,&FE,&0D,&2B,&21,&7A
,&CD,&AA,&9C,&3B,&F2,&32,&54
960 DATA &A6,&CD,&06,&BB,&57,&FE,&0D,&2B
,&10,&7A,&CD,&AA,&9C,&3B,&F2
970 DATA &32,&55,&A6,&CD,&06,&BB,&FE,&0D
,&20,&F9,&3A,&53,&A6,&4F,&3A
980 DATA &54,&A6,&57,&3A,&55,&A6,&5F,&3E
,&14,&32,&53,&A6,&32,&54,&A6
990 DATA &32,&55,&A6,&7B,&FE,&14,&20,&04

```

```

,&5A,&51,&0E,&00,&7B,&FE,&14
1000 DATA &20,&02,&7A,&C9,&7A,&FE,&00,&2
B,&12,&7B,&42,&C6,&0A,&10,&FC
1010 DATA &67,&79,&FE,&00,&7C,&CB,&41,&C
6,&64,&10,&FC,&C9,&79,&FE,&00
1020 DATA &7B,&C8,&1B,&F3,&F5,&57,&7B,&9
2,&F2,&DC,&9D,&F1,&37,&C9,&F1
1030 DATA &37,&3F,&C9,&FE,&0A,&F2,&EA,&9
D,&C6,&0E,&C3,&F1,&9D,&DD,&36
1040 DATA &00,&19,&DD,&23,&0C,&DD,&77,&0
0,&DD,&23,&0C,&C9,&DD,&36,&00
1050 DATA &00,&DD,&23,&0C,&FD,&2A,&5D,&A
6,&FD,&71,&00,&FD,&36,&01,&00
1060 DATA &C9,&DD,&22,&5D,&A6,&DD,&23,&D
D,&23,&DD,&73,&00,&DD,&23,&DD
1070 DATA &72,&00,&DD,&23,&DD,&36,&00,&B
F,&DD,&23,&DD,&36,&00,&22,&DD
1080 DATA &23,&0E,&06,&E5,&21,&0A,&00,&1
9,&54,&5D,&E1,&C9,&DD,&36,&00
1090 DATA &22,&DD,&23,&0C,&DD,&36,&00,&3
B,&DD,&23,&0C,&C9,&0D,&7F,&FO
1100 DATA &F1,&F2,&F3,&10,&17,&09,&02,&0
3,&05,&07,&01,&04,&06,&0B,&0A
1110 DATA &0B,&0C,&0E,&0F,&15,&16,&1B,&1
9,&1A,&1C,&1F

```

CHECK V1: PRUEFSUMMEN:

ZEILENNR.: SUMMEN

```

10- 20: 29448
30- 40: 45067
50- 60: 61122
70- 80: 45374
100- 110: 40864
120- 130: 7762
140- 150: 71462
160- 170: 45862
180- 190: 20883
200- 210: 55402
220- 230: 52655
240- 250: 59824
260- 270: 66349
280- 290: 60907
300- 310: 60962
320- 330: 60462
340- 350: 60404
360- 370: 58383

```

```

380- 390: 59433
400- 410: 63008
420- 430: 60562
440- 450: 63480
460- 470: 60197
480- 490: 63498
500- 510: 59414
520- 530: 63148
540- 550: 57924
560- 570: 59092
580- 590: 62014
600- 610: 59343
620- 630: 53878
640- 650: 59195
660- 670: 56073
680- 690: 59619
700- 710: 55873
720- 730: 54700
740- 750: 58525
760- 770: 56222
780- 790: 59306

```

```

800- 810: 56833
820- 830: 55283
840- 850: 57911
860- 870: 55687
880- 890: 61650
900- 910: 51344
920- 930: 60928
940- 950: 58361
960- 970: 59249
980- 990: 52673
1000- 1010: 55999
1020- 1030: 60850
1040- 1050: 54444
1060- 1070: 57933
1080- 1090: 55751
1100- 1110: 72965

```

GESAMTCHECKSUMME: 60861

Tips & Tricks für den CPC 464

Grafische Bewegungen mit dem PRINT-Befehl

Haben Sie schon einmal mit Hilfe eines PRINT-Befehls versucht eine Graphik auf dem Schneider zu bewegen.

Sie meinen, das kann nicht funktionieren? Oh, doch! Herr Krizanac aus Burgdorf hat es mit Hilfe eines Tricks geschafft, dieses Problem zu verwirklichen. Tippen Sie das nachfolgende Listing ein und Sie werden den Effekt sehen.

Sind Sie nicht auch der Ansicht, daß der Autor mit diesem kleinen Trick einen ganz beachtlichen Erfolg erzielt hat?

Programm-Listing siehe nächste Seite →

werkstatt

```

100 *****
*****
110 *** Das ist die Graphik, die bewe
gt werden soll ***
120 *****
*****
130
140 MODE 2 :
' Modus
150 BORDER 0:INK 0,0:PEN 1 :
' Farben setzen
160 CLG :
' Anzeige loeschen
165 LOCATE 1,24 :
' Cursor links unten setzen
170 t%=RND*5+1 :
' Setze Pseudo Radom Integer Variable
180 k%=RND*5+1 :
' Setze Pseudo Radom Integer Variable

```

```

190 ORIGIN 320,200 :
' Graphik Nullpunkt festlegen
200 FOR a=0 TO 1000 STEP PI/30
210 x%=100*COS(a)
220 MOVE x%,x% :
' Verschieben des Graphikcursors
230 DRAW 200*COS(a/t%),200*SIN(a/k%),3:
' Zeichnet die Linie
240 PRINT" " : ' DAS IST DER TRICK !!!:
' Leerzeichen dadurch die Bewegung
250 NEXT
260 GOTO 160 :
' Zurueck nach 160

```

CHECK V1:PRUEFSUMMEN:

ZEILENNR.:	SUMMEN
100- 110:	121588
120- 130:	65416
140- 150:	83601
160- 165:	67438

170-	180:	8764
190-	200:	27288
210-	220:	57813
230-	240:	69385
250-	260:	64411

GESAMTCHECKSUMME: 41416

Werkstatt:

Alle Diskettenanwender finden heute ein kurzes Anwendungsprogramm unter der Rubrik „Werkstatt“, das hervorragende Eigenschaften in Punkto Verwaltung einer Schneider-Diskettenstation aufweist.

Disc-Help

für den Schneider CPC 464 und 664

Das Programm DISCHELP ist für eine komfortable Verwaltung einer Diskettenstation gedacht.

Es läßt sich z. B. als "\$.BAS" auf jede Diskette abspeichern.

Der Benutzer ist somit in der Lage, ohne CP/M einladen zu müssen, seine Disketten zu formatieren.

Weiterhin sorgt eine kleine Maschinensprach-Routine für den Ausdruck des Direktorys.

Dieses Programm ist als Anregung für Disketten-Arbeit gedacht und kann durch einige Modifikationen auch zu anderen Zwecken benutzt werden (z. B. durch kleine Veränderungen im Assemblerli-

sting kann ein HARDCOPY-Programm entstehen).

Die Steuerzeichen in den Zeilen 110 und 140 sind die Zeichen, die durch gleichzeitiges Drücken der CTRL und der X-Taste erreicht werden (CHR\$(24) Umtausch von Vorder- und Hintergrundfarbe).

DISC-HELP ist ein BASIC-Programm, das folgende Möglichkeiten für eine komfortable Diskettenverwaltung bietet:

1. Löschen eines Programmes auf der Diskette
Wird eine „1“ gedrückt, erscheint im oberen Bildschirmteil „Name :“. Geben Sie nun bitte den vollständigen Namen

des Programms ein, das gelöscht werden soll. Nach Betätigen der ENTER-Taste wird das Programm auf der Diskette gelöscht und ein neues Inhalts-

verzeichnis ausgegeben.

2. Umbenennen eines Programms auf der Diskette

Nach dem Drücken einer „2“ werden

Sie nach dem Namen gefragt, den das Programm erhalten soll.

Wurde die Eingabe mit ENTER bestätigt, werden Sie nach dem Namen des Programms gefragt, das Sie umbenennen wollen. Nach Betätigen der ENTER-Taste wird der Name des Programms auf der Diskette geändert und ein neues Inhaltsverzeichnis ausgegeben.

3. Tape-Disc Umschaltung

Bei Taste „3“ erscheinen die verschiedenen Möglichkeiten der Umschaltung. Durch Drücken der entsprechenden Tasten wird diese Umschaltung durchgeführt.

4. Änderung der USER-Ebene

Die USER-Nummer kann durch die Taste „4“ geändert werden. Es wird nach der neuen USER-Nummer gefragt, die durch die Eingabe der Zahlen 1 bis 15 beantwortet werden muß. Wurde „ENTER“ gedrückt, erscheint das Inhaltsverzeichnis der neuen USER-Ebene.

5. Ausgabe des Direktorys

Die Taste „5“ bewirkt die Ausgabe des Direktorys z. B. nach dem Wechseln der Diskette.

6. Umschaltung des Laufwerks

Wurde eine „6“ eingegeben, erscheint im oberen Bildschirmteil: „Laufwerk“.

Durch Eingabe von „A“ oder „B“ wird auf das entsprechende Diskettenlaufwerk umgeschaltet.

7. Formatieren einer Diskette

Wenn Sie eine „7“ gedrückt haben, sollte vorher eine Diskette in das Laufwerk eingelegt sein, da das Programm sofort mit dem Formatieren beginnt. Die Diskette wird nun, genau wie bei dem CP/M Kommando „FORMAT D“, in Spuren und Sektoren aufgeteilt.

8. Drucken des Direktorys

Nach Eingabe einer „8“ wird der im unteren Bildschirmfenster sichtbare Teil des Direktorys auf den Drucker ausgegeben, und die Schrift auf dem Monitor wird gelöscht.

```

10 *Dischelp 28.07.85 by J.-Holger Lange
20 FOR I=&9000 TO &9087:READ R:POKE I,R:
NEXT
30 DATA 62,2,205,180,187,205,156,187,33,
1,1,205,117,187,1,0,5,205,96,187,254,15
40 DATA 32,2,62,20,87,205,43,189,122,210
,27,144,197,62,32,205,93,187,193,11,121
50 DATA 254,0,32,226,120,254,0,32,221,20
5,156,187,62,0,205,180,187,201,0,0,0,62
60 DATA 0,14,193,30,0,87,245,33,100,144,
223,97,144,241,60,254,40,200,6,9,33,100
70 DATA 144,17,4,0,119,25,16,252,24,225,
82,198,7,0,0,193,2,0,0,195,2,0,0,197,2
80 DATA 0,0,199,2,0,0,201,2,0,0,194,2,0,
0,196,2,0,0,198,2,0,0,200,2
90 MODE 2:WINDOW #0,1,80,8,25:WINDOW #2,
1,80,1,7
100 PAPER 1:PEN 0:CLS:CAT:PEN 1:PAPER 0:
WINDOW SWAP 0,2
1 = Loeschen
:LOCATE 22,2:PRINT" 2 = Umbenennen "
120 LOCATE 42,2:PRINT" 3 = Tape / Disc
":LOCATE 62,2:PRINT" 4 = USER Nummer "
130 LOCATE 2,4:PRINT" 5 = Directory "
:LOCATE 22,4:PRINT" 6 = Laufwerk "
140 LOCATE 42,4:PRINT" 7 = Formatieren
"
150 ON BREAK GOSUB 420
160 i$=INKEY$:IF i$ <"1" OR i$>"8" THEN
160
170 i=ASC(i$)-&30:ON i GOTO 180,200,230,
330,340,350,390,400
180 LOCATE 2,6:INPUT "Name :",n$:WINDOW
SWAP 0,2:öERA,$n$
190 WINDOW SWAP 0,2:CLS:WINDOW SWAP 0,2:
GOTO 100
200 LOCATE 2,6:INPUT "Neuer Name :",nn$:
LOCATE 38,6:INPUT "Alter Name :",an$

```

```

210 WINDOW SWAP 0,2:LOCATE 2,1
220 öREN,$nn$,$an$:GOTO 190
230 LOCATE 2,6:PRINT"t = Tape d = Disc
i = Tape.in"
240 LOCATE 38,6:PRINT"o = Tape.out e =
Disc.in a = Disc.out"
250 i$=INKEY$:IF i$="" THEN 250
260 IF i$="t" OR i$="T" THEN öTAPE:CLS:G
OTO 110
270 IF i$="d" OR i$="D" THEN öDISC:CLS:G
OTO 110
280 IF i$="i" OR i$="I" THEN öTAPE.IN:CL
S:GOTO 110
290 IF i$="o" OR i$="O" THEN öTAPE.OUT:C
LS:GOTO 110
300 IF i$="e" OR i$="E" THEN öDISC.IN:CL
S:GOTO 110
310 IF i$="a" OR i$="A" THEN öDISC.OUT:C
LS:GOTO 110
320 GOTO 250
330 LOCATE 2,6:INPUT "Neue USER Nummer :
",n:öUSER,n:CLS
340 WINDOW SWAP 0,2:GOTO 100
350 LOCATE 2,6:INPUT "Laufwerk :",i$
360 IF i$="a" OR i$="A" THEN WINDOW SWAP
0,2:öA:GOTO 190
370 IF i$="b" OR i$="B" THEN WINDOW SWAP
0,2:öB:GOTO 190
380 GOTO 350
390 CALL &9040:RUN
400 LOCATE 2,6:INPUT "Disc-Name :",n$:PR
INT#8,,n$
410 CALL &9000:WINDOW SWAP 0,2:GOTO 190
420 MODE 2:END

```

CHECK VI:PRUEFSUMMEN:

ZEILENR. : SUMMEN	
10-	20: 111309
30-	40: 4763
50-	60: 1238
70-	80: 87487
90-	100: 43173
110-	120: 58669

130-	140: 45947
150-	160: 32221
170-	180: 86479
190-	200: 60026
210-	220: 26672
230-	240: 76460
250-	260: 79342
270-	280: 117394
290-	300: 63948

310-	320: 2028
330-	340: 16193
350-	360: 85508
370-	380: 64279
390-	400: 45363
410-	420: 12751

GESAMTCHECKSUMME: 29138

programme

Hier sind Spieler mit äußerst kurzen Reaktionszeiten gefragt! Nur mit Hilfe des rasenden Fahrstuhls können Sie Ihre Fabrik vor dem sicheren Ruin bewahren.

Die verrückte Fabrik *im Besitz der Schneider-User*

Noch zu Zeiten des deutschen Wirtschaftswunders haben Sie sich eine große Porzellanfabrik aufgebaut. Nachdem die Wirtschaftsrezession der sechziger Jahre mit Unternehmensgeist und Engagement überstanden wurde, entschlossen Sie sich zur Modernisierung Ihrer Produktionsstätten. Modernisierung bedeutet in diesem Fall, Roboter und Computer übernehmen den Produktionsablauf. Als die Modernisierung scheinbar die ersten Früchte trägt, passiert das Unfaßbare. Nach dem

Motto „Die Module spielen verrückt“, drehen die Computer und die von Ihnen gesteuerten Roboter durch. Erst wird die Fabrik halb zerstört und dann beginnen die Roboter, die wertvollen Meißener Porzellankisten in Richtung eines gähnenden Abgrundes zu schieben.

Es gibt nur einen Weg, das kostbare Porzellan, welches gleichbedeutend mit dem Betriebskapital ist, zu retten und damit die Fabrik vor dem sicheren Ruin zu bewahren. Mit dem Fabrikfahrstuhl müssen Sie

auf und ab rasen und immer an der Stelle ein Fließband aufbauen, wo eine Kiste in den Abgrund zu fallen droht. Es können immer nur gleichzeitig zwei Fließbänder aufgebaut werden. Doch aufgepaßt! – Die „Rettungsanker“ verschwinden nach kurzer Zeit wieder und sobald 10 Kisten im Abgrund verschwunden sind, ist Ihr einst so florierendes Unternehmen am Ende. Gesteuert wird per Joystick. Die Fließbänder werden durch Betätigen der Feuertaste aufgebaut.

```
1 REM *****
2 REM * Die verrueckte Fabrik *
3 REM *      Written by      *
4 REM *   Ottfried Schmidt   *
5 REM *****
6 REM
10 hsc=0
20 GOSUB 970
30 sc=0
40 DIM a(7),b(7)
50 kiste=0
60 DIM e(2):a#=CHR$(200):FOR i=1 TO 16:a
  #=a#+CHR$(201):NEXT i:a#=a#+CHR$(202)
70 DIM f(2)
80 MODE 1
90 INK 1,24:INK 2,0:INK 3,4:INK 0,6
100 BORDER 0:PAPER 2:PEN 3
110 CLS
120 FOR i=21 TO 24:FOR l=1 TO 40
130 LOCATE 1,i:PRINT CHR$(203)
140 NEXT l:NEXT i
150 FOR i=2 TO 21 STEP 3
160 FOR l=1 TO 19
170 PEN 1
180 LOCATE 1,i:PRINT CHR$(204)
190 NEXT l:LOCATE 1,i:PEN 3:PRINT CHR$(2
03):PEN 1:NEXT i
200 FOR i=1 TO 20
210 LOCATE 40,i:PEN 3:PRINT CHR$(203):NE
XT i
220 PEN 1
230 c=10:LOCATE 40,c:PRINT CHR$(208):LOC
ATE 40,c+1:PRINT CHR$(209)
240 FOR i=1 TO 7:a(i)=1:NEXT i
250 b(1)=1:b(2)=4:b(3)=7:b(4)=10:b(5)=13
:b(6)=16:b(7)=19
260 PEN 0
```

```
270 FOR i=1 TO 7
280 LOCATE a(i),b(i):PRINT CHR$(205);CHR
$(206);CHR$(207)
290 NEXT i
300 IF a(x)=18 THEN IF e(1)=b(x) OR e(2)
=b(x) THEN GOSUB 730
310 PEN 1:LOCATE 1,24
320 PRINT "kisten=";kiste:LOCATE 15,24:P
RINT "HIGH=";hsc:LOCATE 29,24:PRINT "SCO
RE=";sc
330 c1=c
340 IF JOY(0) AND 2 THEN IF c<19 THEN c=
c+3
350 IF JOY(0) AND 1 THEN IF c>2 THEN c=c
-3
360 LOCATE 40,c1:PEN 3:PRINT CHR$(203):L
OCATE 40,c1+1:PRINT CHR$(203)
370 PEN 1:LOCATE 40,c:PRINT CHR$(208):LO
CATE 40,c+1:PRINT CHR$(209)
380 x=INT(RND(1)*7)+1
390 LOCATE a(x),b(x):PEN 0:PRINT " ";CHR
$(205);CHR$(206);CHR$(207)
400 a(x)=a(x)+1:IF a(x)=19 THEN IF e(1)<
>b(x) AND e(2)<>b(x) THEN LOCATE a(x)+2,
b(x):PRINT " ":GOSUB 470
410 IF a(x)=19 THEN IF e(1)=b(x) OR e(2)
=b(x) THEN GOSUB 730
420 IF JOY(0) AND 16 THEN GOSUB 630
430 IF f(1)>0 THEN f(1)=f(1)-1:IF f(1)<=
0 THEN LOCATE 21,e(1)+1:PRINT "
":e(1)=0
440 IF f(2)>0 THEN f(2)=f(2)-1:IF f(2)<=
0 THEN LOCATE 21,e(2)+1:PRINT "
":e(2)=0
450 SOUND 1,0,4,5,0,0,x
460 GOTO 330
```

```

470 a1=a(x)+3:b1=b(x)
480 a2=a1:b2=b1
490 b1=b1+1:IF b1=21 OR b1=e(1) OR b1=e(
2) THEN GOTO 530
500 LOCATE a2,b2:PRINT " ":PEN 1:LOCATE
a1,b1:PRINT CHR$(207)
510 SOUND 1,80+(b1*3),5,14,0,0,1
520 GOTO 480
530 LOCATE a1,b1-1:PRINT CHR$(238)
540 SOUND 1,0,12,15,1,1,8
550 LOCATE a1,b1-1:PRINT " "
560 PEN 0
570 LOCATE a(x),b(x):PRINT " "
580 a(x)=1
590 LOCATE a(x),b(x):PRINT CHR$(205);CHR
$(206);CHR$(207)
600 kiste = kiste+1:LOCATE 8,24:PRINT ki
ste
610 IF kiste>10 THEN GOTO 840

```

Druckertische Staubschutzhauben

```

620 f(1)=30:PEN 1:LOCATE 21,c+1:PRINT a#
:PEN 0:e(1)=c
690 RETURN
700 IF e(2)<>0 THEN RETURN
710 f(2)=30:PEN 1:LOCATE 21,c+1:PRINT a#
:PEN 0:e(2)=c
720 RETURN
730 FOR i=a(x) TO 38
740 LOCATE i,b(x):PRINT " ";CHR$(207)
750 SOUND 1,70-i,4,14:NEXT i
760 FOR k=1 TO 50:SOUND 1,300-(2*(k+k)),
1,5:NEXT k
770 LOCATE 39,b(x):PRINT " "
780 a(x)=1
790 sc=sc+10:LOCATE 35,24:PRINT sc
800 LOCATE a(x),b(x):PRINT CHR$(205);CHR
$(206);CHR$(207)
810 IF e(1)=b(x) THEN e(1)=0:f(1)=0:LOCA
TE 21,b(x)+1:PRINT " "
820 IF e(2)=b(x) THEN e(2)=0:f(2)=0:LOCA
TE 21,b(x)+1:PRINT " "
830 RETURN
840 MODE 0:INK 3,0:PAPER 3:CLS
850 LOCATE 6,12:PEN 1:INK 1,2,25
860 PRINT "GAME OVER"
870 LOCATE 4,20:PEN 2:INK 2,16
880 PRINT "SCORE=";sc
890 IF sc>hsc THEN hsc=sc:PRINT CHR$(7);
"Neuer HIGH-SCORE"
900 FOR i=10 TO 100 STEP 4:SOUND 1,i,5:S
OUND 1,i+10,5:SOUND 1,i,5:NEXT i
910 CLS:PRINT "NEUES SPIEL?"
920 PRINT "FEUER=JA sonst NEIN"
930 IF JOY(0)=0 THEN GOTO 930
940 IF JOY(0) AND 16 THEN kiste=0:sc=0:G
OTO 80
950 MODE 1:INK 1,1:PAPER 1:BORDER 1

```

```

960 INK 0,27:PEN 0:CALL &BB03:CLS:END
970 SYMBOL AFTER 199
980 SYMBOL 200,63,120,204,132,132,204,12
0,63
990 SYMBOL 201,255,32,80,32,4,10,4,255
1000 SYMBOL 202,252,30,51,33,33,51,30,25
2
1010 SYMBOL 203,255,24,24,26,66,129,255,
255
1020 SYMBOL 204,255,129,66,60,60,66,129,
255
1030 SYMBOL 205,128,159,124,61,255,255,2
55,64
1040 SYMBOL 206,0,128,129,130,230,250,19
8,129
1050 SYMBOL 207,0,126,102,74,82,102,126,
66
1060 SYMBOL 208,60,60,102,102,255,165,16
5,255
1070 SYMBOL 209,219,255,219,127,63,63,31
,28
1080 MODE 1:PAPER 0:INK 0,0:BORDER 0:PEN
1:INK 1,6:INK 2,24
1090 CLS:PRINT "D I E V E R R U E C K T
E F A B R I K":PRINT
1100 PEN 2:PRINT "In der Hypermodernen P
orzellan-Fabrik sind die Computer ausg
eflippt"
1110 PRINT "Erst haben sie die halbe Fab
rik zerstoeht und jetzt schieben 7 "
1120 PRINT "Robotbulldozer Kisten mit we
rtvollem Maisner Porzellan auf den Ab
grund zu"
1130 PRINT "Nur Du kannst das Porzellan
retten mit Deinem Fahrstuhl kannst
du auf und ab fahren und 2 Fliesbaender
aufbauen"
1140 PRINT "Die Fliesbaender bleiben nur
kurtze Zeitstehen"
1150 PRINT "Sind mehr als 10 Kisten runt
ergefallen so ist das Spiel beendet (da
nn ist Dein Chef pleite)"
1160 PRINT "Gesteuert wird mit dem Joyst
ick"
1170 PRINT:PRINT "V I E L S P A S S "
1180 PRINT "Joystick bewegen"
1190 IF JOY(0)=0 THEN GOTO 1190
1200 RETURN

```

CHECK VI:PRUEFSUMMEN:

ZEILENNR.:	SUMMEN	370-	380:	24575	810-	820:	8840	
1-	2:	47107	390-	400:	87976	830-	840:	6188
3-	4:	40930	410-	420:	43195	850-	860:	12612
5-	6:	16144	430-	440:	123447	870-	880:	15429
10-	20:	5223	450-	460:	8538	890-	900:	100330
30-	40:	10873	470-	480:	66889	910-	920:	30932
50-	60:	52644	490-	500:	15365	930-	940:	68320
70-	80:	2561	510-	520:	19123	950-	960:	18976
90-	100:	14547	530-	540:	30129	970-	980:	29913
110-	120:	28089	550-	560:	14482	990-	1000:	41600
130-	140:	22979	570-	580:	30571	1010-	1020:	57914
150-	160:	16381	590-	600:	84373	1030-	1040:	65142
170-	180:	17025	610-	620:	17398	1050-	1060:	54797
190-	200:	65727	630-	640:	69877	1070-	1080:	46729
210-	220:	30354	650-	660:	52818	1090-	1100:	54433
230-	240:	55629	670-	680:	25852	1110-	1120:	30571
250-	260:	28345	690-	700:	11196	1130-	1140:	72640
270-	280:	27914	710-	720:	15116	1150-	1160:	113516
290-	300:	31871	730-	740:	48849	1170-	1180:	31089
310-	320:	25094	750-	760:	27577	1190-	1200:	11627
330-	340:	63562	770-	780:	18627			
350-	360:	68581	790-	800:	59439			

GESAMTCHECKSUMME: 7749

NEU!

THE GAMES

Commodore 64

DM 29,50

Tennis



Der alte Magier will einen Zaubertrank brauen. Dazu benötigt er allerlei Zutaten.

Also ist er unterwegs, um Spinnen, Kraken, Pilze und andere Gegenstände einzusammeln. Doch Hexen, Fledermäuse und anderes Ungetier versuchen ihn daran zu hindern. Der Magier ist aber nicht hilflos!

Die Monster kann er mit Zaubersternen abwehren. Hat er eine Zutat gefunden, so muß er sie in den Topf werfen. Ein Spaß für die ganze Familie.

Das TENNIS-Fieber hat uns alle ergriffen. Nicht zuletzt durch die großartigen Erfolge von Boris Becker ist der „Weiße Sport“ zu einem Volkssport geworden. In unserem Game treten zwei Tennis-Cracks gegeneinander an; sie schenken sich nichts; um jeden Ball wird gekämpft. Das Spiel basiert auf den Grundregeln des internationalen Tennis'. Hart umkämpfte Tie-Breaks sind ebenso an der Tagesordnung wie ein zermürbendes KUR-Fünfsatz-Spiel.

Das Spiel besteht aus 10 Kisten im Schwierigkeitsgrad. Sie haben hierbei die Möglichkeit, die Geschwindigkeit und den Ballwinkel zu variieren. Ein Spiel führt zu einem glücklichen oder unglücklichen Ende.

Das Spiel wird per Joystick gesteuert. Die Fließbänder werden durch Betätigen der Feuertaste aufgebaut.

ZX-Spectrum

DM 29,50

Witchcraft



THE GAMES

- Software

★ SUPER! ★

Postfach · D-3444 Wehretal 1 · Tel. 05651/40643