

Computronic

software für Heimcomputer

Doppelausgabe
Jan./Februar 85 2/2. Jahrgang

DM 6,50
öS 55
s.Fr 6,50

C=64

VIC-20

ATARI

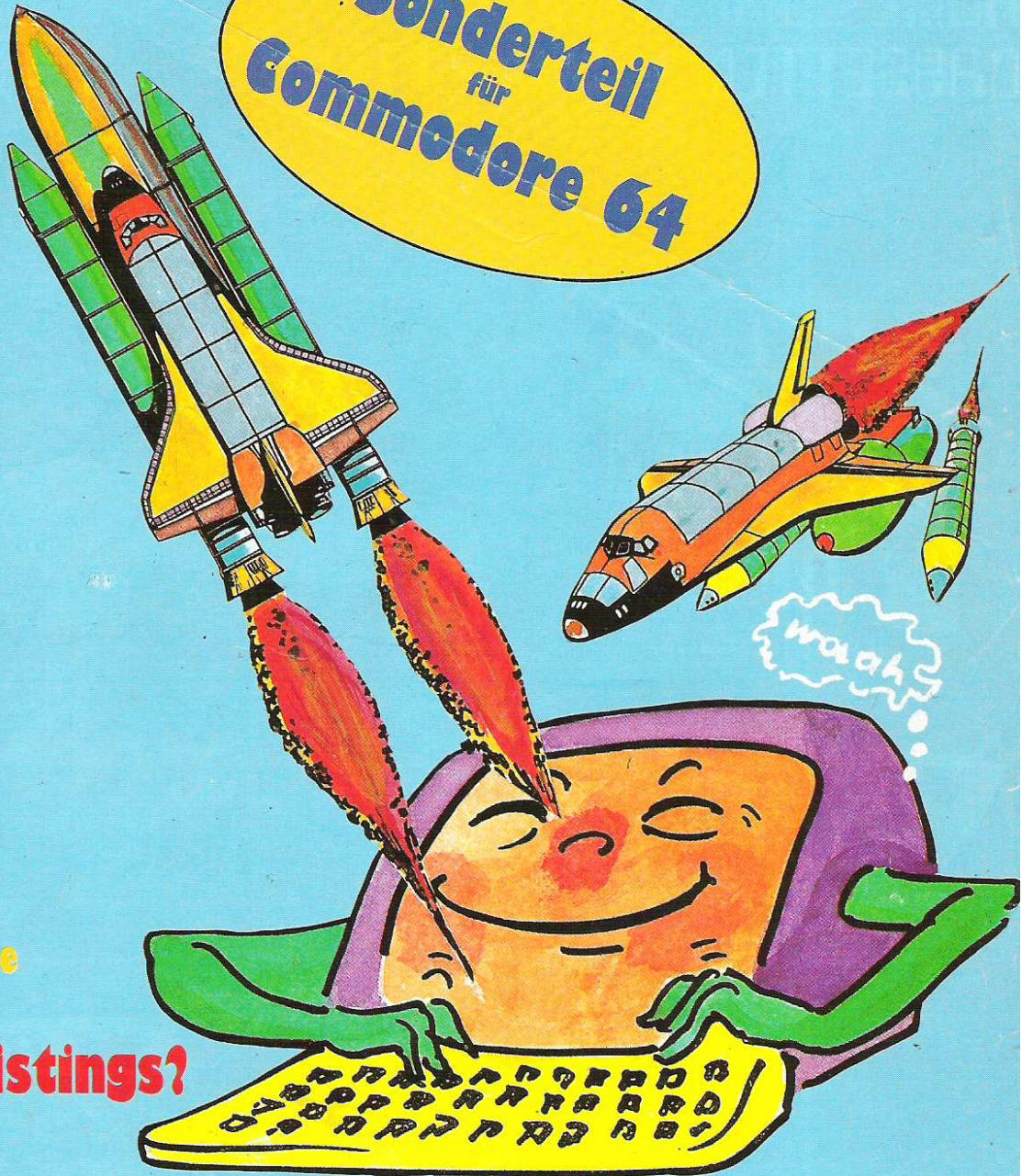
TI-99/4A

apple

sinclair ZX81

sinclair Spectrum

Sonderteil für Commodore 64



Interview

Heimlicher Handel mit Nato-Technologie

Fehlerhafte Listings?

Wir zahlen:

500,- Belohnung! Seite 86

12 tolle Spiele und Anwenderprogramme

magna

HOME-COMPUTER CASSETTEN + DISKETTEN

CASSETTEN

Präzise Cassetten-Mechanik
Hohe Speicherdichte
Für alle Data-Recorder

DATA-DISK DISKETTEN

Extreme Lebensdauer durch
zusätzliche Oxygenbeschichtung
Zuverlässige Datensicherheit
durch mehr als 70 chemische,
magnetische und elektrische
Qualitäts-Tests



TONTRÄGER

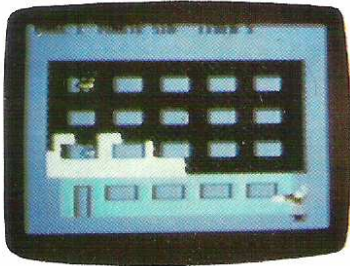
magna tonträger vertriebs gmbh

Bunzlauer Straße 3 · Postfach 400340 · 5000 Köln 40
Telefon (02234) 74054 · Telex 889975

INHALT

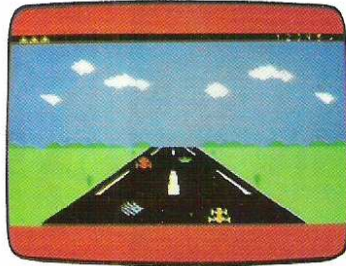
Heft Jan./Februar – Jahrgang 2 – 1985

Painter



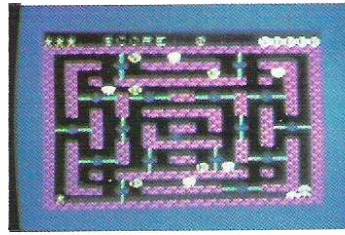
Commodore 64

Slicks



TI-99

Powerpack



VC-20

TOP

Star-Baddle



Commodore 64

Aktuelles

Software-Reviews	4
News	5
Interview mit Jay Tuck	4
Die Computerspione	
Büchermarkt	6
Commodore 64	7
Großer Sonderteil	

Service

Computer-Börse	50
Kleinanzeigen	
Software-Service	81
Kassetten und Diskettenkatalog	
Korrekturen	31
Haben Sie ein gutes Programm geschrieben	68

Commodore-Programme

Painter	8
Eine lustige Anstreicherei	
Star-Baddle	12
Krieg der Sterne	
Editor	20
Ein leistungsstarker Editor in Basic	

VC-20-Programme

Powerpack	26
Für VC-20, ohne Erweiterung	
Joy-Man	28
Spielversion, ohne Erweiterung	

TI-99-Programme

Frogger, Topprogramm	35
Moon-Race	38
Alarm auf der Mondbasis	
Slicks	43
Autorennen	

Atari-Programme

Super Miner	52
Für alle Atari-Computer	
Diamonds	57

ZX-81-Programme

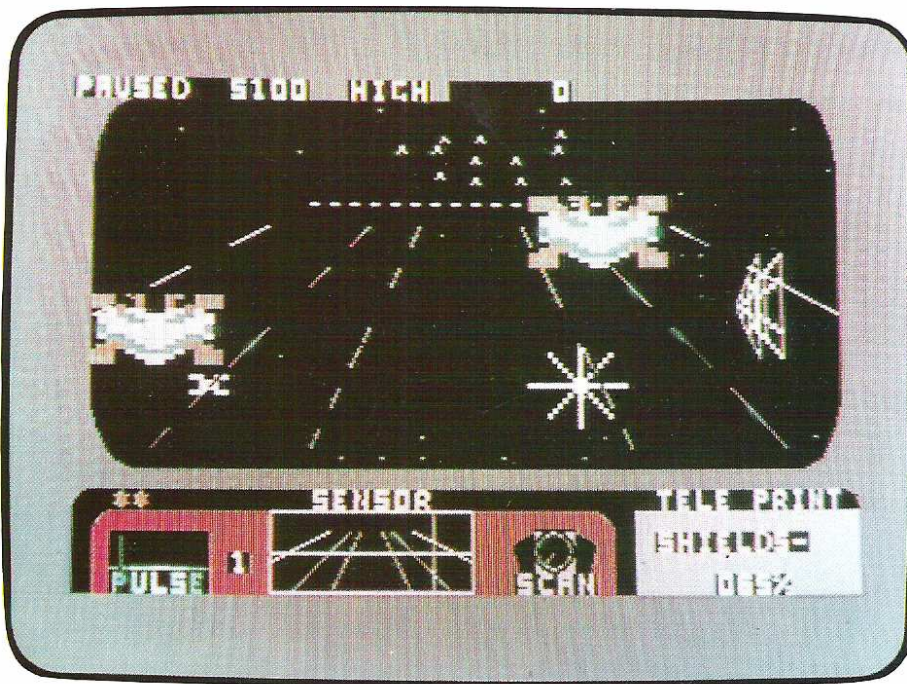
Spinnen	60
----------------------	----

Apple-Programme

Painter	66
Ein Zeichenprogramm	
Bowling	69
Spielversion	
10 Funktionstasten	73
Ein Anwenderprogramm*	

ZX-Spectrum-Programme

Jump	75
Ein schnelles Spiel	



Review

Software von Dynamics

für Commodore 64

„TRIAD“

In einem Raumschiff der Forschungsgruppe Erde sind Sie in ein feindliches Gebiet des Planeten TRIAD gekommen. Sofort werden Sie von den Triads und ihren Energiestrahlen angegriffen. Kämpfen Sie sich Ihren Weg durch die Gefahrenzone.

Interview mit Jay Tuck

Autor von „Die Computerspione – Der heimliche Handel mit NATO-Technologie“, Heyne Verlag, München

Frage: In Ihrem Buch schildern Sie mysteriöse Mitternachtstreffs auf dem Wiener Flughafen, die Entdeckung einer vereisten Leiche in den französischen Alpen und die ransante Fahrt von US-Fahndern durch den Hamburger Hafen. Auf weiten Strecken liest sich das wie ein Polit-Thriller. Warum der ungewöhnliche Stil?

Tuck: Die Geschichte der sowjetischen Computerspionage ist in der Tat ein Ost-West-Thriller. Er spielt sich in unserer Mitte ab und geht uns alle etwas an. Im Gegensatz zu einem Roman handelt es sich hier jedoch um die authentische Darstellung von Ereignissen im Leben wirklicher Menschen. Ich habe die Organisation von acht Schieberingen im Detail studiert und mir große Mühe gemacht, möglichst viel von der Dramaturgie und Farbe des Geschehens einzufangen – von den beteiligten Persönlichkeiten, über deren Ostkontakte, Schmuggelrouten und Schiebertricks, bis hin zu den Gentricks der Ermittlungsbehörden. Der illegale Handel mit strategischer Elektronik wird von westlichen Abwehrbehörden inzwischen als „Spionage-Thema des Jahrzehnts“ eingestuft.

Frage: Wer sind diese „Computerspione“?

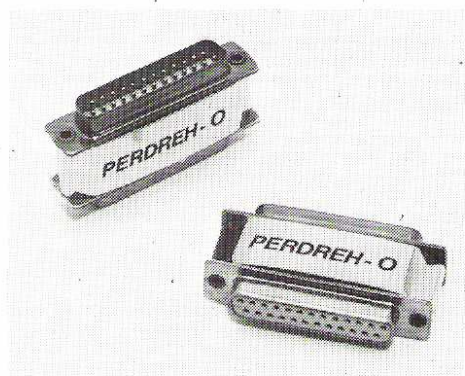
Tuck: In einem Fall handelt es sich um einen chinesischen Bankier, der von Hongkong in die US-Elektronikhochburg Silicon Valley reiste, um kalifornische Banken im Auftrag des Geheimdienstes KGB einzukaufen. In einem anderen Fall investierte ein deut-

scher Mehrfachmillionär in das Heimorgelgeschäft. Er suchte aber nur ein Tarnunternehmen für seine dubiosen Computergeschäfte und wurde am Ende von Agenten des Bundesnachrichtendienstes, des britischen MI-6 und der amerikanischen CIA rund um den Globus gejagt. Teilweise verfügen die westlichen Geschäftsmänner, die mit Embargoware handeln, über geheimdienstliche Kontakte. Häufig wurden sie unwissentlich in die Beschaffungspläne von Sowjetspionen eingespannt.

weiter Seite 32

Nullmodem zum Aufstecken

Unter der Bezeichnung PERDREH-0 bietet das Zevener Unternehmen



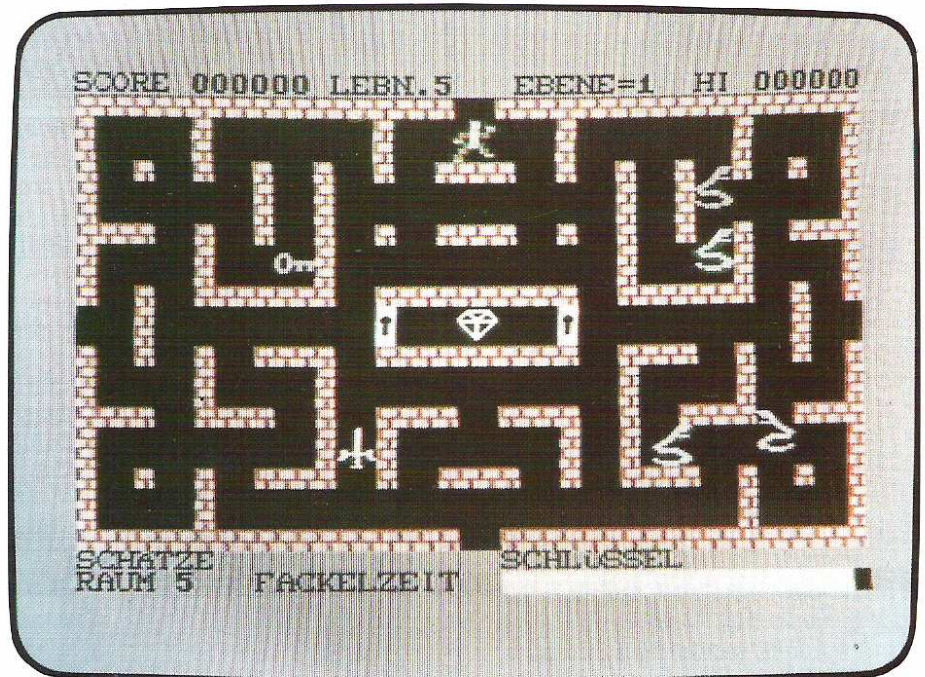
BAUZ GmbH ein Nullmodem in der Form eines Adapters an. Dieser Adapter kann einfach auf einen Kabel- oder Geräteanschluß aufgesteckt werden. Das Nullmodem dient dazu, gleichartige Geräte miteinander zu verbinden, ohne ein Spezialkabel herstellen zu müssen. Haupteinsatzgebiet für ein Nullmodem ist die Verbindung von Computern miteinander über eine V24/RS 232-Schnittstelle und einem normalen Druckerkabel.

Informationen:
BAUZ GmbH, Postfach 13 29,
2730 Zeven.

„BONGO“

Der Ritter Bongo hat im Wald ein geheimnisvolles Schloß entdeckt. In jedem Stockwerk dieses Schlosses sind 9 Räume, in jedem Raum ein Schatz versteckt. Um diesen Schatz zu besitzen, braucht er einen Schlüssel, der jedoch meistens in einem anderen Raum liegt.

Die Schätze werden von todbringenden Tieren bewacht, die es auf unsere Ritter abgesehen haben. Nur wenn er das Schwert erreicht, kann er sich ihrer erwehren.



The Factory

Kannst Du eine Fabrik alleine führen? Die Maschinen sind fertig und warten auf Dich!

So heißt es auf der Diskettenhülle des Programmes „THE FACTORY“. In der Tat, es geht darum, wie in einer richtigen Fabrik Produkte herzustellen. Dazu dienen drei zur Verfügung stehende Maschinen, und zwar:

PUNCH = Locher (stanzt aus einem Werkstück 1-3 quadratische oder runde Löcher aus)
STRIPE = Streifen (schleift dünne, mittlere oder dicke Streifen auf das Werkstück)

und

ROTATE = Rotieren (dreht ein Werkstück um 45, 90, 135 oder 180 Grad)

Nach dem Start des Programmes (und einem eventuellen Einstellen der Bildschirmfarben) hat man im wesentlichen drei Wahlmöglichkeiten:

(1) Test a Machine

Hierbei kann man sich jeweils eine der drei Maschinen aussuchen und in ihrer Wirkung vorführen lassen.

(2) Build a Factory

Wählt man diesen Programmpunkt, so kann man bis zu acht Maschinen zu einer Fließbandstraße zusammensetzen. Ist man damit fertig, so durchläuft ein Werkstück dieses „Fließband“ und

ein interessantes Spiel von Hes-Ware

wird entsprechend bearbeitet. Anschließend ist es möglich, daß ein Freund die Aufgabe bekommt, dieses Werkstück nachzumachen, d. h., er muß durch Kombinieren die richtigen Maschinen in der richtigen Reihenfolge zusammensetzen.

(3) Make a Produkt

Hat man keinen Freund zur Verfügung, kann man sich durch Anwählen dieses Programmpunktes auch vom Computer ein Werkstück vorgeben lassen, welches es dann nachzumachen gilt. Dabei kann man unter drei verschiedenen Schwierigkeitsgraden wählen.

Nach Angaben auf der Verpackung ist dieses Lernspiel für Kinder ab 7 Jahren geeignet. Und Kindern macht diese Art des Umganges mit dem Computer Spaß, wie ich bei meinen Nichten feststellen konnte. Zwar sehen die Maschinen etwas abenteuerlich aus, aber wenn sich dann ein Werkstück auf dem selbstentworfenen Fließband bewegt, dann kommt sogar Spannung auf. Ist das Teil richtig oder falsch geworden?!

Zur Bedienung des Programmes kann man nur sagen, daß sie dem Adressatenkreis (Kinder) angemessen ist. Das Programm ist menuorientiert und läßt sich mit drei Tasten (<, > und

RETURN) bedienen. Mit < und > steuert man ein Rechteck auf den Menüpunkt seiner Wahl und drückt die RETURN-Taste. Das ist alles.

Zusammenfassend möchte ich sagen, daß „THE FACTORY“ eines der schönsten und interessantesten Lernspiele für Kinder ist, welches ich für den VC-64 kenne. Gewünscht hätte ich mir nur, daß die Anweisungen des Programmes in Deutsch zu lesen gewesen wären, denn mir lag nur eine englische Version vor. Trotzdem hat das Programm allen meinen „Testpersonen“ gefallen, denn dieses Programm ist keineswegs nur als reines Kinderspiel zu betrachten. Auch für Erwachsene hat das Programm seinen Reiz. Probieren Sie es aus!

Programm: The Factory. Von: Hes-Ware. Datenträger: Diskette; Preis: 109,- DM.



Bücher-Report



Der Heim- oder Hobby-Computer ist besonders vielseitig. Man kann damit nicht nur spielen, sondern auch lernen oder nützliche kleine Anwendungen im häuslichen Bereich realisieren. In spielerischer Weise eignet man sich gleichsam nebenbei ein Wissen über Elektronik und Logik an, das in Zukunft im Berufsleben immer wichtiger werden wird. Der Laie, der sich einen Freizeit-Computer kaufen möchte, sieht sich allerdings vielen Fragen gegenüber, denn das Angebot ist riesen-groß.

Gilbert Obermaier hat mit seinem HEIMCOMPUTER REPORT '85 eine umfassende Marktübersicht geschaffen: Hardware, Software, genaue Beschreibungen, Preisangaben und exakte technische Daten, ein Heimcomputer-ABC, das alle Fachausdrücke erklärt, Anschriften von Vertriebsfirmen, Fachzeitschriften und Clubs. Heyne-Buch Nr. 08/4981, DM 6,80, Originalausgabe



Ist ein Heimcomputer im Haus, gibt es zwei Möglichkeiten. Entweder Vater und Sohn sehen sich nur noch zu den Mahlzeiten – weil der Sohn auf die neue Technik total abfährt und weil der Vater auf die neue Technik pfeift. Oder Vater und Sohn machen gemeinsame Sache.

Aber: der Sohn will spielen, programmieren, hackern.

Der Vater will den Computer sinnvoll anwenden, die neue Technik ausprobieren und einüben.

Der Mikrocomputer-Markt existiert – obwohl hier inzwischen Milliarden umgesetzt werden – ganze sieben Jahre. Und er wächst und wächst und wächst... Entsprechend schnell wandelt sich die Szene: Anbieter kommen und gehen, Adressen, Preise, technische Daten, geschäftliche Bedingungen ändern sich ständig. Dieses Jahrbuch ist ein Nachschlage- und Lesebuch für Einsteiger, Freaks und Profis auf dem neuesten Stand. Marktübersichten – Entscheidungshilfen – Informationen – Daten – Adressen. Einkauf – Anwendungen – Kontakte. Heyne-Buch Nr. 15/6, DM 14,80, Originalausgabe



Wie der Heimcomputer für Vater und Sohn gemeinsam Spaß und Nutzen, Abenteuer und Herausforderung bedeuten kann, zeigt dieses Buch von Tom (45) und Fabian (13) Werneck. Heyne-Buch Nr. 15/7, DM 9,80, Originalausgabe



„Der mächtige Draht“, herausgegeben von Key B. Hacker, ist ein Buch über die verschiedenen Aspekte moderner Kom-



munikationstechniken. Es ist das erste Buch, das einerseits die Möglichkeiten von Mailbox-Systemen beschreibt und sich andererseits kritisch mit der Datenfernübertragung allgemein auseinandersetzt. Die Mailbox selbst ist hierbei als Vehikel benutzt worden, das Buch zu konzipieren, zu erarbeiten und zu erstellen, denn die einzelnen Kapitel von sehr unterschiedlicher Länge und Thematik sind an das elektronische Postfach des Herausgebers adressiert und über Datex-p dem Fach entnommen worden. Sie wurden anschließend elektronisch gespeichert und mit einem Computer bis zur fertigen Druckvorlage weiterverarbeitet.

Der Herausgeber hat sich jedoch nur auf das Sammeln und Zusammenstellen der in seiner Mailbox eingehenden Artikel beschränkt. Vielmehr kommentiert und beschreibt er das rege Treiben in seinem elektronischen Postfach, das im Verlauf der Bucherstellung vorzufinden war. Dadurch werden Lebendigkeit und Zweck eines solchen „computer based message systems“ besonders transparent. Alle Beiträge stammen von einer „closed user group“, zu der auch Computerbenutzer gehören, die die Aufgabe hatten, das Mailbox-System auf Standfestigkeit mit allen Tricks und Mitteln auszutesten.

Von erheblicher Bedeutung werden Mailbox-Systeme für ausreichend große Organisationen sein, die einen hohen Bedarf an Informationsaustausch haben. Dazu gehören z. B. auch Vereine. So ist die A.U.G.E. (Apple User Group Europe) bereits Nutzer einer Mailbox. Diese wie auch verschie-

dene Firmen, nutzen jetzt schon Systeme, die auf dem deutschen Markt vertreten werden. Die Bundespost wird 1985 ihr Telebox-System anbieten. Key B. Hackers Buch ist deshalb so

wichtig, weil es dem Leser ein Gefühl vermittelt, was die Nutzung von Mailbox-Systemen, Datenbanken und Telekommunikationsdiensten für weite Kreise bedeuten wird und weil es ent-

sprechende Entscheidungen wesentlich erleichtern oder erst ermöglichen wird. Key B. Hacker (Hrsg.) Braunschweig: Vieweg 1984. Ca. 230 S. Gbd. ca. DM 45,-

Macintosh

Ein Computer und seine Mitwelt

Der Macintosh, der neue PC von Apple, wird als faszinierendes Computer-Phänomen der Zukunft gesehen. Jedoch befaßt sich der Autor nicht nur mit einem hochaktuellem Produkt der „32-bit-Welt“, sondern sieht sich gleichzeitig in der Welt der Computer um, damit auch der Leser seinen Blick schärfen kann.

Das Buch ist in einem für herkömmliche Computer-Bücher ungewöhnlichen Stil verfaßt und setzt sich kritisch mit Produkt und Hersteller in der neuen Welt der Computer auseinander.

Der Macintosh, Lisas Sohn und Maus-Besitzer, ist der erste Schritt zur Buch-

herstellung im Wohnzimmer; die Programme MacWrite und MacPaint ermöglichen dies.

So wurde dieses Buch vom Autor bis auf wenige verlagsbedingte Kleinigkeiten auf dem Macintosh hergestellt, einschließlich der in dem Buch enthaltenen Illustrationen und Bilder. Über dem Macintosh sagt das Buch nicht nur viel, sondern alles. Key B. Hacker Braunschweig: Vieweg 1984. 197 S., DIN C 5. Br. DM 48,-

Commodore 64 Sonderteil

Beste Software – schöne Spiele

Buchbesprechung:

6502-Assemblerkurs für Beginner

Der „6502-Assemblerkurs für Beginner“ von Andreas Dripke, dem „Vater“ von Exbasic Level II, stellt eine Einführung in die Assemblersprache des 6502-Mikroprozessors dar, wie er sich z. B. im Apple II/IIe, in CBM- und Atari-Computern und im VC-20 befindet. Auch Besitzer eines VC-64, zu welchem ich persönlich gehöre, sind mit dem Buch angesprochen, denn deren 6510-Mikroprozessor ist voll kompatibel zum 6502. Voller Erwartungen begann ich, das

Buch, welches nun (1 Jahr nach Erscheinen!) bereits in der 2. Auflage vorliegt, durcharbeiten, um endlich direkt in der schnellen Maschinensprache Programme schreiben zu können. Etwas skeptisch war ich schon, denn auch in anderen Büchern wird eine leicht verständliche Einführung in diese Materie versprochen, dieses Versprechen dann aber keineswegs eingelöst. Aber schon das Äußere des Buches
weiter Seite 32



Painter Eine lustige Anstreicherei!

PAINTER VORPROGRAMM

```

10 REM *****
11 REM *
12 REM *
13 REM * P A I N T E R (VORPROGRAMM)
14 REM * VON
15 REM * THOMAS GOESMANN
16 REM *
17 REM * 5778 MESCHEDE
18 REM *
19 REM *
20 REM *****
50 POKE198,3:POKE631,76:POKE632,111:POKE633,13
60 POKE43,1:POKE44,34:POKE34*256,0:NEW: REM BASIC
   START AUF 8704
  
```

Painter ist in Maschinensprache geschrieben und hat dadurch einen sehr schnellen Bewegungsablauf.

Eine Hauswand soll neu angestrichen werden. Für Farbe und Pinsel ist gesorgt. Versuchen Sie nun, die Hausfassade vollständig mit neuer Farbe zu bestreichen. Sie können Ihre Spielfigur nach rechts, links oben und unten steuern. Pro Spiel stehen Ihnen drei Spielfiguren zur Verfügung. Um das Spiel erfolgreich ausführen zu können, weichen Sie den etwas seltsam aussehenden Bewohnern des Hauses aus. Diese werden auf jeden Fall versuchen, die Anstreicherei vorzeitig zu beenden. Also – aufgepaßt und rechtzeitig aufhören zu arbeiten, sonst haben Sie das Spiel verloren.

Ein Tip: versuchen Sie nicht nur gegen die seltsamen Hausbewohner, sondern zusätzlich gegen die Uhr anzutreten. Wenn Sie es geschafft haben, die Fassade vollständig in eine neue Farbe zu kleiden, kommen Sie ins nächste Bild.

Das Anstreichen erfolgt über-Steuerung des Joysticks Port II oder über die Tastatur >W<, >A<, >S< und >Z<.

```

10 REM *****
11 REM *
12 REM * P A I N T E R (HAUPTPROGR.)
13 REM *
14 REM * VON
15 REM * THOMAS GOESMANN (C)'1984
16 REM *
17 REM *****
  
```

Hier einige Tips zum Laden und Abspeichern: Tippen Sie das Vorprogramm von Painter (Painter 1) ab! Speichern Sie es anschließend und starten Sie es. Nun tippen Sie das Hauptprogramm (Painter 2) ab und speichern es direkt hinter Painter 1 auf Kassette. Zum späteren Laden brauchen Sie nur SHIFT + RUN zu drücken!


```

18 REM
30 PRINT"III":V=53248:SI=54272:CS=58732:RU=1:LE=3:SC=0:POKEV+21,0
35 DIMP(15,2):DIMP(94):DIMP(16)
40 FORA=2040TO2046:READB:POKEA,B:NEXT:FORA=V+32TOV+45:READB:POKEA,B:NEXT
45 FORA=0TO16:READB:PO(A)=B:POKEA+V,B:NEXT
50 FORA=0TO15:FORB=0TO2:READA1:P(A,B)=A1:NEXT:NEXT
55 FORA=0TO94:READB:PA(A)=B:NEXT
60 POKESI+1,100:POKESI+5,0:POKESI+6,240:POKESI+24,15:POKESI+12,0:POKESI+13,240
65 POKE8181,3:POKE8182,0:POKE8184,1:POKE8183,0:POKE8185,0:POKE8187,0:POKE8188,0
70 POKEV+22,PEEK(V+22)OR16:POKEV+27,112
85 FORA1=0TO64*6-1:READB1:POKEA1+8192,B1:NEXT
88 REM
90 REM   ZEICHEN DEFINIEREN
95 REM
100 IFPEEK(2)=255THENPOKEV+24,19:GOTO200
102 PRINT"J":POKE214,9:SYSCS:PRINT"BITTE WARTEN SIE CIRCA 40 SEKUNDEN"
105 POKES6334,127:POKE1,51:FORA=0TO2047
110 POKEA+2048,PEEK(53248+A):NEXT:POKE1,55:POKES6334,129:POKEV+24,19:POKE2,255
120 FORA=4096TO4977:READB:POKEA,B:NEXT
130 READB1:ADR=B1*8+2048:FORA=0TO7:READB2:POKEA+ADR,B2:NEXT
140 PRINT"J":POKE211,0:POKE214,9:SYSCS:PRINT"JUEYSTICK (PORT II) ODER ATARSTATU
R"
145 POKE198,0:WAIT198,1:GETA$:IFA$="J"THEN170
150 IFA$<>"T"THEN145
160 FORA=4096 TO4130:READB:POKEA,B:NEXT
170 GOSUB 10000
185 REM
190 REM   SPIELFELD ZEICHNEN
195 REM
200 PRINT"J RUNDE"RU" PUNKTE"SC:PRINT"J"TAB(23)"LEBEN"LE:POKE8186,0:POKE8176,0
202 PRINT"JUN":
205 GOSUB250:GOSUB250:FORA1=0TO2:GOSUB260:GOSUB270:GOSUB250:GOSUB250:NEXT
210 FORA1=0TO4:PRINT"#####" " :NEXT
220 X=19:Y=11:GOSUB280:Y=17:GOSUB280:Y=23:GOSUB280:Y=29:GOSUB280
225 POKE211,7:POKE214,19:SYSCS:PRINT"##### "
226 POKE 53280,2
230 GOTO300
250 PRINT"#####":RETURN
260 PRINT"#####":RETURN
270 PRINT"#####":RETURN
280 POKE214,X:POKE211,Y:SYSCS:PRINT"#####":RETURN
300 FORA=0TO16:POKEA+V,PO(A):NEXT
302 POKEV+28,127:POKEV+21,127:POKE8189,18:POKE8190,20:POKE8186,0:PRINT"J"
305 TI$="000000"
310 POKEV+30,0:SY84096
315 IF PEEK(8186)=255 THEN1000
485 REM
490 REM   LEBEN VERLOREN
495 REM
500 FORA=PEEK(V+1)TO200STEP2:POKEV+1,A:POKEV+3,A+21:POKEV+5,A+36
505 POKESI+8,220-A:POKESI+11,33:NEXT:POKESI+11,0:POKE8182,0
510 FORA=0TO16:POKEA+V,PO(A):NEXT:POKE8189,18:POKE8190,20:POKE8186,0:POKEV+30,0
520 LE=LE-1:PRINT"J"TAB(28)LE
525 IFLE=0THENPOKE211,12:POKE214,12:SYSCS:PRINT"S P I E L E N D E":GOTO2000
530 FORA=0TO2000:NEXT:GOTO300
985 REM
990 REM   RUNDE BEENDET
995 REM
1000 SC=PEEK(8187)+256*PEEK(8188):BO=6000-TI:FORA=0TOBOSTEP20:SC=SC+20
1005 POKESI+11,0:POKESI+8,30:POKESI+11,33:PRINT"J"TAB(16):SC:NEXT:POKESI+8,0
1010 POKE8188,INT(SC/256):POKE8187,SC-INT(SC/256)*256:RU=RU+1:POKE8182,0
1015 IFPEEK(8181)>1THENPOKE8181,PEEK(8181)-1

```


Commodore 64

```

1020 FORA=0TO16:POKEA+V,PO(A):NEXT:POKE8189,18:POKE8190,20
1025 POKE4469,PEEK(4469)+3:GOTO200
1985 REM
1990 REM           SPIELEND
1995 REM
2000 FORA=0TO3000:NEXT
2010 POKE211,0:POKE214,24:SYS$C:PRINT"WOLLEN SIE NOCH EINMAL SPIELEN ? (J/N)";
2015 POKE198,0:WAIT198,1:GETA$:IFA$="J"THENPRINT"J":RUN
2020 STOP
9985 REM
9990 REM           SPIELANLEITUNG
9995 REM
10000 POKEV,32:POKEV+2,32:POKEV+4,32:POKEV+1,129:POKEV+3,150:POKEV+5,165
10005 POKEV+28,7:POKEV+21,7:X=3:Y=10:PRINT"J":POKE646,6
10010 FORA=0TO94:Z=PA(A):X1=P(Z,0):Y1=P(Z,1):P=P(Z,2):X=X+X1:Y=Y+Y1:X1=X*8+8:
10012 IFX1>255THENX1=X1-255:POKEV+16,7:GOTO10015
10014 POKEV+16,0
10015 Y1=Y*8+47:POKEV,X1:POKEV+2,X1:POKEV+4,X1:POKEV+1,Y1:POKEV+3,Y1+21
10020 POKEV+5,Y1+36:IFP=1THENPOKE211,X:POKE214,Y:SYS$C:PRINT"  ";
10050 NEXT:POKEV+21,0:A$="VON THOMAS GOESMANN":POKE214,12:POKE646,0
10060 FORA=1TOLEN(A$):FORB=30TOA+9STEP-1:POKE211,B:SYS$C:PRINTMID$(A$,A,1)" ";
10070 POKESI+11,0:POKESI+8,B+20:POKESI+11,17:NEXT:NEXT:POKESI+11,0
10080 PRINT:PRINT"DRUECKEN SIE EINE TASTE":POKE198,0:WAIT198,1
10100 PRINT"  "           S P I E L A N L E I T U N G
10110 PRINT"WBEI DIESEM SPIEL MUESSEN SIE VERSUCHEN, HAEUSERWAENDE ANZU";
10120 PRINT"STREICHEN, OHNE VON HERABFALLENDEN STEINEN ODER IM HAUS ";
10130 PRINT"HERUMIRRENDEN MONSTERN BERUEHRT ZU WER- DEN."
10140 PRINT"FUER JEDES STUECK, DAS SIE UMFAERBEN, ERHALTEN SIE 10 PUNKTE.";
10150 PRINT"AM ENDE JEDER RUNDE BEKOMMEN SIE EINEN BONUS, DER VON DER BE";
10160 PRINT"NDIGTEN ZEIT ABHAENGT."
10170 PRINT"WDIE STEUERUNG ERFOLGT WAHLWEISE UEBER: - JOYSTICK (PORT II)"
10180 PRINT"- TASTATUR (W,A,S,Z)":PRINT"DRUECKEN SIE EINE TASTE"
10190 PRINT"DRUECKEN SIE EINE TASTE"
10195 POKE198,0:WAIT198,1:RETURN
48985 REM
48990 REM   DATAS FUER INITIALISIERUNG
48995 REM
49000 DATA130,131,129,133,132,132,132
49100 DATA14,14,0,2,0,7,8,10,10,9,4,9,4,6
49200 DATA171,191,171,212,171,227,0,0,130,100,76,132,200,164,0,0,0
49485 REM
49490 REM   DATAS FUER SPIELANLEITUNG
49495 REM
49500 DATA0,-1,1,1,-1,1,1,0,1,1,1,1,0,1,1,-1,1,1,-1,0,1,-1,-1,1
49510 DATA0,-1,0,1,-1,0,1,0,0,1,1,0,0,1,0,-1,1,0,-1,0,0,-1,-1,0
49520 DATA0,,,,,2,2,2,3,4,4,5,6,6,10,10,10,9,2,2,3,4,4,4,6,6,6,,1,2,9,9,10,2
49530 DATA4,4,4,4,10,10,2,0,0,0,0,3,1,2,3,4,4,4,10,10,2,,,,,7,2,,,4,3,11,12,3
49540 DATA2,2,2,2,0,7,6,6,5,4,4,3,2,2,10,10,10,2,0,0,0,0,3,1
49985 REM
49990 REM           DATAS FUER SPRITES
49995 REM
50000 DATA48,0,12,63,255,252,63,255,252,15,235,240,,170,,170,,2,170,128,2,170
50010 DATA128,,,1,255,64,1,255,64,1,124,64,,124,,85,,85,,20,,20,,20,,,
50020 DATA,,,,,
50030 DATA48,0,12,63,255,252,63,255,252,15,235,240,,170,,170,,2,170,128,2,170
50040 DATA128,,,,,
50050 DATA,,,,,42,,106,,106,,64,,16,,16,,16,,20,16,,85,16,
50060 DATA85,12,,85,12,,85,60,,20,49,3,255,240,15,255,192,
50070 DATA12,255,,12,255,,12,255,,12,255,,5,40,,5,170,,170,,170,,130,,130,,
50080 DATA130,,130,,130,,130,,2,130,128,2,130,128,,,,,
50090 DATA63,15,192,63,15,192,2,170,128,10,170,160,41,170,104,42,170,168,170
50100 DATA165,90,165,85,90,170,170,170,42,170,168,2,0,128,2,128,160,,,,,
50220 DATA,,,,,

```


Commodore 64

```
50250 DATA,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,170,170,10,170,175,170,170
50260 DATA255,85,95,255,85,95,255,85,95,255,85,95,240,85,95,,
50985 REM
50990 REM      DATAS FUER M-CODE
50995 REM
50999 REM
51000 DATA169,224,141,2,220,173,0,220,170,41,1,208,3,76,73,16,138,41,2,208
51001 DATA3,76,137,16,138,41,4,208,3,76,201,16,138,41,8,208,3,76,9,17,32,86
51002 DATA17,173,250,31,201,0,240,11,169,0,141,11,212,169,255,141,2,220,96
51003 DATA32,75,17,32,209,17,32,234,18,76,5,16,173,1,208,201,63,208,8,169,0
51004 DATA141,4,212,76,40,16,169,129,141,4,212,169,8,141,255,31,206,253,31
51005 DATA206,1,208,206,3,208,206,5,208,32,70,17,32,209,17,206,255,31,208,236
51006 DATA169,0,141,4,212,169,129,141,250,7,76,40,16,0,0,0,173,1,208,201,199
51007 DATA208,3,76,40,16,169,8,141,255,31,238,253,31,238,1,208,238,3,208,238
51008 DATAS,208,32,75,17,32,209,17,206,255,31,208,236,76,40,16,0,0,0,0,0,0
51009 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,173,16,208,41,7,234,208,10,173
51010 DATA0,208,201,35,208,3,76,40,16,169,8,141,255,31,206,254,31,173,0,208
51011 DATA201,255,208,8,173,16,208,41,248,141,16,208,206,0,208,205,2,208,206
51012 DATA4,208,32,75,17,32,209,17,206,255,31,208,221,76,40,16,173,16,208,41
51013 DATA7,240,10,173,0,208,201,51,208,3,76,40,16,169,8,141,255,31,238,254
51014 DATA31,173,0,208,208,8,173,16,208,9,7,141,16,208,238,0,208,238,2,208
51015 DATA238,4,208,32,75,17,32,209,17,206,255,31,208,223,76,40,16,169,128
51016 DATA141,250,7,162,6,160,255,136,208,253,202,208,248,96,169,0,141,11,212
51017 DATA173,254,31,133,211,173,253,31,133,214,32,108,229,164,211,177,209
51018 DATA201,65,208,60,169,160,145,209,169,1,145,243,32,173,17,173,254,31
51019 DATA24,109,253,31,105,0,141,8,212,169,33,141,11,212,169,17,133,211,169
51020 DATA0,133,214,32,108,229,173,251,31,24,105,10,141,251,31,170,173,252
51021 DATA31,105,0,141,252,31,32,205,189,96,169,0,133,251,169,4,133,252,162
51022 DATA4,160,0,177,251,201,65,208,1,96,136,208,246,230,252,202,208,239,169
51023 DATA255,141,250,31,96,0,0,0,238,246,31,173,246,31,205,245,31,240,1,96
51024 DATA169,0,141,246,31,173,249,31,208,35,238,8,208,208,8,173,16,208,9,16
51025 DATA141,16,208,173,16,208,41,16,240,12,173,8,208,201,40,208,5,169,1,141
51026 DATA249,31,76,31,18,206,8,208,173,8,208,201,255,208,8,173,16,208,41,239
51027 DATA141,16,208,173,16,208,41,16,208,12,173,8,208,201,48,208,5,169,0,141
51028 DATA249,31,173,248,31,208,35,238,10,208,208,8,173,16,208,9,32,141,16
51029 DATA208,173,16,208,41,32,240,12,173,10,208,201,40,208,5,169,1,141,248
51030 DATA31,76,124,18,206,10,208,173,10,208,201,255,208,8,173,16,208,41,223
51031 DATA141,16,208,173,16,208,41,16,208,12,173,10,208,201,48,208,5,169,0
51032 DATA141,248,31,173,247,31,208,35,238,12,208,208,8,173,16,208,9,64,141
51033 DATA16,208,173,16,208,41,64,240,12,173,12,208,201,40,208,5,169,1,141
51034 DATA247,31,76,201,18,206,12,208,173,12,208,201,255,208,8,173,16,208,41
51035 DATA191,141,16,208,173,16,208,41,64,208,12,173,12,208,201,48,208,5,169
51036 DATA0,141,247,31,76,41,19,0,0,0,0,0,6,7,8,9,10,12,13,14,15,16,2,19,20
51037 DATA21,22,24,25,26,27,28,30,31,32,33,34,173,30,208,170,41,7,240,16,138
51038 DATA41,112,240,11,138,41,8,240,6,169,0,141,30,208,96,138,41,7,208,1,96
51039 DATA138,41,8,208,22,138,41,112,208,1,96,162,25,189,209,18,205,254,31
51040 DATA240,6,202,224,255,208,243,96,169,1,141,250,31,96,173,240,31,208,48
51041 DATA173,4,220,201,128,176,1,96,169,1,141,240,31,173,16,208,41,119,141
51042 DATA16,208,173,16,208,41,1,240,8,173,16,208,9,8,141,16,208,173,0,208
51043 DATA141,6,208,169,0,141,7,208,96,234,234,234,238,7,208,173,7,208,201
51044 DATA240,240,1,96,169,0,141,240,31,96
51985 REM
51990 REM      DATAS FUER ZEICHENSATZ
51995 REM
52000 DATA 65,255,186,186,186,255,171,171,171
52985 REM
52990 REM      DATAS FUER TASTATURSTEUERUNG
52995 REM
53000 DATA234,234,234,234,234,234,165,203,170,201,9,208,3,76,73,16,138,201
53005 DATA12,208,3,76,137,16,138,201,10,208,3,76,201,16,138,201,13
```

READY.

Commodore 64

Top programm

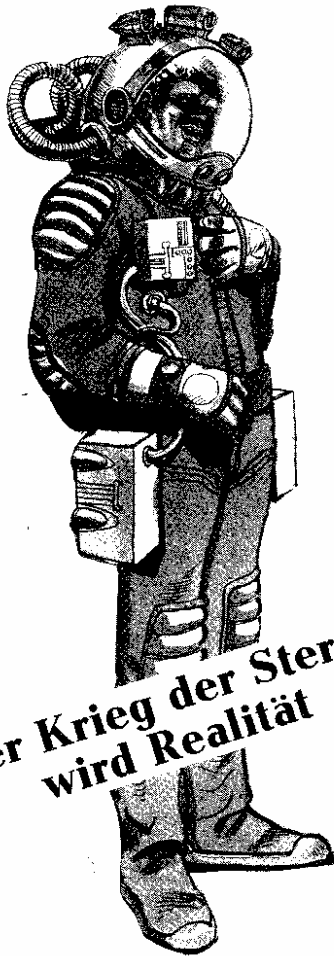
Böse Mächte sind in die Galaxis eingedrungen, um den Hauptplaneten Thor zu zerstören. Der Todesstern der Eindringlinge nähert sich unaufhörlich dem Planeten Thor.

Star Baddle

Verteidigen Sie Ihren Hauptplaneten. Zerstören Sie mit Ihren Marschflugkörpern die feindlichen Abfangjäger. Aber - nehmen Sie sich in acht vor den Geschützen der feindlichen Jäger, denn mit jedem Treffer, den Sie einstecken müssen, nähert sich der Todesstern dem Planeten Thor und zerstrahlt diesen nach 4 Treffern zu kosmischen Staub.

Nach dem Programmstart braucht der Rechner eine kurze Zeit, um die Grafik zu initialisieren. Dies geschieht jedoch nur vor dem ersten Spiel. Sobald ein Spiel beendet ist und ein neues Spiel gestartet wird, entfällt die Initialisierungsphase.

Sie sitzen in Ihrem Raumschiff und schauen durch das Cockpitfenster ins All. Sie rasen mit Ihrem Sternenjäger durch den Graben des Todes. Sobald das Spiel begonnen hat, taucht in der Mitte des Grabens hinter dem Todesstern ein feindlicher Jäger auf. Um diesen Jäger abzuschießen, drücken Sie den Aktionsknopf am Joystick und sofort setzt sich ein lenkbarer Marschflugkörper in Bewegung. Lenken Sie diesen Marschflugkörper zum Feind. Wenn Sie getroffen haben, so explodiert der Jäger. Denken Sie jedoch daran, daß nur ein exakter Treffer den



Der Krieg der Sterne
wird Realität

Jäger zerstört. Von Zeit zu Zeit schleudert Ihnen der feindliche Jäger eine Feuerkugel entgegen. Gleichzeitig mit der Feuerkugel erscheint in der Mitte des Grabens ein Zielkreuz, welches Sie mit dem Joystick steuern können. Führen Sie das Zielkreuz exakt über die Feuerkugel, und drücken Sie den Aktionsknopf, um die Kugel zu zerstören. Gelingt es Ihnen nicht, sie zu zerstören, bevor diese Ihren Jäger erreicht, so müssen Sie einen Treffer einstecken, und nach 4 Treffern heißt es: EXITUS für Thor. Nach diesen 4 Treffern sendet der Todesstern Laserstrahlen zum Planeten Thor aus und löst ihn zu kosmischen Staub auf. Danach können Sie durch Druck einer beliebigen Taste ein neues Spiel beginnen.

Während des Spieles kann durch die Taste F7 das Spiel angehalten werden. Durch Drücken des Joysticks wird das Spiel wieder fortgesetzt.

Um Star Battle möglichst interessant zu machen, steigt der Schwierigkeitsgrad bei jedem Treffer. Selbst erfahrene „Piloten“ werden kaum mehr als 5000 Punkte erreichen.

Am oberen Bildschirmrand ist die Anzeige eingelebnet. Die rechte Punktanzeige zeigt die aktuellen Punkte an. Die 4 gelben Kästchen in der Bildmitte geben die Anzahl der Schutzschilder an. Sobald Sie einen Treffer hinnehmen mußten, entfällt ein Schutzschild. Sind alle Kästchen erloschen, wird der Planet zerstört.

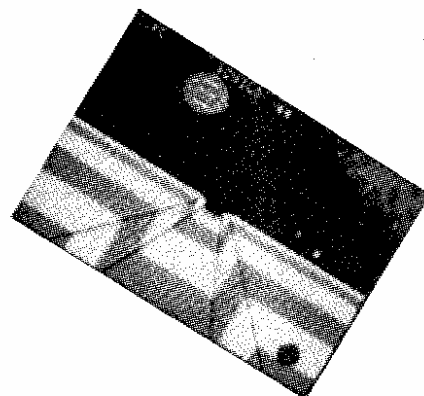
```
1 REM *****
2 REM *  TRONIC-SOFT  *
3 REM *---PRESENTS---*
4 REM *  TODESSTERN  *
5 REM *  C64  VERSION  *
6 REM * (C) BEI F.BRALL *
7 REM *****
8 POKE 56,128:POKE 648,4
9 POKE 53281,0:POKE 53280,2
10 POKE 54290,129:POKE54296,15
11 POKE 54292,255:POKE54287,10
14 PRINT"COMPTONIC-VERLAG  POSTFACH 41 3444 WEHRETAG";
```



```

15 PRINT"
20 PRINT"
30 PRINT"
40 PRINT"
50 PRINT"
60 PRINT"
70 PRINT"
80 PRINT"
90 PRINT"
91 PRINT"
100 PRINT"
110 PRINT"
120 PRINT"
130 PRINT"
140 PRINT"
150 PRINT"
160 PRINT"
170 PRINT"
175 FOR I =0 TO 16:POKE 53281,I:FORO=1 TO 200:NEXTO:NEXTI
190 FORI=1024 TO 1024+999:IF PEEK(I)=42 THEN POKE I,160
191 NEXTI
1000 U=0:FORI= 33000 TO 34897 :READ DA:POKEI,DA:U=U+DA:NEXTI
1010 IF U <> 241000 THEN PRINT"FEHLER IN DATA-ZEILEN":STOP
10000 DATA 234,32,52,133,120,169,103,141,20,3,169,132,141,21,3,88,32
10001 DATA 88,130,32,56,130,76,123,134,173,138,128,174,137,128,142,0,208
10002 DATA 160,1,162,254,76,138,129,173,140,128,174,139,128,142,2,208,160
10003 DATA 2,162,253,32,138,129,173,163,128,208,1,96,76,13,130,173,142
10004 DATA 128,174,141,128,142,4,208,160,4,162,251,76,138,129,173,144,128
10005 DATA 174,143,128,142,6,208,160,8,162,247,76,138,129,173,146,128,174
10006 DATA 145,128,142,8,208,160,16,162,239,76,138,129,173,148,128,174,147
10007 DATA 128,142,10,208,160,32,162,223,76,138,129,173,150,128,174,149,128
10008 DATA 142,12,208,160,64,162,191,76,138,129,173,152,128,174,151,128,142
10009 DATA 14,208,160,128,162,127,76,138,129,201,0,240,8,152,13,16,208
10010 DATA 141,16,208,96,138,45,16,208,141,16,208,96,173,30,208,41,192
10011 DATA 201,192,240,1,96,169,0,141,30,208,32,119,131,169,0,141,13
10012 DATA 208,141,163,128,169,8,141,249,207,32,35,130,238,249,207,32,35
10013 DATA 130,238,249,207,32,35,130,169,6,141,249,207,24,173,166,128,105
10014 DATA 150,141,166,128,173,167,128,105,0,141,167,128,162,0,160,6,24
10015 DATA 32,240,255,169,0,133,199,174,166,128,173,167,128,32,205,189,32
10016 DATA 56,130,173,27,212,41,3,208,8,169,1,141,162,128,32,65,134
10017 DATA 32,3,132,96,173,140,128,174,139,128,142,14,208,172,3,208,140
10018 DATA 15,208,160,128,162,127,76,138,129,72,138,72,152,72,162,63,160
10019 DATA 255,136,208,253,202,208,248,104,168,104,170,104,96,169,0,141,140
10020 DATA 128,169,200,141,139,128,169,150,141,3,208,32,17,129,206,3,208
10021 DATA 32,35,130,173,3,208,201,130,208,243,96,172,165,128,169,208,153
10022 DATA 19,204,169,7,153,19,216,152,72,32,124,130,32,35,130,32,35
10023 DATA 130,32,35,130,32,35,130,104,168,136,208,224,96,169,0,141,4
10024 DATA 212,169,220,141,1,212,169,10,141,5,212,169,8,141,6,212,169
10025 DATA 19,141,4,212,96,160,3,169,2,141,33,208,32,35,130,169,7
10026 DATA 141,33,208,32,119,131,32,35,130,169,1,141,33,208,136,208,229
10027 DATA 169,0,141,33,208,32,35,130,32,35,130,172,165,128,169,32,153
10028 DATA 19,204,206,165,128,172,165,128,208,27,104,104,160,8,185,71,136
10029 DATA 153,200,205,136,16,247,169,0,133,198,32,230,131,165,198,240,249
10030 DATA 76,232,128,234,32,124,130,96,32,204,131,56,173,137,128,237,149
10031 DATA 128,141,134,128,173,138,128,237,150,128,141,135,128,201,0,240,1
10032 DATA 96,173,134,128,240,1,96,56,173,1,208,237,13,208,141,134,128
10033 DATA 41,224,240,1,96,169,8,141,254,207,32,119,131,32,35,130,32
10034 DATA 35,130,238,254,207,32,35,130,32,35,130,238,254,207,32,35,130
10035 DATA 32,35,130,169,0,141,162,128,141,1,208,141,13,208,32,35,130
10036 DATA 24,173,166,128,105,200,141,166,128,173,167,128,105,0,141,167,128
10037 DATA 162,0,160,6,24,32,240,255,169,0,133,199,174,166,128,173,167

```



Commodore 64

10038 DATA 128,32,205,189,104,104,76,123,134,169,0,141,4,212,169,15,141
 10039 DATA 1,212,169,12,141,5,212,169,10,141,6,212,169,131,141,4,212
 10040 DATA 96,72,169,0,141,4,212,169,5,141,1,212,169,13,141,5,212
 10041 DATA 169,10,141,6,212,169,19,141,4,212,104,96,72,173,13,208,233
 10042 DATA 80,141,1,212,104,96,165,203,201,3,240,1,96,169,224,141,2
 10043 DATA 220,173,0,220,41,16,208,249,96,169,0,141,11,212,169,8,141
 10044 DATA 8,212,169,5,141,12,212,169,10,141,13,212,169,131,141,11,212
 10045 DATA 96,169,10,141,250,207,141,251,207,141,252,207,141,253,207,169,9
 10046 DATA 141,250,207,141,251,207,141,252,207,141,253,207,96,173,168,128,201
 10047 DATA 255,208,1,96,238,168,128,173,169,128,201,18,240,9,206,169,128
 10048 DATA 173,169,128,141,5,220,173,168,128,201,20,208,5,169,1,141,254
 10049 DATA 129,173,168,128,41,3,208,217,173,17,135,201,2,240,210,206,17
 10050 DATA 135,206,17,135,96,160,217,132,248,160,0,132,247,177,247,41,15
 10051 DATA 201,5,208,4,169,13,208,6,201,13,208,2,169,5,145,247,200
 10052 DATA 208,233,230,248,166,248,224,220,208,225,96,173,133,128,238,133,128
 10053 DATA 201,7,208,8,169,0,141,133,128,32,63,132,173,136,128,238,136
 10054 DATA 128,201,1,240,3,76,222,132,169,0,141,136,128,238,40,208,238
 10055 DATA 153,128,173,153,128,201,5,208,25,169,0,141,153,128,173,249,207
 10056 DATA 201,6,208,4,169,7,208,6,201,7,208,2,169,6,141,249,207
 10057 DATA 238,161,128,173,161,128,201,15,208,13,169,0,141,161,128,173,27
 10058 DATA 212,41,1,141,160,128,234,173,160,128,201,1,208,3,32,226,132
 10059 DATA 173,160,128,201,0,208,3,32,12,133,234,32,184,131,76,49,234
 10060 DATA 173,140,128,208,15,173,139,128,201,50,208,8,169,0,141,160,128
 10061 DATA 76,12,133,56,173,139,128,233,1,141,139,128,173,140,128,233,0
 10062 DATA 141,140,128,76,17,129,173,140,128,240,15,173,139,128,201,50,48
 10063 DATA 8,169,1,141,160,128,76,228,132,24,173,139,128,105,1,141,139
 10064 DATA 128,173,140,128,105,0,141,140,128,76,17,129,169,14,141,37,208
 10065 DATA 169,7,141,38,208,169,4,141,39,208,169,4,141,40,208,169,5
 10066 DATA 141,41,208,169,5,141,42,208,169,5,141,43,208,169,5,141,44
 10067 DATA 208,169,2,141,45,208,169,60,141,28,208,169,255,141,21,208,169
 10068 DATA 2,141,27,208,169,0,141,248,207,169,6,141,249,207,169,11,141
 10069 DATA 250,207,169,12,141,251,207,169,13,141,252,207,169,14,141,253,207
 10070 DATA 169,5,141,254,207,169,15,141,255,207,169,0,141,138,128,141,140
 10071 DATA 128,141,142,128,141,144,128,141,146,128,141,148,128,141,150,128,141
 10072 DATA 152,128,169,100,141,141,128,141,145,128,169,124,141,143,128,141,147
 10073 DATA 128,169,55,141,5,208,141,7,208,169,76,141,9,208,141,11,208
 10074 DATA 32,42,129,32,58,129,32,74,129,32,90,129,169,0,141,153,128
 10075 DATA 141,154,128,141,155,128,141,156,128,141,157,128,141,158,128,141,159
 10076 DATA 128,141,161,128,141,166,128,141,167,128,173,27,212,41,1,141,160
 10077 DATA 128,169,0,141,162,128,169,4,141,165,128,160,6,169,32,153,5
 10078 DATA 204,136,208,248,160,9,169,32,153,199,205,136,208,248,169,0,141
 10079 DATA 30,208,169,0,141,168,128,169,50,141,5,220,141,169,128,169,13
 10080 DATA 141,17,135,169,3,141,254,129,96,169,175,141,137,128,169,0,141
 10081 DATA 138,128,169,100,141,1,208,32,1,129,169,1,141,254,207,32,145
 10082 DATA 131,169,240,141,5,212,173,3,208,141,13,208,173,139,128,141,149
 10083 DATA 128,173,140,128,141,150,128,32,106,129,169,1,141,164,128,96,173
 10084 DATA 162,128,208,3,76,95,135,169,224,141,2,220,173,0,220,41,1
 10085 DATA 208,3,32,87,135,173,0,220,41,2,208,3,32,91,135,173,0
 10086 DATA 220,41,4,208,3,32,29,135,173,0,220,41,8,208,3,32,57
 10087 DATA 135,173,0,220,41,16,208,3,32,238,130,169,255,141,2,220,32
 10088 DATA 1,129,173,164,128,73,1,141,164,128,208,68,238,13,208,173,13
 10089 DATA 208,233,130,141,1,212,173,13,208,201,180,208,3,238,254,207,201
 10090 DATA 190,208,11,238,254,207,169,0,141,164,128,173,13,208,201,200,208
 10091 DATA 3,238,254,207,201,210,208,3,238,254,207,201,250,208,14,169,0
 10092 DATA 141,162,128,141,1,208,32,150,130,76,123,134,162,13,160,255,136
 10093 DATA 208,253,202,208,248,76,123,134,173,138,128,208,5,173,137,128,240
 10094 DATA 17,56,173,137,128,233,1,141,137,128,173,138,128,233,0,141,138
 10095 DATA 128,96,173,138,128,240,7,173,137,128,201,80,16,243,24,173,137
 10096 DATA 128,105,1,141,137,128,173,138,128,105,0,141,138,128,96,206,1
 10097 DATA 208,96,238,1,208,96,169,224,141,2,220,173,0,220,41,16,208
 10098 DATA 3,32,25,136,173,0,220,41,4,208,3,32,223,135,173,0,220

Commodore 64

```
10099 DATA 41,8,208,3,32,252,135,169,255,141,2,220,173,163,128,208,3
10100 DATA 76,16,135,206,13,208,32,158,129,32,173,131,173,13,208,201,220
10101 DATA 208,3,206,254,207,201,210,208,3,206,254,207,201,200,208,3,206
10102 DATA 254,207,201,190,208,8,206,254,207,169,1,141,164,128,201,133,208
10103 DATA 28,169,0,141,163,128,169,0,141,13,208,141,1,212,173,27,212
10104 DATA 41,3,208,8,169,1,141,162,128,32,65,134,76,16,135,173,164
10105 DATA 128,73,1,141,164,128,56,173,149,128,237,164,128,141,149,128,173
10106 DATA 150,128,233,0,141,150,128,76,106,129,173,164,128,73,1,141,164
10107 DATA 128,24,173,149,128,109,164,128,141,149,128,173,150,128,105,0,141
10108 DATA 150,128,76,106,129,173,163,128,240,1,96,169,5,141,254,207,169
10109 DATA 3,141,164,128,169,230,141,13,208,169,170,141,149,128,169,0,141
10110 DATA 150,128,32,106,129,169,1,141,163,128,173,30,208,32,145,131,96
10111 DATA 7,1,13,5,32,15,22,5,18,10,224
20488 REM VIDEOCONTROLLER UMSTELLEN.
50001 REM SPRITES AB 49152 = $C000
50002 REM SPRITESPOINTER 53240 = $CFF8
50003 REM VIDEOSCREEN AB 52224 = $CC00
50004 REM ZEICHENSATZ AB 53248 = $D000
50005 REM
50100 U=0:FOR I= 32768 TO 32900 :READ DA:U=U+DA:POKE I,DA:NEXT I:SYS 32768
50110 IF UC>14282 THEN PRINT"*****FEHLER IN DATA-ZEILEN":STOP
54990 REM *** MASCHINENPROGRAMM ***
55000 DATA 76,6,128,76,73,128,120,173,0,221,41,252,141,0,221,169,52
55001 DATA 141,24,208,169,204,141,136,2,169,97,141,17,3,169,128,141,18
55002 DATA 3,160,0,132,3,169,208,133,4,162,16,169,51,133,1,177,3
55003 DATA 72,169,48,133,1,104,145,3,200,208,239,230,4,202,208,234,169
55004 DATA 55,133,1,88,96,120,165,1,72,32,253,174,32,235,183,169,48
55005 DATA 133,1,160,0,138,145,20,104,133,1,88,96,165,20,72,165,21
55006 DATA 72,32,247,183,165,1,72,169,52,120,133,1,160,0,177,20,168
55007 DATA 104,133,1,88,104,133,21,104,133,20,76,162,179,255
55009 REM *** SPRITES 0 BIS 15 ***
55010 FOR I= 49152 TO 50175 :READ DA:POKE I,DA:NEXT I
55011 DATA 0,62,0,0,235,128,3,136,224,6,8,48,12,28,24,24,73
55012 DATA 12,24,8,12,49,8,70,48,8,6,98,20,35,127,227,255,98
55013 DATA 20,35,48,8,6,49,8,70,24,8,12,24,73,12,12,28,24
55014 DATA 6,8,48,3,136,224,0,235,128,0,62,0,64,0,0,0,0
55015 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
55016 DATA 0,14,0,0,63,0,0,63,0,0,31,0,0,30,0,0,0
55017 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
55018 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
55019 DATA 0,0,0,0,0,0,0,14,0,0,127,0,0,255,128,0
55020 DATA 255,128,0,255,192,1,255,192,0,255,192,0,127,128,0,30,0
55021 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
55022 DATA 0,0,0,0,192,0,0,0,0,0,0,0,0,0,0,0,0
55023 DATA 0,62,0,1,255,192,1,255,192,3,255,224,3,255,224,7,255
55024 DATA 240,7,255,240,7,255,240,3,255,224,3,255,224,1,255,192,0
55025 DATA 255,192,0,30,0,0,0,0,0,0,0,0,0,0,0,0
55026 DATA 160,0,0,0,0,0,0,31,0,0,255,192,3,255,224,7
55027 DATA 255,240,7,255,240,15,255,248,15,255,248,15,255,252,31,255,252
55028 DATA 31,255,252,15,255,248,15,255,248,7,255,240,3,255,240,1,255
55029 DATA 224,0,255,128,0,62,0,0,0,0,0,0,0,31,0,62,0
55030 DATA 0,255,128,3,255,224,7,255,240,15,255,248,31,255,252,31,255
55031 DATA 252,63,255,254,63,255,254,127,255,255,127,255,255,127,255,255,63
55032 DATA 255,254,63,255,254,31,255,252,31,255,252,15,255,248,7,255,240
55033 DATA 3,255,224,0,255,128,0,62,0,95,0,0,0,0,0,0,0
55034 DATA 0,0,4,0,16,8,0,0,20,0,20,19,0,100,16,128,132
55035 DATA 16,93,4,16,127,4,16,201,132,16,127,4,16,93,4,16,128
55036 DATA 132,19,0,100,20,0,20,8,0,8,4,0,16,0,0,0,0
55037 DATA 0,0,0,0,0,191,0,0,0,0,0,0,0,0,0,1,128
55038 DATA 192,2,0,32,4,0,16,11,0,104,16,128,132,16,93,4,16
55039 DATA 127,4,16,201,132,16,127,4,16,93,4,16,128,132,11,0,104
```


Commodore 64

```
55040 DATA 4,0,16,2,0,32,1,128,192,0,0,0,0,0,0,0,0
55041 DATA 0,255,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
55042 DATA 0,0,0,0,0,0,0,0,0,0,26,0,0,110,0,0,53
55043 DATA 128,0,30,0,0,63,0,0,21,128,0,0,0,0,0,0,0
55044 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,64,0,0
55045 DATA 0,0,0,0,0,0,0,0,0,0,0,0,8,0,0,0,128,0
55046 DATA 2,0,0,64,0,0,154,64,0,110,32,0,37,128,0,16,32
55047 DATA 1,181,32,0,21,128,0,64,0,2,17,64,0,6,0,0,0
55048 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,0
55049 DATA 0,0,0,0,0,0,18,8,8,0,0,128,24,2,6,0,0
55050 DATA 0,32,0,68,8,0,32,0,0,10,96,64,32,40,0,32,0
55051 DATA 0,2,32,0,0,2,0,72,0,6,0,32,0,0,0,32,0
55052 DATA 12,72,0,0,0,64,64,0,0,0,0,0,0,0,0,0,0
55053 DATA 0,0,0,0,21,0,0,90,0,1,170,0,22,170,0,26,170
55054 DATA 0,90,169,0,85,85,1,165,127,6,149,255,6,86,162,5,90
55055 DATA 170,26,170,170,21,170,170,71,250,149,85,171,170,90,170,170,106
55056 DATA 170,254,160,0,0,0,0,0,0,0,0,0,0,0,0,84,0
55057 DATA 0,165,0,0,170,64,0,170,148,0,170,164,0,106,165,0,85
55058 DATA 85,0,253,90,64,255,86,144,138,149,144,170,165,80,170,170,164
55059 DATA 170,170,84,86,175,209,170,234,85,170,170,165,191,170,169,31,106
55060 DATA 170,254,90,170,90,85,171,102,71,250,85,21,170,170,26,170,150
55061 DATA 5,90,166,6,86,146,6,149,215,1,165,127,1,85,85,0,86
55062 DATA 169,0,89,170,0,86,170,0,5,170,0,0,90,0,0,21,0
55063 DATA 0,1,0,0,0,0,0,0,0,0,0,95,191,170,169,170,170
55064 DATA 165,170,234,85,86,175,209,170,170,84,170,170,164,170,165,80,138
55065 DATA 149,144,255,86,144,253,90,64,85,85,0,106,165,0,170,164,0
55066 DATA 170,148,0,170,64,0,165,0,0,84,0,0,0,0,0,0,0
55067 DATA 0,0,0,0,0,0,0,159,0,0,0,0,0,0,0,0,0,0
55068 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,8
55069 DATA 0,0,28,0,0,28,0,0,8,0,0,0,0,0,0,0,0
55070 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
55071 DATA 0,0,0,255
55073 REM *** ZEICHENSATZ AENDERN ***
55074 DATA 27 ,0,0,0,0,32,0,0,0
55075 DATA 29 ,0,0,96,0,0,4,0,0
55076 DATA 135 ,0,0,0,0,1,7,14,15
55077 DATA 136 ,3,6,12,28,248,104,208,240
55078 DATA 150 ,11,6,7,14,24,56,112,192
55079 DATA 151 ,248,240,224,64,0,0,0,0
55080 DATA 165 ,4,221,221,221,220,255,0,255
55081 DATA 166 ,48,182,176,183,55,255,0,255
55082 DATA 176 ,254,254,253,253,251,251,247,247
55083 DATA 177 ,239,239,223,223,191,191,127,127
55084 DATA 178 ,127,127,191,191,223,223,239,239
55085 DATA 179 ,247,247,251,251,253,253,255,254
55086 DATA 180 ,0,0,0,0,0,15,63,255
55087 DATA 181 ,0,0,0,0,0,240,252,255
55088 DATA 205 ,127,191,223,239,247,251,253,254
55089 DATA 206 ,254,253,251,247,239,223,191,127
55091 DATA -1
55101 READ ZE
55102 IF ZE<>-1 THEN FOR I=0 TO 7: READ DA:SYS32771,(53248+(ZE*8)+I),DA:NEXT
55103 IF ZE<>-1 THEN 55101
60000 PRINT "SCORE: SCHILD: ";
60001 PRINT " ";
60002 PRINT " ";
60003 PRINT " ";
60004 PRINT " ";
60005 PRINT " ";
60006 PRINT " ";
60007 PRINT " ";
```



```

60008 PRINT"
60009 PRINT"
60010 PRINT"
60011 PRINT"
60012 PRINT"
60013 PRINT"
60014 PRINT"
60015 PRINT"
60016 PRINT"
60017 PRINT"
60018 PRINT"
60019 PRINT"
60020 PRINT"
60021 PRINT"
60022 PRINT"
60023 PRINT"
60024 PRINT"
60025 POKE 52224+999,160
60026 POKE 55296+999,5
60030 SYS33000
    
```

Inmac Kabel-Tester: defekte Leitungen im Nu entdeckt

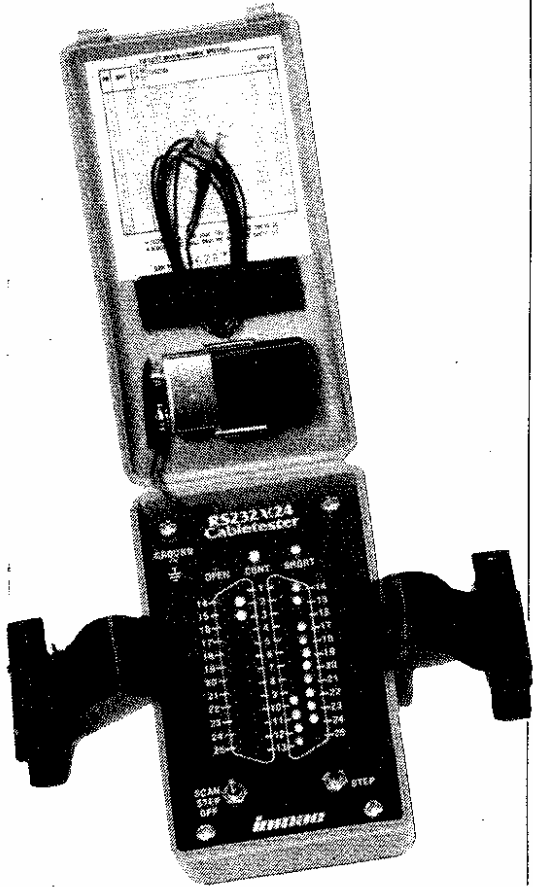
Kabel-Funktionstest in Sekundenschnelle – das bietet Inmac mit dem Kabel-Tester für RS232/V24-Kabel.

Noch nicht installierte Kabel lassen sich an beiden Enden mit dem Kabel-Tester verbinden. Alle Adern des Kabels werden nacheinander vollautomatisch getestet. Pro Sekunde erfolgt eine Messung. Die integrierte 9-Volt-Batterie versorgt das Testgerät mit Strom. Bis zu 70 Stunden lang. Der Testablauf läßt sich an den 50 LEDs genau verfolgen. Sie zeigen an, welche Leitungen Verbindung haben. Das jeweilige Testergebnis läßt sich an drei weiteren Leuchtdioden ablesen.

Bereits installierte Kabel prüft der zugehörige Tele-Motor. Er zeigt Testablauf und -ergebnis an, während der Kabel-Tester die Prüfpulse sendet. Hierfür müssen jedoch beide Geräte geerdet sein. Zusätzlich läßt sich mit dem Kabel-Tester die Steckerbelegung von Fremdkabeln blitzschnell entschlüsseln.

Den Inmac Kabel-Tester für RS232/V24-Kabel gib es – mit 30tägiger Probezeit – schon 24 Stunden nach dem Auftragseingang bei Inmac, Frankfurter Straße 103, 6096 Raunheim, Tel.: 0 61 42 / 40 60.

Oder bei Inmac, Heerdter Lohweg 51, 4000 Düsseldorf 11. Der Kabel-Tester kostet netto DM 798,-.



Software Infos

Aktuelle Arbeitsprogramme von Dynamics marketing für den Commodore 64

„FAKTURA 64“

Die Fakturierung ist eines der leistungsfähigsten Programme auf dem Commodore C64 überhaupt. Vorbildlich in der Benutzung, menuegesteuert, bildschirmorientiert und mit erklärenden Texten, die jederzeit abgerufen werden können, erfüllt das Programm alle Wünsche des Anwenders.

Das Programm verwaltet 1000 Datensätze, die wahlweise aufgeteilt, der Kundendatei, Lieferantendatei oder Artikeldatei zugeordnet werden. Festwerte wie Mehrwertsteuer, Rabatt- und Skontosätze werden einmal erfaßt, gespeichert und vom Programm automatisch eingesetzt. Das Programm unterstützt bis zu zwei Laufwerken, Druckeradressen sind einstellbar. Bis zu 9 Druckmasken können frei erstellt werden, der Ausdruck von Kopien ist selbstverständlich. Das Programm erlaubt, die Ausgabe der Daten im DATEV-Format. Um dem Datenschutz Rechnung zu tragen, verwehrt ein Paßwort Unbefugten den Zugriff auf das Programm. Die Daten werden automatisch in LAGER 64 und MAHN 64 übernommen. Programm auf Diskette.

Commodore 64



Brandneu auf dem Markt und mit 249,- DM empfohlenem Verkaufspreis geradezu sensationell preiswert ist der ASCOM Akustik-Koppler für den Commodore 64 der Firma DYNAMICS marketing GmbH. Das Gerät ist komplett anschlussfertig an den C 64 und wird mit einem sehr leistungsfähigen Betriebsprogramm auf Diskette geliefert.

Das Gerät besteht aus einem Steckmodul und dem Handset. Das Steckmodul wird auf den Expansion-Port des C 64 geschoben und beinhaltet das Interface und den Modulator. Das Handset wird mit einem Klettverschluss direkt auf dem Telefonhörer befestigt, eine ideale Voraussetzung für eine einwandfreie Übertragung. Es besteht aus zwei verstellbaren Auflagemuscheln, verschiebbar auf einer Gabel, um auch ungewöhnliche Hörerformen aufzunehmen.

Der Koppler arbeitet Voll-Duplex, wahlweise schaltbar in Originale oder Answer Modus. In der Schaltung ist eine ACIA 6551, die, über die Software eingestellt, für die Umwandlung der Daten beim Senden und Empfangen sorgt. Die Frequenzaufbereitung

erfolgt über einen Modulator mit nachgeschalteten Aktiv-Filtern. Über Dip-Schalter ist der Adressbereich des Kopplers über weite Bereiche einstellbar.

Das Betriebsprogramm CONTACT 64 steuert die Datenübertragung des ASCOM Akustik-Kopplers. Das Programm bietet folgende Möglichkeiten: englische oder deutsche Benutzerführung; Telefonnummernspeicher (Deutschland und Großbritannien); Parametereinstellung (Datenbits, Stopbits, Parität); Druckersteuerung (Protokollmodus); Upload - Download; Texteditor.

In Kürze wird das Hamburger Unternehmen für weitere Home- und Personal-Computer Akustik-Koppler auf den Markt bringen. Mit Geräten der Commodore-3000/8000-Serie, dem Spectrum, dem Atari 400/800 und den XL-Modellen, aber auch mit Computergeräten mit einer RS232-Schnittstelle kann man dann problemlos und kostengünstig an der Telekommunikation teilnehmen, die unverbindliche Preisempfehlung für diese ASCOM Akustik-Koppler ist 279,- DM.

Aktuelle Arbeitsprogramme

von Dynamics marketing GmbH

Maschine 64

Das Programm „Maschine 64“ ist ein leistungsfähiges Entwicklungswerkzeug für den Commodore 64. Einfache Bedienung und hohe Leistungsfähigkeit erfüllen voll die Ansprüche der Einsteiger ebenso wie die der Profis. Alle zur Maschinenprogrammierung notwendigen Hilfsmittel sind in dem Programm vereint.

Leistungsmerkmale:

- Editor

Erweiterung des Basic-Editors zur einfachen Eingabe und Korrektur von Assemblerprogrammen. Der Diskettenbetrieb wird durch zusätzliche DOS-Befehle erleichtert. Alle Werkzeuge werden über den Editor angewählt.

- Assembler

Der Assembler übersetzt das mit dem Editor erstellte Programm in Maschinencode. Er verarbeitet die normierte Syntax von MOS-Technology Inc., berücksichtigt Labels beliebiger Länge, Tabellen mehrerer Arten und volle Arithmetik bei der Operandenbildung.

- Reassembler

Der Reassembler erzeugt aus reinen Maschinencode eines vorhandenen Programmes ein vollwertiges Assemblerprogramm, das sofort mit dem Editor bearbeitet, analysiert oder mit dem Assembler neu übersetzt werden kann. Labels werden automatisch erzeugt, Tabellen berücksichtigt.

- Maschinensprachemonitor

Der Maschinensprachemonitor ermöglicht das Austesten von assemblierten Programmen auf der Prozessorebene. Dazu stehen Hilfsmittel wie Disassemblerliste, Hex-Dump, Speicher- und Registermodifikation, Break-Pointverarbeitung, Speichersuchoperationen, Umrechnungsfunktionen, Diskettenhandling und Miniassembler zur Verfügung.

- Diskettenmonitor

Der Diskettenmonitor liest und schreibt Sektoren der Diskette, stellt sie auf dem Bildschirm dar und ermöglicht Änderungen in diesem Sektor. Die Änderungs- und Darstellungsmöglichkeiten entsprechen denen des Maschinensprache-monitors.

Planer 64

Der „Planer 64“ ist ein Tabellenkalkulationsprogramm, das bei gezieltem Einsatz Taschenrechner, Bleistift und Papier ersetzt. Das Programm kann zur Ermittlung von Verkaufsstatistiken, Betriebskennzahlen und Kostenübersichten, der Lösung von Steuerproblemen bis hin zur Haushaltsplanung und Kontoführung eingesetzt werden. An beliebigen Stellen des Blattes eingegebene Zahlen können durch Angabe von Rechenanweisungen verknüpft werden. Änderungen von Einzelwerten führt zu einer Nachberechnung aller eingegebenen Verknüpfungen.

- 2600 Eingabepositionen
- Zahleneingabe und formatierte Darstellung
- Texteingabe zur Kommentierung
- Formeleingabe zur Verknüpfung von Positionen
- Editiermöglichkeiten des Blattes
- Einfache Cursorsteuerung
- Automatische Berechnung des Blattes
- Einfache Disketten- und Kassettenoperationen
- Selektive Druckausgabe

Lager 64

Das Lagerverwaltungsprogramm „Lager 64“ bietet alle Vorteile der elektronischen Datenverarbeitung, die auf dem Commodore 64 möglich sind. Das Programm ist menügesteuert, bildschirmorientiert und gibt bei Bedarf erklärende Texte zu den einzelnen Programmpunkten. Bei Einsatz der „Faktura 64“ werden die Daten übernommen.

- Bis zu 950 Artikel verwaltbar
- Such- und Sortierfunktion
- Druckausgabe
- 1 oder 2 Diskettenlaufwerke wählbar
- Druckeradressen frei einstellbar
- Passwordeingabe für Datenschutz
- Kompatibel zu „Faktura 64“ und „Lager 64“

DATEI 64

Die Datei 64 ist ein universelles Datenverwaltungsprogramm mit nahezu unbegrenzten Anwendungsmöglichkeiten, da der Anwender den Aufbau der Datei selbst bestimmen kann. Bei der Entwicklung wurde besonderes Gewicht auf eine einfache Bedienung und hohe Leistungsfähigkeit gelegt. So sind alle Funktionen menügesteuert und mit ausführlichen Anleitungen auf dem Bildschirm versehen. Besonders bei der vom Anwender zu definierenden Dateistruktur wird deutlich, wie einfach und übersichtlich die Benutzung ist, da ohne Vorkenntnisse eine sofort zu benutzende Dateiverwaltung erstellt werden kann. Stundenlanges Studium der Bedienungsanweisung und mühseliges Auspro-

bieren der Arbeitsweise wurden dem Anwender bewußt erspart.

Für einige Anwendungen einer Dateiverwaltung sind bereits Masken auf der Programmdiskette vorhanden.

- Daten eingeben, korrigieren, suchen, sortieren, drucken
- Alphanumerische Felder mit beliebigem Inhalt
- Numerische Felder für Zahlenwerte
- Betragsfelder für DM-Werte
- Fließkommfelder für Meßwerte
- Datums- und Uhrzeitfelder mit Plausibilitätskontrolle
- Sachbezogene Felder für Telefon und Kontonummern
- Merkerfelder für Ja/Nein-Eingaben oder drei definierbare Kennungen
- Ergebnisfelder für Berechnungen innerhalb eines Datensatzes
- Bis zu 16 Felder je Datensatz definierbar und einiges mehr.

Buchbesprechung

Das Maschinensprachebuch für Fortgeschrittene zum Commodore 64

In den letzten Monaten erscheinen laufend neue Bücher der Düsseldorfer Firma DATA BECKER auf dem Markt. Unter diesen befindet sich auch „Das Maschinensprachebuch für Fortgeschrittene zum Commodore 64“, welches von Lothar Englisch verfaßt wurde. Das Buch ist praktisch in drei voneinander unabhängige Teile aufgespalten. Folgende Bereiche der Programmierung des VC 64 in Maschinensprache werden dabei angesprochen:

1. Programmierung von Fließkommfunktionen, Fließkommaarithmetik
2. Möglichkeiten von Interruptquellen und daraus resultierend der Interruptprogrammierung
3. Programmierung eigener BASIC-Erweiterungen.

Dem Autor Lothar Englisch, bereits bekannt durch die Arbeit an anderen DATA-BECKER-Büchern (Das Maschinensprachebuch, 64 Tips & Tricks und 64 Intern), ist es gelungen, diese sehr anspruchsvollen Themen in einer einfach durchzuarbeitenden und leicht zu verstehenden Weise darzustellen, so daß auch Programmierer, denen das Verständnis der Hardware ihres Computers nicht so leicht fällt,

ihren Nutzen aus diesem Buch ziehen können.

Zu allen drei Themenbereichen finden sich zahlreiche Beispielprogramme (z. B. schnellere Mathematikroutinen, Darstellung von 16 (!) Sprites gleichzeitig auf dem Bildschirm, Programmierung eines REPEAT-UNTIL-Befehles und viele mehr!). Besonders gefallen hat mir, daß der Autor bei praktisch allen Programmen Hinweise zur Abänderung und damit zur Anpassung an die persönlichen Bedürfnisse des Lesers gibt. Neben diesen ausgearbeiteten Beispielen sind es aber auch gerade die auf weitergehende Programme hinweisenden Tips von Lothar Englisch, welche das Buch lesenswert machen.

In einem der letzten Kapitel namens „Die Vektoren des Betriebssystems“ werden die Ein/Ausgaberroutinen des Betriebssystems beschrieben. Dieses Kapitel verdient deshalb besondere Erwähnung, weil alle Routinen mit einfachen Worten sehr ausführlich beschrieben werden. Insbesondere wird die Parameterübergabe an bzw. von diesen Routinen erklärt, ein Punkt, welche ich bei der Besprechung des Maschinensprachebuches (64'er 7/84) noch kritisieren mußte. Mit den in diesem Buch gegebenen

Commodore 64

Informationen sollte es nun aber jedem Maschinenspracheprogrammierer möglich sein, bei der Programmierung von Ein/Ausgabeprogrammen die bereits vorhandenen Routinen optimal zu nutzen.

Ein Punkt, welcher an den DATA-BECKER-Büchern oft zu recht kritisiert wurde, ist, daß die Werbung für andere DATA-BECKER-Produkte in ihren Büchern einen zu großen Raum einnimmt. Zwar besitzt auch das hier

EDITOR

Ein leistungsstarker
Editor in Basic

Dieser Editor läßt die Standard-6502-Schreibweise zu und verfügt darüber hinaus über einige Besonderheiten.

Hier eine kurze Befehlsbeschreibung:

APPEND: 1. Dieser Befehl lädt bereits geschriebene Editorfiles.

2. Befindet sich bereits ein File im Speicher, besteht die Möglichkeit, Programme aneinander zu hängen.

SAVE: Es werden geschriebene Files gesichert.

VERIFY: Es wird das Programm, das sich im Speicher befindet, mit einem auf Diskette befindlichen verglichen.

ADD: Gestattet die Erstellung von Editorlistings im 6502-Format.

COPY: Gestattet das Kopieren von einem Bereich in einen anderen.

Syntax: Es wird 1. und letzte Zeile des zu kopierenden Bereichs angegeben. Es muß die Zeile angegeben werden, hinter der der zu kopierende Block stehen soll.

INSERT: Fügt Zeilen ein. Es muß die Zeile angegeben werden, hinter der eingefügt werden soll. Es wird solange eingefügt, bis einmal RETURN ohne Eingabe eingegeben wird.

EDIT: Editiert das Listing zeilenweise, einzeln, oder von ... bis.

DELETE: Löscht Zeilen. Es kann die 1. und letzte Zeile angegeben werden. Bei keiner Angabe wird das Listing gelöscht.

LIST: Listet das Listings vollständig oder teilweise. Der Listvorgang kann mit ‚S‘ unterbrochen werden, um ihn mit ‚Q‘ ganz abzubrechen oder mit irgendeiner Taste fortzusetzen.

PRINTER: Listet das Listing auf den

Drucker. Der Filename wird auf jeder Seite gedruckt. Es besteht die Möglichkeit, die erste Seite festzulegen. Wird keine Eingabe gemacht, wird automatisch ‚O‘ angenommen.

QUIT: Verläßt den Editor, aber nur um in den Assembler oder den Disassembler zu kommen.

M: DISKMENUE

-1- LOESCH DATEI: Es werden Dateien gelöscht. Wenn ‚↑‘ statt einem Namen eingegeben wird, gelangt man wieder ins Menü ohne zu Löschen.

-2- DATEI UMBENENNEN: Filenamenn lassen sich ändern.

-3- DATEI KOPIEREN: Files lassen sich unter einem anderen Namen noch einmal ablegen.

-4- DISKETTE AUFRÄUMEN: Ist gleich dem Validatebefehl.

-5- NEUE DISK: Formatieren und Einrichten einer neuen Diskette.

-6- ENDE MENUE: Sprung zum Hauptmenü.

-7- DATEI LADEN: Wie APPEND

FILE CHANGE: Man kann das aktuelle Gerät (Disk/Cassette) umschalten, z. B. falls beide Geräte existieren, Dateien auf Cassette zu sichern.

Noch einmal zu APPEND. Mit ‚↑‘ kann man zum Menü zurück, ohne ein File zu laden.

Beim Abspeichern auf Diskette wird jedem Editorfile eine Kennung für den Editor zugewiesen. Dieses dient der Übersichtlichkeit beim Lesen des Di-

besprochene Buch einen „Anhang“ von 10 Seiten, in welchem die anderen DATA-BECKER-Bücher bzw. Software dieser Firma beschrieben wird, aber im Text des Buches selber ist diese Eigenwerbung nicht mehr so aufdringlich vorhanden. Es wurden zwar alle Assemblerlistings mit PROFI-ASS angefertigt und abgedruckt, dies geschieht aber, ohne dauernd die Vorzüge dieses Programmes zu preisen.

Zusammenfassend möchte ich sagen, daß ich dieses Buch aufgrund der vielen Tips und Tricks und der vielen, sehr schön verwertbaren Beispiele weiterempfehlen kann. Der gut lesbare Druck und die saubere Verarbeitung sind weitere Pluspunkte, die den Preis von 39,- DM wohl rechtfertigen. Das Maschinensprachebuch für Fortgeschrittene zum Commodore 64, von Lothar Englisch, 206 Seiten, DATA BECKER GmbH - 1984, ISBN 3-89011-022-3.

Von: Wolfgang Willing, Weidenhäuser Str. 28, 3550 Marburg.

rektorics. Die Kennung besteht aus „E“.

Zum PRINTER. Dieses Programm ist zum Betrieb mit einem VC1526 geschrieben worden. Wer keinen VC1526 besitzt, sollte folgende Zeilen ändern:

1310 OPEN4,4,0: Den Rest löschen

1380 PRINT#4 Den Rest löschen

1390 FOR I = 1TO66-N3:PRINT-

#4:NEXT:CLOSE4:GOTO280: Einfügen

1440 FOR I = 1TO6:PRINT#4:NEXT: Einfügen

1420 IFN3<66...: Ändern

Es brauchen die Befehlsörter nicht ganz eingegeben werden, der inverse Buchstabe reicht.

Wenn im ADD-Modus Befehle hinter der Zeilennummer stehen und RETURN gedrückt wird, werden diese ausgeführt.

Wird in einer Zeile kein Label verwendet, so muß zwischen der Zeilennummer und der Mnemonik ein Leerzeichen stehen. Dto. zwischen Labels und Mnemonik. Labels dürfen eine maximale Länge von 8 Zeichen haben. Es besteht die Möglichkeit der Kommentierung. Die Kommentare müssen mit einem „;“ am Anfang gekennzeichnet sein.

ORG gibt die Startadresse des Assemblerprogramms an. ORG muß vor der ersten Assemblerzeile stehen. Die Eingabe ist in Hex- und Dezimal möglich. Hexzahlen werden mit einem vorwegführenden „\$“ gekennzeichnet.

EQU definiert Labels, d. h. Adressen werden Namen zugewiesen. Auf Labels, die durch EQU definiert wurden, kann addiert oder subtrahiert werden. DFB = Define Byte ist eine DATA-Anweisung für Assembler. Es können Zahlen oder Texte eingegeben werden. Sie müssen durch „“ getrennt werden. Texte müssen in „“ eingeschlossen sein. DFB muß auch einen Namen als Label haben. Weiterhin kann bei den Lade- und

Vergleichsbefehlen, bei denen eine Konstante verwendet wird, auch ein ASCII-Zeichen, in „“ eingeschlossen, eingegeben werden, z. B. LDA #'A' oder CMP #','. Dann gibt es noch die Möglichkeit der Adreßtrennung in Low- und Highbyte. „<“ gibt das Highbyte an und „>“ das Lowbyte.

Eine Befehlsfolge als Beispiel:

```
1 ORG $0000; Startadresse
2 TEST EQU $FFD2; Testadresse
```

```
3; ★★★★★★★★★★★★★★
4 LDA #<TEST; Lade Highbyte
5 LDY #>TEST; Lade Lowbyte
6 STY $BA
7 STA $BB
8 NAME DFB $0D,$93,'DIES IST
   EIN TEST',$0D,$0D
9 LDA #'A'
10 JSR TEST
Dieses kleine Beispielprogramm zeigt
wohl am besten die Möglichkeiten des
Editors.
```

EDITOR

```
70 POKE45,24:POKE46,46:CLR
80 CLR:DIMA$(1000):POKE53280,7:POKE53281,7
90 POKE828,PEEK(55):POKE829,PEEK(56):FI=PEEK(186)
110 L$=" ":FORI=1TO5:L$=LEFT$(L$+L$,21):NEXT:L$=L$+" "
200 PRINT" "SPC(11)"***** EDITOR *****"
210 PRINT" *APPEND *          *SAVE * [A] [-] [E]"
220 PRINT" *VERIFY * [A] [-] [E] *ADD *COPY"
230 PRINT" *INSERT * [A] *EDIT * [A] [-] [E]"
240 PRINT" *DELETE * [A] [-] [E] *LIST * [A] [-] [E]"
250 PRINT" *PRINTER * [A] [-] [E] *JIT ->ASS +DISASS"
260 PRINT" *DISKMENU * *FILE CHANGE"
280 IFB=1ORB=5ORB=12THENA$="L":GOTO300
290 INPUT" *BEFEHL *";A$
300 PRINT:RESTORE:FORB=1TO12:READB$:IFLEFT$(A$,LEN(B$))<>B$THENNEXT
310 IFB<>2ANDB<11THENGOSUB5000
320 IFB=10ANDFI<>8THENPRINT" * * * * *":GOTO280
330 ONBGO TO400,600,500,700,800,1000,1200,900,1300,3000,3500,1100,540,200
400 REM *APPEND*
405 IFLA=0THENLA=1:GOTO419
410 PRINT" *DATEI * *BI *NDEN ODER NEU *LA *DEN"
411 INPUTG1$:IFG1$<>"BI"ANDG1$<>"LA"THENPRINT" * * * * *":GOTO410
412 IFG1$="BI"THEN419
413 IFG1$="LA"THEN800
419 GOTO6000
420 INPUT#1,A$
430 IFN2<NTHENFORC=NTON2+1STEP-1:A$(C+1)=A$(C):NEXT
440 N2=N2+1:N=N+1:A$(N2)=A$:IFST=66ANDFI=8THENGOSUB2500
445 IF(64ANDST)=0THEN420
450 CLOSE1:GOTO280
500 REM *SAVE*
501 IFFI=8ANDOP=0THENOPEN2,8,15:OP=1
505 INPUT"FILENAME";F$:IFFI=8THENF$="E "+F$+",S,W"
510 OPEN1,F1,1,F$:SA=1:GOSUB2500:SA=0:IFEN<>0ANDFI=8THEN5300
520 FORC=N1TON2:PRINT#1,CHR$(34)A$(C):NEXT:CLOSE1.
540 CLOSE2:OP=0:PRINT" * * * * * ASSEMBLER EINLADEN ?":WAIT198,3:GETG$
550 IFG$="J"THENLOAD"ASSEMBLER",F1
555 PRINT" *DISASSEMBLER":WAIT198,3:GETG$
560 IFG$="J"THENLOAD"DISASSEMBLER",F1
565 GOTO200
600 REM *** ADD ***
610 LA=1:C=N:PRINT" * * * * *"
620 GOSUB4000:IFBFTHEN300
630 PRINTL$:GOSUB4500
```


Commodore 64

```
640 A$(C)=A$:N=C:GOTO620
700 REM *LIST*
710 PRINT"Q";FORC=N1TON2:A#=A$(C):GOSUB4500
720 GETG$:IFG$="S"THENWAIT198,1:GETG$
730 IFG$="Q"THEN280
740 NEXT:GOTO280
800 REM *DELETE*
810 D=N2-N1+1:IFD>10THENPRINT"33 ARE YOU SURE ? 33":WAIT198,7:GETG$:IFG$="N"THEN
280
820 N=N-D
830 FORC=N1TON:A$(C)=A$(C+D):NEXT
835 N2=N:N1=N
840 IFG1$="LA"THENG1$="":GOTO419
850 LA=0:GOTO280
900 REM *VERIFY*
910 VE=1:GOTO6000
920 PRINT"Q";VE=0:FORC=N1TON2:INPUT#1,A$
930 IF(A$(C)>A$(C))OR(64ANDST)AND(C<N2)THENPRINT"33 ?VERIFY ERROR IN";C;"33":ER=1
940 NEXT:CLOSE1:IFER=0THENPRINT"** VERIFY OK **"
950 ER=0:GOTO280
1000 REM *EDIT*
1010 FORC=N1TON2:PRINT"33000":S=PEEK(214):A#=A$(C):GOSUB4500
1015 PRINT"Q";:IFPEEK(214)>STHEN1015
1020 A$="":SYS42336
1030 FORI=515T0600:IFPEEK(I)THENA#=A#+CHR$(PEEK(I)):NEXT
1040 GOSUB4100:IFBFTHEN300
1050 A$(C)=A$:NEXTC:GOTO280
1100 REM *COPY*
1110 INPUT"COPY-BEREICH";A$
1120 GOSUB5000:INPUT"WOHIN (ZEILENNUMMER)";C
1130 IFC>NTHENC=N
1140 D=N2-N1+1
1150 IFC<NTHENFORI=NTOC+1STEP-1:A$(I+D)=A$(I):NEXT
1160 N=N+D:IFN1>CTHENN1=N1+D
1170 FORI=C+1TOC+D:A$(I)=A$(N1):IFN1=CTHENN1=N1+D
1180 N1=N1+1:NEXT
1190 GOTO280
1200 REM *INSERT*
1210 C=N1
1220 GOSUB4000:IFBFTHEN300
1230 PRINTL$:GOSUB4500
1240 FORI=NTOCSTEP-1:A$(I+1)=A$(I):NEXT:N=N+1:A$(C)=A$
1250 GOTO1220
1300 REM **** PRINTER ****
1305 INPUT"FILENAME";F$:N3=0
1306 INPUT"PAGE-ANFANG";PG
1310 OPEN4,4,0:OPEN3,4,3:PRINT#3,CHR$(66):PRINT#4,CHR$(147)CHR$(141);
1320 GOSUB1450
1330 FORC=N1TON2:A#=A$(C)
1340 PRINT#4,RIGHT$(" "+STR$(C),3);
1350 FORI=1T07:IFMID$(A$,1,1)=";"THENI=7:PRINT#4,A$:NEXTI:GOSUB1400:GOTO1370
1355 I$=MID$(A$,I,1):PRINT#4,I$;:IFI$<>" "THENNEXTI:I=I-1.
1360 PRINT#4,CHR$(16)"15"MID$(A$,I+1)" ":GOSUB1400
1370 NEXTC
1380 PRINT#4:PRINT#4,CHR$(19)CHR$(141):CLOSE4:CLOSE3:GOTO280
1400 N3=N3+1:IFLEN(MID$(A$,I+1))+15>80THENN3=N3+(INT(LEN(MID$(A$,I+1)))/80)
1420 IFN3<64THENRETURN
1430 N3=0:PG=PG+1
1450 PRINT#4,CHR$(14)CHR$(16)"15";F$:CHR$(15):PRINT#4,CHR$(16)"68PAGE";PG:RETURN
```



```

1700 REM ***** DIREKTORY *****
1701 PRINT "          VORHANDENE DATEIEN          "
1702 PRINT "SIE KOENNEN ALLE DATEIEN ODER NUR GRUP- PEN AUSGEBEN LASSEN."
1703 PRINT "ALLE DATEIEN: 'RETURN-TASTE' DRUECKEN
1704 PRINT "GRUPPEN VON DATEIEN : Z.B. 1.BUCHSTABEN + *"
1705 PRINT "DIES GILT AUCH ZUM LOESCHEN VON DATEIEN"
1706 IFOP=0THENOPEN2,8,15:OP=1
1710 INPUT "      NAME DER DATEI ODER RETURN ";F$: IFF$=" "THENF$="*"
1711 ZZ=1:OPEN1,8,0,"$0:E "+F$
1715 GET#1,T1$: IFST<>0THEN1760
1720 IFT1$<>CHR$(34)THEN1715
1725 T$="":FORTT=1TO2:GET#1,T1$:T$=T$+T1$:NEXT
1727 IFT$<>"E "THEN1715
1728 T$=""
1730 GET#1,T1$: IFT1$<>CHR$(34)THENPRINTT1$;T$=T$+T1$:GOTO1730
1740 GET#1,T1$: IFST=64THEN1760
1745 IFINT(ZZ/2)*2=ZZTHENPRINT:GOTO1750
1746 PRINTSPC(22-LEN(T$))
1750 ZZ=ZZ+1: IFZZ=10THENPOKE198,0:PRINTSPC(7)"WEITER J/N"
1752 IFZZ=10THENGETJ$: IFJ$<>"J"ANDJ$<>"N"THEN1752
1753 IFJ$="J"THENZZ=1:GOTO1715
1754 IFJ$="N"THENGOTO1760
1755 IFST=0THEN1715
1760 CLOSE1:PRINT: IFNU>1ANDNU<6THENRETURN
1780 IFNU=1THENPRINTSPC(5)"***** DATEI LOESCHEN *****"
1785 PRINT "      ↑ = ZUM MENUE      "
1790 INPUT "      NAME DER DATEI ";F$: IFF$=" "THENPRINT" ":GOTO1790
1792 IFF$="↑"THENRETURN
1794 IFLEN(F$)>12THENGOSUB2810:PRINT" ":GOTO1790
1796 IFNU=1THENGOSUB2200
1798 RETURN
1900 REM **** DATEN LOESCHEN ****
1910 GOSUB1700: IFF$="↑"THENRETURN
1915 IFJ$="N"THENRETURN
1920 OPEN1,8,15,"S:E "+F$:GOSUB2500:CLOSE1:RETURN
2000 REM **** DATEI UMBENENNEN ****
2005 PRINT " ":GOSUB1700
2010 PRINTSPC(5)"***** DATEI UMBENENNEN *****"
2015 INPUT "NEUER DATEINAME";NT$: IFLEN(NT$)>12THENGOSUB2810:PRINT" ":GOTO2015
2020 INPUT "ALTER DATEINAME";AT$
2030 OPEN1,8,15,"R:E "+NT$+"=E "+AT$:GOSUB2500:CLOSE1
2040 RETURN
2100 REM **** DATEI COPIEREN ****
2110 PRINT " ":GOSUB1700
2120 PRINTSPC(5)"***** DATEI COPIEREN *****"
2125 INPUT "NEUER DATEINAME";NT$: IFLEN(NT$)>12THENGOSUB2810:PRINT" ":GOTO2015
2130 INPUT "ALTER DATEINAME";AT$
2140 OPEN1,8,15,"C:E "+NT$+"=E "+AT$:GOSUB2500:CLOSE1
2150 RETURN
2200 REM **** W I R K L I C H ? ****
2210 PRINT "          "
2220 PRINT "SOLL DIES WIRKLICH GESCHEHEN J/N ?"
2230 GETJ$: IFJ$<>"J"ANDJ$<>"N"THEN2230
2240 IFJ$="N"THENRETURN
2250 PRINT "W I R K L I C H ??????"
2260 GETJ$: IFJ$<>"J"ANDJ$<>"N"THEN2260

```


Commodore 64

```
2270 RETURN
2300 REM **** VALIDATE DISK ****
2310 PRINT"***** **** AUFRAEUMEN DER DISKETTE *****"
2315 PRINT"*****"
2320 PRINT"***** WERDEN ALLE NICHT GESCHLOSSENEN DATEIEN GELOESCHT";
2330 PRINT"*****"
2340 GOSUB2200: IFJ$="N"THENRETURN
2390 OPEN1,8,15,"V":GOSUB2500:CLOSE1:RETURN
2500 REM ***** DISK FEHLERMELDUNG *****
2505 IFFI=1THENRETURN
2510 INPUT#2,EN,EM$,ET,ES
2520 IFEN=>0ANDSA=1THEN2550
2530 PRINTSPC(5)"***** DISK FEHLERMELDUNG *****"
2540 PRINT"EN";EN;"EM";EM$;"ET";ET;"ES";ES:POKE198,0:PRINTSPC(20)"DRUECKE TASTE":WAIT198
,1
2550 RETURN
2700 REM **** DISK FORMATIEREN ****
2710 PRINT"***** **** DISK FORMATIEREN ****"
2720 INPUT"NEUE DISK EINGELEGT J/N";J$: IFJ$<>"J"ANDJ$<>"N"THENPRINT"*****":GOTO27
20
2725 IFJ$="J"THEN2740
2730 INPUT"WEITER/ABBRUCH W/A";J$
2735 IFJ$<>"W"ANDJ$<>"A"THENPRINT"*****":GOTO2730
2736 IFJ$="W"THEN2710
2737 RETURN
2740 INPUT"NAME DER DISKETTE";NA$
2750 IFLen(NA$)>16THENGOSUB2810:PRINT"*****":GOTO2740
2760 INPUT"KENNUNG DER DISK (2 STELLIG)";NR$
2770 IFLen(NR$)>2THENPRINT"***** KENNUNG ZU LANG *****":GOTO2760
2780 OPEN1,8,15,"N:"+NA$+",""+NR$:GOSUB2500:CLOSE1:RETURN
2800 REM **** FEHLERLISTE ****
2810 PRINT"*****":PRINT"NAME ZU LANG !*****":RETURN
3000 NU=0:PRINT"***** DISK MENUE *****"
3010 PRINT"0000-1- LOESCH DATEI"
3020 PRINT"0001-2- DATEI UMBENENNEN"
3030 PRINT"0002-3- DATEI COPIEREN"
3040 PRINT"0003-4- DISKETTE AUFRAEUMEN"
3045 PRINT"0004-5- NEUE DISK"
3050 PRINT"0005-6- ENDE MENUE"
3052 PRINT"0006-7- DATEI LADEN"
3054 IFOP=0THENOPEN2,8,15:OP=1
3060 GETNU$: IFVAL(NU$)=0THEN3060
3065 NU=VAL(NU$): IFNU=0ORNU>7THEN3060
3070 IFNU=6THENNU=0:GOTO200
3075 IFNU=7THENB=1:GOTO330
3080 ONNUGOSUB1900,2000,2100,2300,2700
3090 GOTO3000
3500 REM *** FILE CHANGE ***
3510 PRINT"DISK/ASSETTE"
3520 GETG$: IFG$="D"THENFI=8:GOTO3540
3530 FI=1: IFG$<>"C"THEN3520
3540 GOTO200
4000 A$="": I=1: C=C+I: C$=RIGHT$(" "+STR$(C),3)
4005 PRINT"*****"
4010 S=PEEK(214): PRINTC$LEFT$(A$,I-1)" "LEFT$(MID$(A$,I,1)+" ",1)" "MID$(A$,I+1)
" "
4015 IFPEEK(214)>STHENPRINT"*****":GOTO4015
4020 WAIT198,15:GETG$:J=ASC(G$)
4030 IFJ=23THEN4090
```

```

4040 IFJ=157THENI=I+(I>1):GOTO4010
4050 IFJ=148THENA#=LEFT$(A$,I-1)+" "+MID$(A$,I):GOTO4010
4060 IFJ=20THENA#=LEFT$(A$,I-1+(I>1))+MID$(A$,I):I=I+(I>1):GOTO4010
4070 IFJ=13THEN4100
4080 A#=LEFT$(A$,I-1)+G$+MID$(A$,I+1)
4090 I=I+1+(I>LEN(A#)):GOTO4010
4100 FORI=1TOLEN(A#):IFMID$(A$,I,1)<>" "THENNEXT:BF=-1:RETURN
4110 IFMID$(A$,I+1,1)=" "THENA#=LEFT$(A$,I)+MID$(A$,I+2):GOTO4110
4120 BF=0:RETURN
4500 PRINTRIGHT$( " "+STR$(C),3);:AI$=""
4510 FORI=1TO8:IFMID$(A$,I,1)=":"THENI=8:PRINTA$:RETURN
4515 I$=MID$(A$,I,1):PRINTI$:IFI$<>" "THENAI$=I$+AI$:NEXT:I=I-1
4520 PRINTSPC(10-LEN(AI$))+MID$(A$,I+1)" "
4525 IFMID$(A$,LEN(AI$)+2,3)="ORG"THENGOSUB5200:RETURN
5000 D=0:FORI=1TO9:J=ASC(MID$(A$,I,1)+"■"):IFJ=45THEND=1
5010 IFJ<48ORJ>57THENNEXT
5020 N1=VAL(MID$(A$,I))
5030 N2=VAL(MID$(A$,I+LEN(STR$(N1))))
5040 IFN1=0THENN1=1ANDB<>7:N2=N
5050 IFDTHENN2=N1:N1=1
5060 IFRIGHT$(A$,1)="-"THENN2=N
5070 IFN2<N1THENN2=N1
5080 IFN2>NTHENN2=N
5090 IFN1>NTHENN1=N
5100 RETURN
5190 REM *** SPEICHERPLATZBEGRENZUNG ***
5191 REM ** FUER ASSEMBLER BERECHNEN: **
5192 REM * LB=(828)=(55):HB=(829)=(56) *
5200 IFMID$(A$,LEN(AI$)+6,1)="#"THEN5260
5210 Z=VAL(MID$(A$,LEN(AI$)+6,6))
5220 HB=INT(Z/256):LB=Z-HB*256
5230 IFHB<=PEEK(829)THENPOKE829,HB:POKE828,LB
5240 IFLB<>PEEK(828)THENPOKE828,LB
5250 RETURN
5260 Z#=MID$(A$,LEN(AI$)+7,4):Z=0
5270 FORI=0TO3:Z1=(ASC(MID$(Z$,I+1,1))-48)*16+(3-I)
5280 IFASC(MID$(Z$,I+1,1))>57THENZ1=(ASC(MID$(Z$,I+1,1))-55)*16+(3-I)
5290 Z=Z+Z1:NEXT:GOTO5220
5300 REM *** FILE EXISTS ***
5310 CLOSE1
5320 PRINT"■ FILE UEBERSCHREIBEN (J/N)■"
5330 GETG$:IFG$="J"THENF$="@:"+F$:GOTO510
5340 IFG$<>"N"THEN5330
5350 PRINT"■ FILE NICHT GESPEICHERT■"
5360 GOTO280
6000 IFFI=8THEN6050
6002 INPUT"FILENAME":F$
6005 PRINTSPC(5)"CASSETTE ZURUECKSPULEN"
6010 PRINT"BIS ZUM ANFANG DES ASSEMBLERFILES!"
6020 PRINT"■ DANN ■ P. ■ DRUECKEN!"
6030 WAIT198,1:GETG$:IFG$<>"P"THEN6030
6040 PRINT"OK":OPEN1,1,0,F$:IFVE=1THEN920
6045 GOTO420
6050 GOSUB1700:IFF$="↑"THENB=0:GOTO200
6100 OPEN1,8,2,"E "+F$+",S,R":IFVE=1THEN920
6110 GOTO420
10000 DATAAP,A,S,L,D,E,I,V,P,M,F,C,Q
READY.

```


VC-20

Powerpack

**Powerpack läuft
ohne
Erweiterung auf dem
VC-20!**



Sie sind in das Labyrinth des gefährlichen Energiemonsters Enzigo eingedrungen, um es zu beseitigen.

Enzigo bekommt genügend Energie aus den im Labyrinth verstreuten rot gekennzeichneten Energiezellen. Solange der Vorrat reicht und Energiezellen vorhanden sind, ist Enzigo unverwundbar.

Ihre Aufgabe ist es, Enzigos Energiezellen zu beseitigen, um ihn zu vernichten. Aber aufpassen – fünf Totenköpfe sind im Labyrinth verstreut. Diese dürfen nicht berührt werden. Zwei Gespenster durchkämmen das Labyrinth nach Eindringlingen. Enzigo ist in der Lage, seine Gespenster auf jeden beliebigen Ort seines Labyrinthes einzusetzen. Also, abwarten wo seine Gesellen eingesetzt werden und überraschend auftauchen. Ist dies geschehen, versuchen Sie geschickt auszuweichen. Manchmal kommt es vor, daß Sie zwischen Gespenst und Hindernis eingekellt werden. Dann gibt es nur eine Rettung: die F7-Taste. Sie bewirkt, daß Sie auf Ihre anfängliche Startposition zurückversetzt werden. Leider hat die Sache einen Haken. Sie können diese Chance nur einmal pro Spiel verwenden. Es sei denn – Sie schaffen es, die im Labyrinth verteilten Buchstaben S u p e r in der richtigen Reihenfolge zu vertilgen. In diesem Falle hätten Sie die Möglichkeit, ein zweites Mal von der F7-Taste Gebrauch zu machen.

Wenn es Ihnen gelungen ist, alle Energiezellen zu beseitigen, müssen Sie nur noch das Monster Enzigo vernichten. Steuerung: >W< oben, >Space< unten, >A< links, >D< rechts.

```
0 REM=====
1 REM=  POWERPACK  =
2 REM=====
3 REM COPYRIGHT BY
4 REM WINKLER GABRIEL
```

```
100 FORT=0TD15:READA:FORB=0TO7:READB:
POKE7168+S+A*8,B:NEXTS,T
110 DATA32,,,,,,,,
120 DATA45,24,24,24,24,24,24,24
130 DATA46,,,,,255,255,,,
140 DATA47,24,24,60,255,255,60,24,24
150 DATA48,251,251,251,,,223,223,223,
160 DATA49,,,,,36,24,24,36,
170 DATA50,56,60,186,162,186,162,186,198
180 DATA51,66,52,110,101,138,123,210,153
190 DATA52,66,60,126,219,255,255,219,153
200 DATA53,60,70,219,219,199,215,90,60
210 DATA54,60,66,223,199,223,66,126,60
220 DATA55,60,70,219,219,199,223,94,60
230 DATA56,60,90,219,219,219,219,66,60
240 DATA57,60,70,223,199,247,247,70,60
250 DATA58,126,219,219,255,255,126,90,90
260 DATA59,24,26,126,88,24,60,36,102
500 POKE36879,14:POKE36878,15
510 PRINT"J";TAB(7);"=POWERPACK"
520 PRINT"XXXXXXXXXXXXLADEN SIE JETZT BITTE  DAS HAUPTPROGRAMM"
530 POKE198,0:WAIT198,1
```

Hauptprogramm

```
100 M=3:GOTO7540:DATA,2,6,10,11,15,19,28,32,33,35,37,39,41,42,43,44,45,46,48,50,
52,54
110 DATA55,57,61,70,74,76,77,79,83,87,88,92,96,98,99,101,118,120,121,123,125,127
,128
120 DATA129,130,131,132,133,134,135,136,138,140,142,143,145,162,164,165,167,168,
169,173
130 DATA174,175,176,177,178,182,183,184,186,187,208,209,210,211,212,213,214,215,
216,217
140 DATA218,219,220,221,222,223,224,225,226,227,228,229,230,,,9,86,90,95,99,172,1
76,185
150 DATA189,262,266,271,275,352,361,,,88,352,54,318:POKE36869,255
400 FORT=0TD98:READA:POKE38674-A,4:POKE38675+A,4:POKE7954-A,48:POKE7955+A,48:NEX
T
```

```

410 FORT=7747T08163:IFPEEK(T)=32THENPOKET+30720,2:POKET,49
420 NEXT:FORT=0T015:READA:A=A+7774:B=A+30720:C=46:D=1:E=22:IFRND(1)>.5THENC=45:D
=22:E=1
430 POKEA+E,32:POKEA-E,32:POKEB,6:POKEA,47:POKEB+D,3:POKEB-D,3:POKEA+D,C:POKEA-D
,C:NEXT
500 FORT=1T05:READA
510 B=A+INT(RND(1)*11):C=A+INT(RND(1)*11):B=B+7748:C=C+7748
520 IFPEEK(B)<>49ORPEEK(C)<>49ORB=CTHEN510
530 POKEB+30720,1:POKEC+30720,7:POKEB,58:POKEC,58-T:POKE7718+T,58-T:NEXT
540 M=PEEK(7650):SC=PEEK(7648)*256+PEEK(7649):C=172:H=1:GOSUB8000
600 FORQ=1T02:IFT(Q)THENOMT(Q)GOTO5000,5010,5020,5040
610 IFRND(1)<SPTHEN(Q)=1
620 P=PEEK(Y(Q)+D(Q))-44:IFP<0THEN670
630 ONPGOTO640,640,640,640,670,670,640,640,670,670,670,670,670,670,8000
640 P=1:IFABS(D(Q))=1THENP=22
650 D(Q)=P*SGN(X-Y(Q)):P=PEEK(Y(Q)+D(Q)):IFP>32ANDP<49THEND(Q)=-D(Q)
660 GOTO690
670 POKEY(Q)+30720,F(Q):POKEY(Q),E(Q):Y(Q)=Y(Q)+D(Q):E(Q)=P+44:F(Q)=PEEK(Y(Q)+30
720)
690 POKEY(Q)+30720,1:POKEY(Q),Z(Q):SP=SP+1.5E-04:IFSP>.05THENSP=.05
1000 P=PEEK(197):IFP=17THENA=-1:GOTO1060
1010 IFF=18THENA=1:GOTO1060
1020 IFF=9THENA=-22:GOTO1060
1030 IFF=32THENA=22:GOTO1060
1040 IFHANDP=63THENH=H-1:GOSUB1600
1050 GOTO1090
1060 POKE32:X=X+A:P=PEEK(X)-44:IFP<0THENPOKE36875,200:POKE36875,0:GOTO1090
1070 ONPGOSUB4000,3000,1100,1100,1110,7500,1400,1400,1300,1300,1300,1300,1300,12
00
1090 POKE32+30720,7:POKE32,59:NEXT:GOTO600
1100 POKE36876,200:X=X-A:POKE36876,0:RETURN
1110 POKE36874,150:C=C-1:SC=SC+1:POKE36874,0:IFCTHEN1230
1120 FORT=130T0250:POKE36874,T:POKE36874,0:NEXT:SC=SC+250:F=1:GOTO1230
1200 FORT=128T0250:POKE36876,T:POKE36876,0:NEXT:M=M-1
1220 POKE32:X=8143:PRINT"SCORE":FORT=1T0M:PRINT";";:NEXT:IFM=0THEN1500
1230 PRINT"SCORE"TAB(5)"SCORE":SC:RETURN
1300 FORT=0T06:POKE36877,130+T*20:NEXT:POKE36877,0
1310 IFF=13-STHENS=S+1:POKE38452-P,7:IFS=5THENSC=SC+225:H=H+1
1320 SC=SC+5:GOTO1230
1400 X=X-A:GOTO8000
1500 PRINT"GAME OVER!"TAB(6)"GAME OVER!":POKE198,0:WAIT198,1:RUN
1600 FORT=250T0130STEP-1:POKE36875,T:POKE36875,0:NEXT:GOTO1220
3000 E=45:T=-2:D=21:B=-23:IFPEEK(X+1)=47THENT=2:D=-21:B=23
3010 GOTO4100
4000 E=46:T=44:D=21:B=23:IFPEEK(X-22)=47THENT=-44:D=-21:B=-23
4100 POKE36876,128:IFPEEK(X+D)>32ORPEEK(X+B)>32THEN1100
4110 POKE32+T,32:POKE32+D+30720,3:POKE32+B+30720,3:POKE32+D,E:POKE32+B,E:POKE36876,0:
RETURN
5000 POKE38696,7:Z(Q)=51:T(Q)=2:GOTO690
5010 POKE36877,130:POKEY(Q)+30720,F(Q):POKEY(Q),E(Q):T(Q)=3:POKE36877,0:GOTO1090
5020 Y(Q)=X-5*A+A*INT(RND(1)*11):P=PEEK(Y(Q)):IFT>2THEN5090
5025 IFF<32ANDP<49ORY(Q)<7748ORY(Q)>8161THEN(T)=T+1:GOTO5020
5030 E(Q)=P:F(Q)=PEEK(Y(Q)+30720):Z(Q)=51:P=ABS(A):T(Q)=4:GOTO650
5040 POKE36874,130:Z(Q)=52:T(Q)=0:POKE38696,6:POKE36874,0:GOTO690
5090 T=0:IFA=1THENA=22:GOTO5020
5095 A=1:GOTO5020
7500 IFF=0THEN1400
7510 FORT=130T0250STEP,2:POKE36875,T:POKE36875,0:NEXT:SC=SC+500:FORT=0T03000:NEX
T
7540 T=INT(SC/256):POKE7648,T:POKE7649,SC-T*256:POKE7650,M+1:PRINT"Q":RUN150
8000 POKE38696,6:POKE7976,50:FORT=1T02:POKEY(T)+30720,F(T):POKEY(T),E(T):Y(T)=79
76
8010 Z(T)=52:F(T)=6:E(T)=50:NEXT:GOSUB1200:GOTO600

```


JOY-MAN

Besteht aus zwei Teilen: Für VC-20 ohne Erweiterung

Der erste Teil wird normal eingetippt und abgespeichert. Danach kann der zweite Teil eingegeben werden.

Nachdem der zweite Teil abgetippt und abgespeichert ist, starten Sie das Hauptprogramm ganz normal mit „RUN“. Es baut sich jetzt ein Zufallswald auf. Pilze sowie Bäume werden mit dem Zufallsgenerator auf dem Bildschirm geprintet.

Ein Monster, das ebenfalls mit dem Zufallsgenerator betrieben wird, soll Ihnen das Leben schwerer machen. Falls Sie den Wald verlassen wollen, müssen Sie zum „HELP“.

Gesteuert wird mit den Tasten! >W< hoch, >X< runter, >D< rechts und >A< links oder mit Joystick!

Mit GOTO 1 erfolgt der Kaltstart des Programmes. Es darf nicht mit !!!RUN!!! gestartet werden, da dann die Variablen „hsc“ und „zS“ gelöscht werden. Aus der Zeile 10 erfolgt ein Sprung zum Aufbau der UDG und zu den Erklärungen.

Display

Das Bild wird mit selbstdef. Grafikzeichen aufgebaut.

Tastaturabfrage

Sie erfolgt über INKEYS. Die Verzweigung zur Steuerung des Männchens wurde mit BOOLSchen Funktionen erreicht. Diese Methode erschien schnell genug. Nach Rücksprung aus den Subroutinen werden eventuell erreichte Punkte errechnet. Die Punkte für die Schlüssel werden über eine ATTR-Abfrage gefunden – ATTR = 51 entspricht einem Schlüssel.

Die unsichtbaren Hausnummern (PAPER-Farbe = INK-Farbe) werden durch SCREENS abgefragt.

Nach Finden aller Schlüssel erfolgt ein Sprung zur Routine in Zeile 3000. Hier wird das Männchen automat. zu seinem Raumschiff geführt. Es startet und fliegt aus dem Bildschirm. Dann erfolgt

der Aufbau des nächsten Bildschirms. 5 solcher Displays müssen durchlaufen werden, um die Bonusrunde zu erreichen.

Steuerung

Die Bewegung des Männchens erfolgt in zwei Stufen:

bei li/re 1 Zeile hoch u. 2 Spalten vor bzw. zurück, dann wieder 1 Zeile tief u. 2 Spalten zurück bzw. vor. Dadurch entsteht der Eindruck eines Hüpfers. Bei hoch/tief springt das Männchen scheinbar erst 2 hoch bzw. 3 tief, dann 3 hoch bzw. 2 tief.

Der Eindruck der Bewegung wird durch Printen des Männchens an der zweiten Position und durch Löschen an seiner ersten erreicht.

Die selbständige Bewegung wurde folgendermaßen realisiert: Nachdem alle Schlüssel gesammelt sind, erfolgt ein Sprung nach 3000. Dort werden die

Positionen der Rakete und des Männchens zueinander berechnet. Daraus läßt sich die Bewegungsrichtung bestimmen; wie hoch und wie weit. Jetzt muß nur noch i\$ mit dem Buchstaben gesetzt werden, der die entsprechende Bewegungsrichtung symbolisiert. Nun erfolgt ein Sprung in die benötigte Bewegungs-Subroutine (1000 o. 2000). Hat das Männchen die Rakete erreicht, zündet sie, hebt ab und fliegt aus dem Bildschirm.

Bonusrunde ist zu erreichen nach 5 Bildschirmen, ohne Verlust des Lebens. Das Männchen schwebt über Pfeilen, zwischen denen das Raumschiff steht. Durch Tastendruck gesteuert, hält es an und sinkt nach unten. Trifft es das Raumschiff, erhält es einen Bonus und das Spiel geht weiter. Trifft es das Raumschiff nicht, verliert es ein Leben. Auch hier startet das Raumschiff und fliegt davon.

Spielende

Im Programmteil ab 8000 werden die Leben gezählt. Sind alle verloren, erklingt ein bekannter Trauermarsch, der Score bzw. neue Highscore wird angezeigt. Dann erfolgt die Anzeige aller 8 besten Spieler, in absteigender Reihenfolge. War der Score größer als der des 8. Spielers, hat man die Möglichkeit, seinen Namen in der Score-Tabelle zu verewigen, sonst erfolgt die Aufforderung, weiterzumachen.

LISTING JOY-MAN TEIL 1

```

10 REM *****
20 REM *ICBY HASEN-*
30 REM *****SOFT*****
40 REM *****
50 POKE51,0:POKE52,28:POKE55,0:POKE56,28:CLR:CB=7168
60 READ:IFA=-1:THEN90
70 FORN=0TO7:READB:POKECB+AN*8+N,B:NEXT
80 GOTO60
90 GOSUB440
100 DATA0,24,24,124,218,218,24,36,102
110 DATA19,66,66,219,66,60,24,36,66
120 DATA1,0,0,0,0,56,84,16
130 DATA2,24,60,60,126,126,255,24,24
140 DATA7,63,72,167,128,225,49,239,195
150 DATA8,3,4,196,35,144,24,56,195
160 DATA9,196,10,17,145,81,81,74,132
170 DATA10,119,66,66,114,66,66,66,66
180 DATA5,151,148,148,247,148,148,148,151
190 DATA6,71,69,69,71,68,68,116
200 DATA3,0,0,0,0,28,54,73,8
210 DATA4,101,85,85,117,87,82,82,98
220 DATA12,0,1,34,34,250,34,33,0
230 DATA11,60,66,153,161,161,153,66,60
240 DATA13,0,136,20,34,34,20,137,0

```



```

60 REM **      **
70 REM *      *
80 REM *      *
90 REM *HASEN-*
100 REM* SOFT *
110 REM**KIEL1**
120 REM*****
130 POKE36869,255:POKE36879,25
140 PRINT"XXXXXXXXXXXXXXXXX U V ← Z [ 1"
150 PRINT"XXXXXXXX X Y ! £ T"
160 PRINT"XXXXXXXXXXXXXXXXXKNOIGHIJLM
170 P=0:P0=0:POKE37139,0:J=1:F1=36877:N9=36878
180 S2=36875:L=36878
190 POKEL,15
200 RESTORE
210 READP
220 IFP=-1THEN270
230 READD
240 POKES2,P
250 FORN=1TOD:NEXTN
260 POKES2,0:FORN=1TO20:NEXTN:GOTO210
270 POKES2,0:GOTO340
280 DATA215,225,207,500,207,225,215,225,219,225,201,500,201,225
290 DATA207,225,209,225,225,225,100,5,225,225,223,225,215,225,219,225,215,225,20
9,225
300 DATA207,500
310 DATA215,225,219,225,219,225,219,225,228,225,225,225,223,225,225,225
320 DATA219,225,215,225,195,225,201,225,207,225,219,225
330 DATA215,500,215,225,209,225,191,225,191,225,195,225,-1
340 D=7712:PRINT"D":POKE36879,25
350 POKE36869,255:P=0:P0=0
360 POKE36879,8
370 PRINT"D":POKE36879,8:FORV=1TO100
380 POKE7680+INT(RND(1)*484),2
390 NEXTV
400 FORDV=1TO100
410 POKE7680+INT(RND(1)*484),16
420 NEXTDV
430 FORT=1TO50
440 POKE7680+INT(RND(1)*484),3
450 NEXTT
460 FORW=1TO35*J
470 POKE7680+INT(RND(1)*484),1
480 NEXTW
490 FORC1=7702TO8164STEP22:POKEC1,16:NEXTC1
500 FORC1=7723TO8185STEP22:POKEC1,16:NEXTC1
510 FORC1=8164TO8185:POKEC1,2:NEXTC1
520 FORC1=8032TO8053:POKEC1,15:NEXTC1
530 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXEF"
540 TI$="000000":D=7712:N2=7915:V5=8033:KL=8064
550 BETA$:POKEB,32:POKEKL,32:IFA$="W"THEND=D-22
560 IFA$="A"THEND=D-1
570 IFA$="X"THEND=D+22
580 IFA$="D"THEND=D+1
590 J9=PEEK(37137):POKE37154,127:J8=PEEK(37152):POKE37154,255:KL=KL+22
600 IF(J9AND4)=0THEND=D-22
610 IF(J9AND8)=0THEND=D+22
620 IF(J9AND16)=0THEND=D-1
630 IF(J8AND128)=0THEND=D+1
640 M2=INT(RND(1)*4)+1:POKEN2,32:IFM2=1THENN1=-22
650 IFM2=2THENN1=1
660 IFM2=3THENN1=-1
670 V5=V5+1

```

```

680 IFM2=4THENM1=22
690 IFPEEK(D)=5ORPEEK(D)=6THENJ=J+1:GOTO370
700 IFV5=8051THENPOKEV5,15:POKEV5+1,15:V5=8032
710 IFHI<PTHENHI=P
720 GOSUB920
730 IFPEEK(D)=3THENPOKEF1,241:POKEN9,15:POKEF1,0:POKEN9,0
740 IFPEEK(N2+N1)=5ORPEEK(N2+N1)=6THENM1=0
750 IFPEEK(N2+N1)=16THENM1=0
760 IFPEEK(N2+N1)=15THENM1=0
770 IFPEEK(N2+N1)=2THENM1=0
780 IFPEEK(N2+N1)=1THENM1=0
790 IFPEEK(N2+N1)=1THENM1=0
800 IFPEEK(N2+N1)=3THENM1=0
810 IFN2=DTHEN990
820 IFKL=DTHEN990
830 IFKL=8152THENKL=8064
840 POKED,0:POKEI+30720,1
850 POKEN2,32:N2=N2+N1:POKEN2,18:N1=0
860 POKEV5,15:POKEV5+2,32:POKEV5+30720,1
870 POKEV5+1,15:POKEV5+1,32:POKEV5+30721,1
880 POKEKL,17:POKEKL+30720,1
890 PRINT"88";P+P0;" ";TI$;J;HI
900 IFVAL(TI$)>150THENGOTO990
910 GOTO 550
920 IFPEEK(D)=3THENP=P+10:RETURN
930 IFPEEK(D)=1THENGOTO990
940 IFPEEK(D)=2THENGOTO990
950 IFPEEK(D)=16THENGOTO990
960 IFD=N2THEN990
970 IFPEEK(D)=15THEN990
980 RETURN
990 POKE36879,159:FORL=1TO500:NEXT
1000 POKE36879,110:FORL=1TO200:NEXT:POKE36879,25
1010 IFHI<PTHENHI=P
1020 IFK<JTHENK=J
1030 POKE36869,240:POKE3687,25
1040 PRINT"*** GAME OVER ***";SCORE:"";P+P0:PRINT"HI SCORE: "
;HI
1050 PRINT"FIELD: ";K
1060 POKE36877,220
1070 FORL=15TO0STEP-1
1080 POKE36878,L
1090 FORM=1TO300
1100 NEXTM
1110 NEXTL
1120 POKE36877,0
1130 POKE36878,0
1140 PRINT" REPLAY (Y/N) ? " :GETA$:IFA$=""THEN1140
1150 IFA$="Y"THENGOTO350
1160 IFA$="N"THENPRINT"J":END
1170 IFA$<>"N"ANDAS$<>"J"THENGOTO1140
1180 END

```

READY.

Buffalo Bill Korrektur

- 1 REM *****
- 2 DATA 143,169,127,141,34,145,173,32,145,41,128,133,251,169,255
- 3 DATA 141,34,145,173,31,145,41,28,24,101,251,133,251,173,31,145
- 4 DATA 41,32,133,252,96
- 5 FORI =4101 TO 4101+35:READ DA:POKE I,DA:NEXTI
- 6 GOTO20000

Nach der Korrektur die Zeile 20010 bitte entfallen lassen!

Ausgabe 11/12 Seite 27 -
Korrektur VC 20

Bücher-Report

Fortsetzung von Seite 7

weicht von der Norm ab. Es ist kein starres gebundenes Buch, sondern es macht mehr einen ringbuchartigen Eindruck. Auch die Seitennumerierung ist anders als bei anderen Büchern, denn Andreas Dripke hat in seinem Buch die Seiten kapitelweise durchnummeriert. Er erklärt auch, warum: Diese Art der Numerierung erlaubt, „für neu erscheinende Computer-Seiten einzufügen etc.“.

Einen kleinen Wermutstropfen gibt es bei diesem Lehrgang auch, aber ein Wermutstropfen, der bei näherer Betrachtung gar keiner ist. Wie Sie sicher wissen (oder zumindest bereits gehört haben), besteht die Assemblerprogrammierung des 6502 darin, daß Befehls Worte aus drei Buchstaben (eventuell mit Argument) in den Computer eingegeben werden. Diese Befehls Wörter werden dann als Zahl vom Computer im Speicher abgelegt. Wenn Sie aber nun „LDA £100“ (das ist ein Assemblerbefehl) in ihren Computer eingeben, so wird der nur mit einem Achselzucken (sprich SYNTAX ERROR) darauf antworten. Assemblersprache kann man also nicht direkt in den Computer eingeben, sondern man benötigt ein Programm, welches die Assemblerbefehle in die entsprechenden Zahlen umwandelt. Solche Programme heißen übrigens auch ASSEMBLER. Das hier besprochene Buch bezieht nun seine Aufgaben auf einen ganz bestimmten Assembler (genannt T.EX.AS = Terminal Extended Assembler), welcher ebenfalls aus dem Hause Dripke stammt. Parallel zur Anschaffung des Buches empfehle ich, daß man sich T.EX.AS anschafft, obwohl der Lehrgang auch mit anderen Assemblern erfolgreich durchzuarbeiten ist. Aber einem Assembler benötigt man früher oder später sowieso und mit T.EX.AS erwirbt man ein absolutes Spitzenprogramm auf diesem Gebiet. Außerdem hat es sicher Vorteile, wenn man bereits beim Erlernen der Assemblersprache mit dem Assembler arbeitet, welchem man auch später bei seiner Programmier Tätigkeit benützt.

Nun aber zu dem Buch selber. Es ist in einer sehr gut dosierten Mischung aus Theorie und Praxis geschrieben, so daß man es am besten am Computersitzend bearbeitet. Die zu lernenden Assemblerbefehle werden in kleinen Programmen vorgestellt und dann deren Wirkung ausprobiert. Zwischen diesen mehr praktisch orientierten Kapiteln sind welche mehr theoretischer Art eingestreut, in welchen von Bits und Bytes erzählt, der Speicher Aufbau erläutert oder die verschiedenen Adressierungsarten in der Assemblersprache besprochen werden. Mit theoretischen Kapiteln meine ich allerdings nicht, daß diese Kapitel trocken dargebracht werden. Auch sie sind, wie das ganze Buch, in einer lockeren, leicht verständlichen, aber doch korrekten Sprache geschrieben. Und hat man wirklich einmal etwas nicht verstanden, so kommt am Ende eines jeden Kapitels noch einmal eine Zusammenfassung, in welcher das Wichtigste der letzten Seiten noch einmal gesagt, neu erlernte Befehle noch einmal aufgelistet sind.

Ein Wort noch zum letzten Kapitel des Buches – Betriebssystemroutinen. Dieses für einen Assemblerprogrammierer äußerst wichtige Kapitel wurde, bedingt durch die Konzeption des Buches, für mehrere Computer geeignet zu sein, nur recht allgemein abgehandelt. Ich persönlich hätte mir da mehr Informationen über meinen Computer gewünscht, ohne mir gleich noch ein Buch anschaffen zu müssen. Ein Wunsch, der vielleicht durch einen computerspezifischen Anhang einmal in Erfüllung gehen könnte?

Zusammenfassend möchte ich sagen, daß dieses Buch durch seine Verständlichkeit das Beste ist, was ich bisher zum Thema „Einführung in die 6502-Assemblersprache“ in der Hand hielt. Trotzdem möchte ich dem völligen Anfänger raten, nicht der Versprechung auf dem Buchrücken zu glauben und gleich mit dieser maschinenorientierten Sprache zu beginnen, sondern sich erst in BASIC hineinzuarbeiten. Denn ein Teil der Verständlichkeit des Buches beruht auf Vergleichen der Assemblerbefehle mit BASIC-Befehlen, was, wenn man BASIC nicht beherrscht, natürlich nicht viel bringt. Der BASIC-Programmierer findet aber mit diesem Buch einen so leichten Einstieg in dieses neue Programmiergebiet, daß ich ihm die Anschaffung nur empfehlen kann. Der Preis für die 146 Seiten ist mit 38,- DM zwar hoch, im Vergleich zu anderen Büchern ähnlichen Inhalts aber wohl angemessen, zumal das Buch auch dann noch als Nachschlagewerk zum Thema „Wie war das noch ...?“ dienen kann, wenn die ersten eigenen Maschinenprogramme schon lange laufen.

6502-Assemblerkurs für Beginner von Andreas Dripke, Interface Age Verlag GmbH, ISBN 3-88986-000-1, 146 Seiten, Preis: ca. 38,- DM.

Fortsetzung von Seite 4

Ein eigenbrötlerischer Erfinder aus Kalifornien namens Walter Spawr ist ein klassisches Beispiel. In seiner Garage entwickelte Spawr ein neuartiges Verfahren zur Glaspolitur, das später zum Schlüssel der neuen Laserwaffentechnologien für den Weltraum wurde. Als bald kauften auch die größten Rüstungskonzerne Amerikas bei ihm ein. Politisch war Spawr ein Anhänger von Ronald Reagan. Er hielt sich für einen rechtschaffenen Staatsbürger. Als er jedoch in finanzielle Bedrängnis geriet, konnte er in ein Ost-West-Geschäft gelockt werden, an dessen Ende er eines der meistgehüteten Geheimnisse der USA an Moskau verkaufte. Seine Laserspiegel wurden später in Semiplatinsk und Sarajaua bei sowjetischen Waffentests verwendet.

Frage: Wie kommen Sie darauf, daß solche Ost-West-Geschäfte einen geheimdienstlichen Hintergrund haben?

Tuck: Es ist allgemein bekannt, daß der damalige KGB-Chef Juri Andropow Anfang der siebziger Jahre eine neuartige Abteilung in dem sowjetischen Spionageapparat unter dem Namen Direktorat „I“ gegründet hat. Heute werden dort 5000 Analytiker mit der Zusammenstellung illegaler Einkaufslisten und der Planung optimaler Schmuggelrouten beschäftigt. Sie sind keine Spione im herkömmlichen Sinn, sondern hochkarätige Naturwissenschaftler, Ingenieure und Computertechniker. Ihre ausschließliche Aufgabe ist jedoch der Diebstahl von NATO-Know-how. Anhand von Anklageschriften, Senatsanhörungen, Interviews und Presseauschnitten läßt sich die Spur ihrer weitverzweigten Operationen zurückverfolgen. Einige KGB-Agenten tarnen sich bei Moskauer Handelsunternehmen, wie MASHPRIBORINTORG oder TECHMASHIMPORT, wo sie Westreisende ansprechen, die sie für Embargoschiebereien anwerben sollen. Dazu gehört etwa Viktor Kedrov, der als Agent des militärischen Geheimdienstes GRU in London aufflog und später für das sowjetische Elektrounternehmen ELORG die Einschleusung ganzer Fabrikanlagen in den Ostblock organisierte. Andere werden näher am Geschehen in den westlichen Vertretungen von Ostblock-Unternehmen stationiert, wie etwa der Mitarbeiter der sowjetischen Handelsmission, Guennadi Batachev, der im letzten Jahr wegen Spionage für das KGB vom Oberlandesgericht Düsseldorf verurteilt wurde. Wo auch immer sie sind,

mit Geld geizen sie nicht. Einige westliche Kaufmänner sind durch ihre Geschäfte mit Sowjetspionen schwerreich geworden.

Frage: Beteiligen sich auch große Konzerne an solchen Geschäften?

Tuck: Selten. Unter den 180 Firmen, die in den schwarzen Listen des US-Handelsministeriums stehen, sind ein paar namhafte deutsche Konzerne aufgeführt. Aber die meisten Firmen in der sogenannten „Export Denial List“ sind kleinere Unternehmen und ...

Frage: ... Sagten Sie, 180? So viele bundesdeutsche Unternehmen stehen in einer US-Schwarzliste?

Tuck: Unternehmen und Privatpersonen, ja. Seit der Verschärfung der amerikanischen Embargobestimmungen im Jahr 1980 ist der illegale Handel mit Spitzentechnologie sprunghaft angestiegen. Die meisten Dealer arbeiten aber, wie gesagt, mit kleineren Firmen.

Frage: Was wird da sonst geschmuggelt?

Tuck: Der bekannteste – wohl auch größte – Schmuggelring, der jemals aufflog, operierte in der Nähe von Hamburg. Geleitet wurde er von dem Jesteburger Kaufmann Richard Müller. Eine Sendung von Müller wurde im vergangenen Herbst am Afrika Kai im Hamburger Freihafen beschlagnahmt. Den Inhalt – unter anderem zwei hochgezüchtete VAX-782/11-Computer der Digital Equipment Corporation – ließ der US-Verteidigungsminister Caspar Weinberger nach Washington einfliegen, wo er ihn der Presse als Beispiel für den sicherheitsgefährdenden Schwarzhandel mit strategischen Computern vorführte. Die brisante Affäre um Müller wurde später sogar im Weißen Haus zur Sprache gebracht.

Frage: Was ist daran so brisant? Der VAX-782 von Digital Equipment ist eigentlich ein gängiger Rechnertyp in der westlichen Wirtschaft.

Tuck: Erst im Zusammenhang mit anderen Hardware- und Software-Sendungen, die Müller gleichzeitig durch Hongkong, Schweden und Südafrika schleuste, ließ sich der Zweck des Einkaufspakets erkennen, das Müller für die Sowjets zusammengestellt hatte. Zusammen mit den VAX-Rechnern reichte das Gerät zur Errichtung einer ganzen Fabrikanlage für moderne Computerchips. Neben Elektronik besorgte Müller auch westliche Ledersessel, Schreibtische, Klimaanlage, Lichtschalter und Feuerlöscher, mit denen das russische Rechenzentrum ausgeschmückt werden sollte. Zwei

Fliesenleger wurden von Hamburg nach Moskau geflogen, um einen antistatischen Fußboden zu verlegen. Hintergrund des Unternehmens war die Herstellung sogenannter „Very-High-Speed-Integrated-Chips“. Diese Superchips oder VHSICs sind der aktuelle Brennpunkt neuester Computertechnologien und Gegenstand eines intensiven Wettlaufs zwischen dem US-Computerriesen IBM und dem japanischen Multi Hitachi. Von wirtschaftlichen Interessen aber einmal abgesehen, haben VHSICs eine Eigenschaft, die für Militärstrategen äußerst interessant ist: sie sind gegen die Strahlen aus einer Atomexplosion widerstandsfähig. Aus eigener Kraft hätte die sowjetische Industrie eine vergleichbare Technologie mit Sicherheit nie zustande gebracht.

Frage: Wo ist dieser Richard Müller heute?

Tuck: Auf der Flucht. US-Behörden sowie die Staatsanwaltschaft in Lübeck suchen ihn per Haftbefehl. Interpol fahndet nach ihm, wohl auch einige westliche Geheimdienste. Nach meinen Informationen hält er sich im Ostblock auf. In Lübeck wird zur Zeit der Prozeß gegen mehrere Mitarbeiter von Müller vorbereitet.

Frage: Richard Müller ist aber nur ein Mann. Ein Einzelfall?

Tuck: Wohl kaum. Auch die acht in meinem Buch beschriebenen Schmuggelfälle sind nur die Spitze eines Eisbergs. Im Herbst 1981 gründete der US-Zoll die Sondertruppe „Operation Exodus“, die sich ausschließlich um Technologie-Delikte kümmert. Seitdem wurden über 2500 Sendungen beschlagnahmt – und die Dunkelziffer ist erheblich. Drehscheibe dieses schwarzen Handels ist die Bundesrepublik Deutschland.

Frage: Welche ernstzunehmende Folgen kann ein Handel mit Computern haben, die zumeist schon in Kaufhäusern für jedermann erhältlich sind? Halten Sie das wirklich für so gravierend?

Tuck: Das Ausmaß des Schadens für die westliche Sicherheit wird erst deutlich, wenn man die Endverbraucher unter die Lupe nimmt. Rund 160 Waffensysteme des Warschauer Pakts enthalten westliche Technologie. In der DDR koordinieren westliche Großrechner die Truppenbewegungen der Volksarmee; in Afghanistan steuern westliche Kleincomputer sowjetische Kampffluger. Kriegsschiffe der Roten Flotte werden in Trockendocks aus Japan und Schweden gewartet. Die ATOLL-Luft-Luft-Rakete,

die im vergangenen Herbst von einem sowjetischen Luftwaffenpiloten abgefeuert wurde und 269 Zivilisten in einem Jumbojet der Korean Airlines tötete, war eine Eins-zu-eins-Kopie der NATO-Sidewinder. Die SS-20-Atomrakete, die sich gegen die Bundesrepublik richtet, ist geradezu vollgestopft mit dem neuesten Know-how der NATO: Sie fahren auf Transportfahrzeugen, die mit Hilfe von IBM-Hardware aus New York und MIT-Soft-



ware aus Massachusetts entworfen wurden; die Raketenspitzen sind mit einem Karbonhitzeschild aus New Jersey beschützt; und sollte der Befehl zum Abschuß jemals kommen, werden ihre Mehrfach-Sprengsätze mit Kreiseln in westliche Ziele gelenkt, die mit Hilfe einer US-Firma in Vermont hergestellt werden. Das halte ich schon für bedenklich.

Frage: Daß die Sowjets westliche Militärtechnologie kopieren, ist aber nicht gerade neu.

Tuck: Der östliche Nachbau von westlichen Waffen hat in der Tat eine lange Tradition. Schon die alten Tupolev TU-4-Fernbomber waren dem US-Weltkriegsveteran B-29 nachempfunden worden. Heute geht es aber nicht mehr um den schlichten Diebstahl von Blaupausen. Schon Mitte der 70er Jahre mußten die Sowjets erkennen,

Bücher-Report

daß ihre einheimische Industrie außerstande ist, die präzisen Herstellungsbedingungen für Computer-Elektronik zu erfüllen. Also änderten sie ihre Beschaffungstaktik. Statt - wie bisher - westliche Prototypen einzuschmuggeln, auseinander zu montieren und in mühsamer Kleinarbeit auf rückständigen Maschinen nachzubauen, beschlossen sie das nötige Herstellungsgerät im Westen gleich mitzubekommen. Trotz der strategischen Brisanz dieser Maschinen werden sie selten als Militärgeheimnis eingestuft. Man mußte nur die Embargobestimmungen der NATO umgehen, die den Export in ein Ostblockland gesetzlich untersagen.

Frage: Die Sowjets verfügen über Forscher und Ingenieure, die zur Weltelite zählen. Wenn man Ihnen glaubt, müßten sie aber in einer hoffnungslosen Abhängigkeit zum Westen stehen. Woraus wollen Sie das schließen?

Tuck: Die Überlegenheit westlicher Elektronik ist unstrittig. Sicherlich verfügt die UdSSR über hervorragende Technologien. Sie kann auch neue Technologien in die Massenproduktion ihrer Waffensysteme überraschend schnell umsetzen. Das sowjetische Problem liegt zwischen Theorie und Technologie. Für die rasanten Entwicklungen auf dem Elektronik-Sektor sind die langfristigen Jahrespläne und schwerfälligen Bürokraten einer Planwirtschaft denkbar schlecht geeignet. Erfindungsgeist und Innovation werden in den Mühlen des Sowjetsystems erstickt. Das zeigen die Entwicklungen der 60er Jahre:

Die Rechner, die seinerzeit in der Computer-Metropole Minsk entstanden, waren begrenzt in der Speicher-

kapazität, rückständig im Software-Bereich und sehr stör anfällig. Die verantwortlichen Werksleiter störte das allerdings kaum. Ihr Erfolg wurde ausschließlich an der Zahl der produzierten Maschinen gemessen. Qualität konnte ihnen egal sein. Als Moskau erkennen mußte, daß seine Bemühungen scheitern würden, entstand ein heftiger Flügelkampf in der Kremlführung über den besten Weg aus der Misere.

Am Ende wurde beschlossen, westliche Erfolgsmodelle einfach nachzubauen. Die zwei großen Computererien, die folgten, waren genaue Kopien der IBM-Serien 360/S und 370/S - vom Betriebssystem über Zusatzgeräte bis hin zur Software. Die Russen taufte ihre Raubrechner RYAD-1 und RYAD-2 und ließen sogar die US-Handbücher wortwörtlich in ihre Landessprache übersetzen und legten sie den eigenen Maschinen bei. Damit sparten sie Milliarden an Forschungs- und Entwicklungskosten. Da die russischen RYAD und die amerikanischen IBM vollkompatibel miteinander waren, konnten sowjetische Computer gegen eingeschmuggelte Originalmaschinen aus USA jederzeit ausgetauscht werden, wenn besonders hohe Zuverlässigkeit verlangt wurde. Dies ist nach Pentagon-Informationen beispielsweise im Luftabwehr-Gürtel um die Moskauer Hauptstadt auch geschehen.

Frage: Ist es nicht legitim, wenn der Westen die UdSSR beim Aufbau ihrer Zivilwirtschaft in diesem wichtigen Bereich unterstützt? Oder plädieren Sie etwa dafür, daß man ihnen den Zugang zu sämtlicher Elektronik verbietet, nur weil das eine oder andere

Gerät eventuell beim Militär landen könnte?

Tuck: Sicherlich sind die Sowjets auch im zivilen Bereich an Computern interessiert. Ihre Planwirtschaft macht sie zu einem geradezu idealen Verbraucher für EDV-Systeme. Man darf aber nicht vergessen, daß dieselben Computer-Chips, die wir in Quarzuhren, Videospiele oder Schreibmaschinen einbauen, ebenfalls das Radar eines MIG-23-Kampffjägers, das Lenksystem einer Atoll-Abwehrrakete oder die atomaren Sprengköpfe der SS-20 steuern können. Gängige westliche Elektronik, die bei uns bereits in Billigprodukten verwendet wird, ist im Ostblock nach wie vor eine Rarität. Daß die sowjetischen Prioritäten nicht in erster Linie Quarzuhren und Videorecorder für den Verbraucher vorsehen, muß hier wohl nicht weiter erläutert werden.

Frage: Sehr entscheidend kann dieser Handel letztlich nicht sein. Wenn einer den anderen kopiert, hinkt er im Wettlauf zwangsläufig hinterher.

Tuck: Nicht zwangsläufig. Der Westen ist zwar in der Entwicklung neuer Computertechnologien sehr schnell, aber erheblich langsamer als der Warschauer Pakt, wenn es um die Dislokierung neuer Waffen geht. Dies wurde Pentagon-Experten deutlich, die den Nachbau eines amerikanischen Computerchips in einer sowjetischen Sonarboje entdeckten. Das US-Original befand sich noch in der Erprobung. Russische Rüstungsplaner hatten die Technologie gestohlen, nachgebaut und disloziert, ehe die Washingtoner Bürokratie eine Entscheidung über die Verwendung im Westen fällen konnte. Seit Kriegsende verläßt sich die NATO-Verteidigung auf die Überlegenheit von westlichem Wissen - auf die Vorstellung, daß Maschinen und Menschenmassen durch immer perfekter werdende Technologien ersetzt werden können. In den nächsten Jahren wird diese Tendenz weiter zunehmen. Die sogenannten „denkenden Waffen“ der 80er Jahre bilden eine wesentliche Stütze der künftigen NATO-Strategie. Es gibt ernstzunehmende Überlegungen, die in Europa stationierten atomaren Gefechtsfeldwaffen mit modernen konventionellen Waffen gänzlich zu ersetzen - ohne Verlust an Verteidigungskraft. Marschall Orgakow, der vor wenigen Wochen als Chefstrateg der Warschauer Pakts abgesetzt wurde, hat die strategische Bedeutung derlei Entwicklungen durchaus erkannt.

NEU Der NEWMAN Beratungs-Katalog

Rund 1.000 Angebote. Alles von COMMODORE, Sinclair, Dragon, Sharp, Spectravideo und anderen mit Original-Werks-Garantie. SOFORT LIEFERBAR. Ob Hardware, Peripherie, Bücher, Programme oder Zubehör, Sie erhalten alles aus einer Hand.

Sofort GRATIS anfordern

Teilzahlung, technischer Service, BERATUNG.

Gutschein für 1 Katalog

Ausfüllen, ausschneiden, auf Postkarte kleben MU 11 und absenden.

Name/Vorname _____

Straße/Nr. _____

PLZ/Ort _____

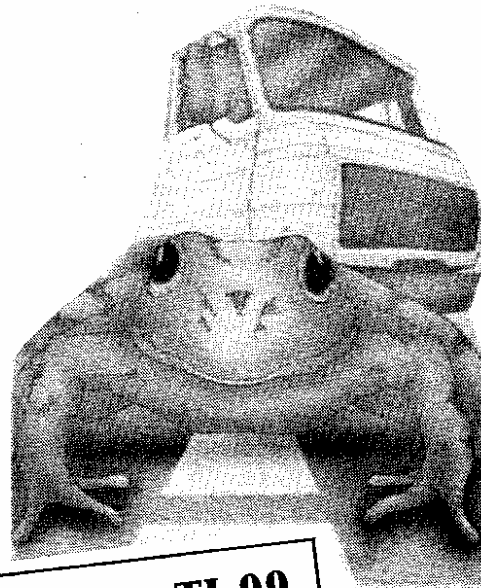
100 Seiten dick

NEWMAN Computer-Versand Postfach 501126, 2000 Hamburg 50, Tel. 040/850 60 71

ALLES FÜR IHRER HOME-COMPUTER

Topprogramm

Frogger



Während des Spielverlaufes müssen Sie versuchen, alle 10 Frösche, die Ihnen zur Verfügung stehen, in ihren Unterschlupf zu bringen.

Zunächst muß jeder Frosch eine vielbefahrene Straße überqueren. Auf dem Mittelweg angekommen, kann er sich erst einmal ausruhen und die Flußüberquerung vorbereiten. – Ab Level 6 wird die Ruhepause auf dem Mittelweg durch eine sich hin- und herbewegende Schlange erschwert. – Auf die andere Seite des Flusses kann der Frosch nur gelangen, wenn er ab-

jetzt auch für den TI-99
mit Ext. Basic-Modul

wärtstreibende Schildkröten und Baumstämme zu Hilfe nimmt. Auf ihnen kann er sich, ohne ins Wasser zu müssen, bewegen.

Ist der Fluß erfolgreich genommen worden, steht das nächste Hindernis bevor. Eine Start- und Landebahn für Flugzeuge muß gekreuzt werden. Aber auch das dürfte für die „echten Könner“ kein Problem sein. Sind alle 10 Frösche gut untergebracht, erfahren Sie über den Score, wieviel Zeit dafür benötigt wurde. Das Spiel ist in 99 Levels unterteilt. 1–5 ist für Anfänger, 6–20 für Amateure und 21–99 für Profis. Gesteuert wird mit Joystick.

```
100 ! #####
110 ! # #
120 ! # SPECIAL - FROGGER #
130 ! # #
140 ! #A GAME FOR 1 PLAYER#
150 ! # #
160 ! # (C) BY KABOSO #
170 ! # KARSTEN BOGUSCH #
180 ! # SOFTWARE #
190 ! # #
200 ! # #
210 ! # D-4600 DORTMUND 41#
220 ! # #
230 ! # #
240 ! #####
250 !
260 !
270 !
```

```
280 ON BREAK NEXT :: ON WARNING NEXT :: CALL SCREEN(2):: CALL CLEAR :: CALL MAGN
IFY(3):: RANDOMIZE :: DIM J(14)
290 FOR I=0 TO 8 :: CALL COLOR(I,4,2):: NEXT I :: CALL COLOR(2,15,1,3,16,2,4,16,
2,9,4,13,10,6,6,11,8,8,14,16,1)
300 DISPLAY AT(12,1):"ICH DEFINIERE MEINE ZEICHEN!"
310 FOR I=65 TO 90 :: CALL SOUND(-4200,110+I*2,0,112+I*2,0,114+I*2,0):: READ A#
:: CALL CHAR(I,A#):: NEXT I :: GOSUB 540
320 DATA 003F666C746466EF,007E313B3633337E,003E634140603F1E,007E321119113F7E,007
F33303C38317F,007F723239383078
330 DATA 003E6648504F663E,007C603F3132327A03,007C38181C1A183C2,003F13060606263F,
007B72363C363673,007830303072637F
340 DATA 007F2B2A49494963,007C76323A323477,003E634559617F3E,007E33313F3E3078,003
E634559617F3E07,007F313A34323179,003E413C06634F3E
350 DATA 007F492A0808081C,006F2626666E7E37,0073622232321E0C,00772222414936,00793
```



```

710 R=0 :: CALL SOUND(100,210,1,230,1,330,1,-2,1):: IF M=0 THEN 1330 ELSE IF M<1
0 THEN CALL HCHAR(24,9+M,141)
720 ON ERROR 710
730 L=161 :: O=INT(RND*160)+40
740 CALL SPRITE(#16,136,16,L,0)
750 CALL JOYST(2,Y,X):: T=T+1 :: CALL HCHAR(23,30-INT(T/5),32):: IF T=100 THEN 1
090
760 IF X=4 THEN L=L-8 ELSE IF Y=4 THEN O=O+8 ELSE IF Y=-4 THEN O=O-8 ELSE IF X=-
4 AND L>160 THEN 750 ELSE 780
770 GOTO 820
780 IF X=-4 THEN L=L+8 :: GOTO 820
790 GOTO 800
800 CALL COINC(ALL,B):: FC=FC+1 :: IF B=-1 THEN 1090 ELSE IF FC=16 THEN 810 ELSE
750
810 FC=0 :: H=-1*H :: GF=-H :: CALL MOTION(#28,0,H*(INT(RND*15)+5)):: CALL MOTIO
N(#27,0,GF*(INT(RND*27)+6)):: GOTO 750
820 IF L<89 THEN 890 ELSE CALL LOCATE(#16,L,0):: W=W+1 :: CALL SOUND(-5,110,8,10
00,5,-2,5):: R=R+SGN(X):: CALL COINC(ALL,B):: IF B=-1 THEN 1090 ELSE 750
830 IF L=89 THEN CALL MOTION(#16,0,0):: GOTO 750
840 CALL JOYST(2,X,Y):: T=T+1 :: CALL HCHAR(23,30-INT(T/5),32):: IF T=100 THEN 1
090
850 IF Y=4 THEN 890 ELSE IF Y=-4 THEN 860 ELSE IF X=-4 THEN 920 ELSE IF X=4 THEN
940 ELSE 830
860 CALL POSITION(#16,L,0):: R=R-1
870 L=L+8 :: CALL LOCATE(#16,L,0):: W=W+5 :: CALL MOTION(#16,0,J(R)):: CALL SOUN
D(-5,110,8,1000,5,-2,5):: CALL COINC(ALL,B):: IF B=-1 THEN 830 ELSE 880
880 IF L=89 THEN 830 ELSE 1090
890 CALL POSITION(#16,L,0):: R=R+1
900 L=L-8 :: CALL LOCATE(#16,L,0):: W=W+5 :: CALL MOTION(#16,0,J(R)):: CALL SOUN
D(-5,110,8,1000,5,-2,5):: CALL COINC(ALL,B):: IF B=-1 THEN 830 ELSE 910
910 IF L=49 THEN 950 ELSE 1090
920 CALL POSITION(#16,L,0):: O=O-8
930 CALL LOCATE(#16,L,0):: W=W+1 :: CALL SOUND(-5,110,8,1000,5,-2,5):: CALL COIN
C(ALL,B):: IF B=0 THEN 1090 ELSE 830
940 CALL POSITION(#16,L,0):: O=O+8 :: W=W+5 :: CALL LOCATE(#16,L,0):: CALL SOUND
(-5,110,8,1000,5,-2,5):: CALL COINC(ALL,B):: IF B=0 THEN 1090 ELSE 830
950 CALL MOTION(#16,0,0):: CALL LOCATE(#16,INT(L/8)*8+1,INT(O/8)*8+8):: W=W+5 ::
L=INT(L/8)*8+1 :: O=INT(O/8)*8+8
960 !
970 CALL JOYST(2,X,Y):: T=T+1 :: CALL HCHAR(23,30-INT(T/5),32):: IF T=100 THEN 1
090
980 IF L=33 AND Y=4 THEN GOSUB 1030 ELSE IF Y=-4 AND L=49 THEN 840
990 IF Y=4 THEN L=L-8 ELSE IF X=4 THEN O=O+8 ELSE IF X=-4 THEN O=O-8 ELSE IF Y=-
4 THEN L=L+8 ELSE 1010
1000 GOTO 1070
1010 CALL COINC(ALL,B):: IF B=-1 THEN 1090
1020 GOTO 970
1030 CALL GCHAR(4,INT(O/8)+1,0):: IF D<>32 THEN 1090
1040 PU=PU+200 :: PP=PU+W :: DISPLAY AT(2,8)SIZE(-5):STR$(PP):: IF PU+W>HI THEN
DISPLAY AT(2,24)SIZE(-5):STR$(PP)
1050 CALL POSITION(#16,L,0):: CALL DELSPRITE(#16):: CALL HCHAR(L/8,(O/8)+1,140)::
GOTO 1140
1060 RETURN
1070 CALL LOCATE(#16,L,0):: W=W+1 :: CALL SOUND(-5,110,8,1000,5,-2,5):: CALL COI
NC(ALL,B):: IF B=-1 THEN 1090 ELSE 960
1080 !
1090 CALL DELSPRITE(#16):: GOSUB 1360 :: M=M-1 :: PP=PU+W :: DISPLAY AT(2,8)SIZE
(-5):STR$(PP):: T=0
1100 IF PU+W>HI THEN DISPLAY AT(2,24)SIZE(-5):STR$(PP)
1110 IF M=0 THEN 1330
1120 CALL HCHAR(23,10,89,20):: FOR X=1 TO 20 :: NEXT X :: GOTO 710
1130 !
1140 FOR Z=1 TO 10 :: CALL SOUND(-90,4000,4,3900,6):: CALL SOUND(-40,3000,2,2900

```



```

,4):: NEXT Z
1150 Q=Q+1 :: IF Q=5 THEN 1170 :: T=0
1160 CALL HCHAR(23,10,89,20):: GOTD 710
1170 DISPLAY AT(24,2)SIZE(-5):"FROGS:" :: CALL HCHAR(24,10,136,9)
1180 !FOR X=1 TO 4 :: CALL SOUND(120,450,2,460,3):: CALL SOUND(120,780,3,790,3):
: CALL SOUND(120,670,3,680,2):: NEXT X
1190 FOR S=500 TO 110 STEP -50 :: CALL SOUND(1,5*S,0):: NEXT S :: FOR S=110 TO 5
00 STEP 50 :: CALL SOUND(1,3*S,0):: NEXT S
1200 M=10 :: Q,T=0 :: CALL HCHAR(23,10,89,20)
1210 V=V+1 :: IF V>40 THEN V=40
1220 DISPLAY AT(24,27)SIZE(-2):STR$(V+1):: CALL SOUND(390,780,5,790,6,770,7,-1,0
)
1230 FOR I=4 TO 28 STEP 6 :: CALL HCHAR(4,I,32):: NEXT I :: FOR X=1 TO 3 :: CALL
MOTION(#X,0,-(V+7)):: NEXT X :: FOR X=4 TO 8 :: CALL MOTION(#X,0,V+5):: NEXT X
1240 FOR X=24 TO 26 :: CALL MOTION(#X,0,-(V+8)):: NEXT X
1250 FOR X=10 TO 12 :: CALL MOTION(#X,0,V+9):: NEXT X
1260 CALL MOTION(#15,0,-(V+10),#22,0,-(V+10))
1270 J(10)=- (V+5):: J(11)=- (V+5):: J(12)=V+7 :: J(13)=V+7
1280 FOR X=20 TO 21 :: CALL MOTION(#X,0,J(10)):: NEXT X :: FOR X=17 TO 19 :: CAL
L MOTION(#X,0,J(12)):: NEXT X
1290 IF V<5 THEN 1320 ELSE H=INT(RND*2):: IF H=0 THEN H=-1
1300 CALL SPRITE(#28,36,15,89,200,0,H*(INT(RND*22)+5))
1310 IF V<7 THEN 1320 ELSE GF=-H :: CALL SPRITE(#27,36,15,49,50,0,GF*(INT(RND*28
)+5))
1320 GOTD 710
1330 HI=MAX((PU+W),HI):: DISPLAY AT(23,2)BEEP:"ddd NOCH EIN SPIEL? J/N ddd":"ddd
dddddddddddddddddddddddd"
1340 CALL KEY(0,K,S):: IF S=0 THEN 1340
1350 IF K=74 OR K=106 THEN 400 ELSE IF K=78 OR K=110 THEN CALL CLEAR :: BREAK EL
SE 1340
1360 CALL SOUND(250,184,8,369,3,739,10):: CALL SOUND(125,195,8,391,3,783,10)
1370 CALL SOUND(125,184,8,369,3,738,10):: CALL SOUND(250,146,8,293,3,587,10)
1380 CALL SOUND(250,138,8,277,3,554,10):: CALL SOUND(1000,123,8,246,3,493,10)::
RETURN

```

MOON RACE

**Abenteuer
auf dem
Mond**

Alarm auf der Mondbasis TI 99/4 A. In der von Ihnen zu überwachenden Sektion Beta ist ein Aufstand der Mondbewesen ausgebrochen. Starten Sie mit Ihrem Moon Racer zu einer abenteuerlichen Fahrt über die Mondoberfläche und zerschlagen Sie den Aufstand der Mondbewohner.



Das Spiel ist auf dem TI 99/4 A in Extended Basic geschrieben und wird mittels Joystick 1 gesteuert.

Die einzigartige Möglichkeit des TI 99/4 A, bis zu 28 Sprites vom Basic ansprechen zu können, erlaubt es, bei diesem Spiel drei verschiedene Grafikebenen in verschiedenen Geschwindigkeiten derart zu verschieben, daß das Bild plastisch wirkt. Dieser Effekt ist besonders beim Einsatz eines Farbfernsehers zu beobachten. Durch diese Maßnahme mußte jedoch der Vergrößerungsfaktor 4 für die Sprites gewählt werden, wodurch die Grafik eine etwas größere Auflösung zeigt. Dies tut jedoch dem Reiz des Spieles keinen Abbruch.

Tippen Sie das Programm ein und starten Sie dieses mit RUN. Nach kurzer Initialisierungsphase erscheint das Titelbild auf dem Bildschirm. Die Titelmelodie erklingt und die beiden hinteren Grafikebenen setzen sich in Bewegung. Diese Bewegung wird während des ganzen Spieles unaufhörlich fortgeführt.

Sie haben jetzt die Aufgabe, mit Ihrem Moon Racer die Sektion Beta zu patrouillieren. Dazu steht Ihnen eine Anzeige von mehreren Informationen zur Verfügung: Am oberen Bildschirmrand ist links hinter den Schriftzug >Zone< die Patrouillenzone eingeblendet, in welcher Sie sich befinden. Die Sektion Beta besteht aus 4 Zonen.

In der Mitte sehen Sie die Ihnen zur Verfügung stehenden Racer (anfänglich 5). Daneben wird hinter dem Schriftzug >Score< Ihr aktueller Punktestand angezeigt. Am unteren linken Bildschirmrand ist der Level eingeblendet, in welchem Sie sich befinden. Da Sie nach jedem Unglück, das Ihnen widerfährt, bei der Mondbasis neu mit der Patrouille beginnen müssen, erhöht sich der Level immer dann, wenn es Ihnen gelungen sein sollte, mit einem Fahrzeug sämtliche 4 Zonen der Sektion Beta zu durchfahren bzw. zu durchfliegen.

Auf dem Titelbild sehen Sie die Mondbasis TI 99/4 A. Beginnen Sie das Spiel, indem Sie den Aktionsknopf des Joystick 1 drücken. An Stelle der Mondbasis erscheint nun Ihr Moon Racer, und die vordere Grafikebene setzt sich in Bewegung. Sie befinden sich jetzt in ...

Zone 1 ist die harmloseste Zone der Patrouille. Sie fahren mit Ihrem Moon Racer einfach so vor sich hin und müssen die Ihnen in den Weg kommenden Krater und Geröllhaufen überspringen. Zum Springen drücken Sie den Joystick einfach nach unten.

Achten Sie jedoch darauf, daß Sie die Krater und Geröllhaufen exakt überspringen, da Sie sonst einen Racer verlieren und mit einem neuen Racer bei der Basis beginnen müssen.

Für jeden erfolgreich übersprungenen Geröllhaufen erhalten Sie 10 Punkte, welche mit dem aktuellen Level multipliziert werden.

Irgendwann im Spiel bleibt die vordere Grafikebene stehen, und es erklingt ein Tusch. Jetzt haben Sie Zone 1 bewältigt und gelangen sofort danach in ...

In **Zone 2** wimmelt es nur so von Mondechsen, welche Ihnen im Weg sitzen. Da die Mondechsen jedoch so groß sind, daß Sie mit Ihrem Racer nicht darüber hinwegspringen können, so müssen Sie diese Echsen eliminieren. Dazu drücken Sie den Aktionsknopf am Joystick. Sofort schleudern Sie damit einen Torpedo in Richtung Mondechse. Ihr Torpedo hat jedoch nur eine bestimmte Reichweite, so daß Sie die Mondechse erst aus einer bestimmten Entfernung vernichten können. Haben Sie die Echse getroffen, so werden Ihnen 5 Punkte gutgeschrieben, und die Echse verwandelt sich in einen Geröllhaufen, den Sie überspringen müssen, um weitere 10 Punkte zu erhalten. Auch in Zone 2 werden die Punkte mit dem augenblicklichen Level multipliziert. Nachdem Sie Zone 2 erfolgreich überstanden haben sollten, ertönt wieder der Tusch, die vordere Grafikebene bleibt stehen, und Sie gelangen in ...

In **Zone 3** werden Sie von fliegenden Mondkrabben angegriffen. Diese Mondkrabben müssen Sie vernichten, da sonst Sie vernichtet werden. Weiterhin ist Zone 3 ein ziemlich unwegiges Gelände, wo Sie laufend über Krater und Geröllhaufen springen müssen. Es kommt noch hinzu, daß Zone 3 die längste Strecke auf Ihrer Patrouille ist und schier nicht enden möchte. Mondkrabben können Sie nur im Sprung vernichten, indem Sie den Joystick nach unten bewegen und während des Sprunges den Aktionsknopf drücken, um einen Torpedo abzufeuern. Als Schwierigkeit kommt noch dazu, daß Sie bei jedem Sprung nur einen Schuß abfeuern können. Wenn dieser Schuß danebengeht, so sind Sie der Krabbe hoffnungslos ausgeliefert. Treffen Sie jedoch hingegen, so werden Ihnen 25mal Level-Punkte gutgeschrieben. Sollte es Ihnen wider Erwarten gelingen, Zone 3 zu überleben, so ertönt der bekannte Tusch, und die Grafik bleibt für kurze Zeit stehen.

In den ersten 3 Zonen bekommen Sie nach einem erfolgreichen Beenden der Zone 100mal Zone mal Level-Bonuspunkte gutgeschrieben.

Wenn Sie es bis hierher geschafft haben sollten, so gelangen Sie in ...

Zone 4 ist radioaktiv verseucht, so daß Sie mit Ihrem Racer diese Zone nicht durchfahren können. Aus diesem Grund verwandelt sich Ihr Racer kurzum in einen Moon Jet, mit wel-

chem Sie Zone 4 durchfliegen müssen. Zone 4 teilt sich in 4 Sektoren auf: In Sektor 1 kommen Ihnen radioaktive Wolken entgegen, denen es auszuweichen gilt.

Dazu bewegen Sie Ihren Jet mit dem Joystick nach oben und unten. Passen Sie jedoch auf, daß Sie am Boden keine Bruchlandung machen und daß Sie nicht mit dem Gebirge kollidieren.

In den Sektoren 2, 3 und 4 kommen Ihnen Strahlen entgegen, welche von Sektor zu Sektor mehr werden. Wie bei Sektor 1 gilt es auch hier, den Strahlen auszuweichen.

Pro bestrittenem Sektor werden Ihnen 500mal Sektor mal Level-Punkte gutgeschrieben.

Sollte es Ihnen wider Erwarten dennoch gelingen, auch Zone 4 zu meistern, so hören Sie wieder Ihren Tusch und kehren zur Basis zurück. Sie sehen die Basis, und die Titelmelodie erklingt. Durch Druck auf den Aktionsknopf starten Sie den nächsten Level, welcher sich in der Länge der einzelnen Zonen vom vorherigen Level unterscheidet.

Mit jedem Verlust eines Racers bzw. eines Jets kehren Sie wieder zur Basis zurück, von wo aus Sie durch Druck auf den Aktionsknopf eine neue Patrouille beim gleichen Level beginnen können.

Nachdem alle 5 Fahrzeuge zerstört wurden, erscheint wieder das Titelbild. Sie können jetzt mittels des Aktionsknopfes ein neues Spiel starten. Moon Race ist ein äußerst interessantes Spiel, welches äußerste Konzentration erfordert und sehr lange Spielfreude beschert.

Das Spiel ist sehr schwer zu beherrschen, und es bedarf einiger Stunden (Tage, Wochen, Monate), bis es einem gelingt, einen kompletten Level durchzuspielen. Viele Spieler werden bereits an Zone 3 verzweifeln. Es sei hierzu jedoch bemerkt, daß es zu schaffen ist. Spätestens bei Zone 4 werden die meisten Spieler sagen, es gibt keine Möglichkeit, durchzukommen. Hierzu sei jedoch gesagt, daß es einen Weg gibt, aber nur einen. Diesen Weg zu finden, sei jedoch dem Spiel-freak überlassen.

Mir bleibt hierzu nur zu bemerken, daß ich allen Spielern viel Spaß und viel Erfolg wünsche bei MOON RACE.

Anmerkung: Die <ALPHA LOCK>-Taste darf während des Spieles nicht gedrückt sein, da es sonst keinen Weg gibt, die Patrouille erfolgreich durchzuführen.


```

690 IF SW THEN RETURN
700 SW=1 :: CALL SPRITE(#3,80,2,105,81,0,120):: CALL SOUND(-50,110,0,-7,0):: CAL
L DISTANCE(#1,#4,WT):: IF WT>11000 THEN 720
710 CALL DELSPRITE(#4):: CALL DELSPRITE(#3):: STV=0 :: P=P+25*MR :: CALL SCORE(P
)
720 CALL DELSPRITE(#3):: RETURN
730 CALL PATTERN(#1,44):: CALL SOUND(-50,-5,0)
740 CALL DELSPRITE(#2):: CALL SOUND(-50,-6,0)
750 CALL COLOR(#1,11):: CALL SOUND(-20,-7,0)
760 CALL DELSPRITE(#1)
770 RC=RC-1 :: CALL RACER(RC)
780 CALL PATTERN(#11,120):: CALL COLOR(#11,14):: STV=0 :: CALL DELSPRITE(#4)
790 IF RC>0 THEN GOSUB 1180 :: CALL WAIT(B(,)):: GOTO 450
800 GOSUB 1180 :: CALL SPRITE(#11,120,14,137,255,0,0,#10,124,4,153,121,0,0)
810 CALL TITEL(B(,),1)
820 RC=5 :: MR=1 :: P=0 :: CALL SCORE(P):: CALL RACER(RC):: CALL ZONE(1)
830 GOTO 450
840 CALL DISTANCE(#1,#11,WT):: IF WT<9000 THEN RETURN
850 CALL PATTERN(#11,84):: CALL COLOR(#11,5):: STV=1 :: RETURN
860 RETURN
870 CALL SPRITE(#4,72,13,113,240,0,-18):: STV=1 :: RETURN
880 CALL PATTERN(#4,CH(LOOP+2)):: RETURN
890 CALL KEY(1,T,S):: IF T=18 THEN 900 ELSE RETURN
900 CALL SPRITE(#3,80,2,121,81,0,120):: CALL SOUND(-50,110,0,-7,0):: CALL DISTAN
CE(#1,#11,WT):: IF WT>9000 THEN 920
910 CALL PATTERN(#11,120):: CALL COLOR(#11,14):: CALL DELSPRITE(#3):: P=P+5*MR :
: CALL SCORE(P)
920 CALL DELSPRITE(#3):: STV=0 :: RETURN
930 GOSUB 1250 :: CALL CHAR(60,CH2$):: CALL ZONE(4)
940 FOR LEVEL=1 TO 4
950 ON LEVEL GOSUB 1050,1060,1070,1080
960 FOR DELAY=1 TO 30+10*MR
970 FOR LOOP=5 TO 6
980 CALL SOUND(-300,-7,0):: CALL PATTERN(#1,CH(LOOP))
990 CALL JOYST(1,X,Y):: CALL MOTION(#1,-2*Y,0)
1000 CALL COINC(ALL,ST):: IF ST THEN 1090
1010 CALL POSITION(#1,X,Y):: IF X>150 THEN 1090
1020 NEXT LOOP :: NEXT DELAY :: P=P+500*LEVEL :: CALL SCORE(P):: NEXT LEVEL
1030 FOR I=1 TO 4 :: CALL DELSPRITE(I):: NEXT I :: CALL CHAR(60,CH1$):: CALL EN
DZONE :: GOSUB 1180 :: CALL WAIT(B(,))
1040 MR=MR+1 :: GOTO 450
1050 CALL SPRITE(#2,44,15,107,240,0,-18):: RETURN
1060 CALL PATTERN(#2,60):: RETURN
1070 CALL SPRITE(#3,60,9,130,240,0,-18):: RETURN
1080 CALL SPRITE(#4,60,8,117,240,0,-18):: RETURN
1090 CALL PATTERN(#1,44):: CALL SOUND(-50,-5,0)
1100 CALL DELSPRITE(#2):: CALL SOUND(-50,-6,0)
1110 CALL COLOR(#1,16):: CALL SOUND(-20,-7,0)
1120 CALL DELSPRITE(#1)
1130 RC=RC-1 :: CALL RACER(RC)
1140 CALL PATTERN(#11,120):: CALL COLOR(#11,14):: STV=0 :: CALL DELSPRITE(#3,#4)
1150 IF RC>0 THEN CALL CHAR(60,CH1$):: GOSUB 1180 :: CALL WAIT(B(,)):: GOTO 450
1160 GOSUB 1180 :: CALL SPRITE(#11,120,14,137,255,0,0,#10,124,4,153,121,0,0)
1170 CALL CHAR(60,CH1$):: GOTO 810
1180 CALL COLOR(9,5,1,10,4,1):: CALL COLOR(#25,5,#26,5,#27,5,#28,5):: CALL COLOR
(#13,5,#14,5,#15,5,#16,5)
1190 CALL DELSPRITE(#4):: CALL PATTERN(#11,120):: CALL COLOR(#11,14,#17,4,#18,4,
#19,4,#20,4)
1200 CALL SPRITE(#11,120,14,137,255,0,-15,#10,124,4,153,121,0,-15):: RETURN
1210 CALL COLOR(9,7,1,10,4,1):: CALL COLOR(#25,7,#26,7,#27,7,#28,7):: CALL COLOR
(#13,7,#14,7,#15,7,#16,7)
1220 CALL DELSPRITE(#10):: CALL SPRITE(#11,120,14,137,255,0,-15):: RETURN
1230 CALL COLOR(9,7,1,10,8,1):: CALL COLOR(#17,8,#18,8,#19,8,#20,8)

```

```

1240 CALL SPRITE(#11,120,14,137,255,0,-15,#10,124,8,153,129,0,-15):: STV=0 :: RE
TURN
1250 CALL COLOR(9,16,1,10,5,1):: CALL COLOR(#25,16,#26,16,#27,16,#28,16):: CALL
COLOR(#13,16,#14,16,#15,16,#16,16)
1260 CALL DELSPRITE(#4):: CALL DELSPRITE(#2):: CALL PATTERN(#11,120):: CALL COLO
R(#11,14,#17,5,#18,5,#19,5,#20,5)
1270 CALL SPRITE(#11,120,14,137,255,0,-15,#10,120,14,137,121,0,-15,#1,64,11,114,
49):: RETURN
1280 SUB MOONRIVER(B(),),KY)
1290 DATA 1,1,6,783,30000,1,2,2,987,1174,4,880,1046,1,3,3,783,987,1,698,880,1,65
9,783
1300 DATA 1,587,698,1,1,4,659,783,2,523,30000,1,3,3,783,987,1,698,880,1,659,783,
1,587,698
1310 DATA 1,1,4,659,783,2,523,30000,2,4,10,587,30000,2,659,30000,1,2,6,523,30000
1320 DATA 1,1,4,659,30000,2,783,30000,1,2,6,1046,30000,1,3,4,1174,30000,2,1046,3
0000,1,1,6,783,30000
1330 DATA 1,3,2,783,30000,1,783,987,1,698,880,1,659,783,1,587,698,1,1,6,659,783,
1,3,1,659,783
1340 DATA 1,523,30000,1,783,987,1,698,880,1,654,783,1,587,698,1,1,6,659,783,1,2,
6,440,523
1350 DATA 1,4,2,440,698,4,349,587,1,6,4,349,587,2,391,659,2,1,12,329,523,0
1360 RESTORE 1290
1370 AZ,STV=0
1380 READ K :: IF K=0 THEN 1360
1390 READ INR
1400 FOR I=1 TO K
1410 FOR J=1 TO 6
1420 STV=STV+1 :: IF STV>AZ THEN READ AZ,M1,M2 :: STV=1
1430 CALL SOUND(400,M1,0,M2,0,B(J,INR),6):: CALL KEY(1,T,S):: IF S<>0 THEN SUBEX
IT
1440 NEXT J :: NEXT I
1450 GOTO 1380
1460 SUBEND
1470 SUB ENDZONE
1480 DATA 300,220,174,300,246,195,1200,261,220,600,293,246,300,329,261,300,349,2
93
1490 DATA 1200,391,329,300,329,261,300,349,293,600,391,329,600,440,349,600,493,2
93,1200,523,329
1500 CALL MOTION(#1,0,0,#2,0,0,#4,0,0,#3,0,0,#10,0,0,#11,0,0):: RESTORE 1480
1510 FOR I=1 TO 13 :: READ L,M1,M2 :: CALL SOUND(L,M1,0,M2,0):: NEXT I
1520 SUBEND
1530 SUB TITEL(B(),),KY)
1540 CALL SPRITE(#1,92,16,113,96,#2,100,16,113,128,#3,92,2,111,98,#4,100,2,111,1
30,#5,60,15,137,45)
1550 CALL MOONRIVER(B(),),KY)
1560 FOR I=1 TO 4 :: CALL DELSPRITE(#I):: NEXT I
1570 SUBEND
1580 SUB SCORE(P)
1590 DISPLAY AT(1,19):USING "mno#####":P
1600 SUBEND
1610 SUB ZONE(Z)
1620 DISPLAY AT(1,1)SIZE(5):USING "abc #":Z
1630 SUBEND
1640 SUB RACER(RC)
1650 CALL HCHAR(1,12,32,5)
1660 IF RC=0 THEN SUBEXIT
1670 CALL HCHAR(1,12,89,RC)
1680 SUBEND
1690 SUB WAIT(B(),)
1700 CALL SPRITE(#11,120,14,137,255,0,0,#10,124,4,153,121,0,0,#5,60,15,137,49)
1710 CALL MOONRIVER(B(),),1)
1720 SUBEND
1730 SUB ASCII

```

```

1740 DATA ,00030506060503,00C4AC2424A4C4,00F79497F414F7,00D252D25F42C2,003F080A0
B0A0A,00FF00DE12121E
1750 DATA 00F500B5D59595,00CF08C80F00CF,00FE013DA5A5BD,00D010B010101,000010107C1
01,01214086080103,200408011882100C
1760 DATA 0020C108C304003,0610400C600CC110,00182424242418,00081808080808,0018240
408103C,00182408042418,00282828380808
1770 DATA 003C2038042418,00182038242418,003C040810101,00182418242418,001824241C0
418,0000303000303,000030300030102
1780 DATA 003E00F0003E,, ,00000000000F0909,0F0F,0000000000000E
1790 DATA FCFF,00000000008F4929,FF2F408,0000000000000E,FCFF,0102050F7FED0B1B,8B
ED7F0F050201
1800 DATA C020D0F8F8D8ED6E,ECD8F8F8D020C,0103050EFF3D0B1B,0B3DFF0E050301,C0E0D03
8F8D8EC6E,EDD8F838D0E0C,
1810 DATA 00000000009F7F9F,, ,0000000000C0E0C,0E173F7FCF0E0E4F,3F0F1F3F7F7F3F1F,00
00808,80C0C0EFFEFCF8F
1820 DATA ,007848F8FCFE44,, ,1B24242020A04,1F2828180F0A291,00008780809867
1830 DATA 0080808E111131CF,FFFFFFFFFFFFFFFF,00F808136282FB,000000D65951D1,000000
74704074,00001CE3A2A21C,0000000E3050900F
1840 DATA 0000D824242522,000000384478C03C,0408143F22227F7F,FEFFFF,000000E02020F0
FC,7FFFFFF,FFFFFFFFFFFFFFFF
1850 DATA 00F08087F414F7,000000BD2525BD,0000005D9C101D,, ,00000054927C9254,0000000
0010E01,0000002A493E492A
1860 DATA ,00000044AA92AA44,000001020508,0000002255495522,, ,01070D1F3F77FFFB,
1870 DATA 80F0F87CEEBEFFEF,FFFFFFFFFFFFFFFF,FF7F3F1F07,FFFFFFFFFFFFFFFF,FFFEFCF8
E,000103070F1F3F3F,7FFF
1880 DATA C0E0F0E0F8F8FCFE,FEFF,FF7FFF7F7F7F3F1E,1C0C,FFFEFCF8FCFCF8F,F06,000000
0101010307
1890 DATA 07070F1F3F77FFFF,00008080C08080C,E0E0F0F8FCFCFEFF,FFFF7F3F1F070301,, ,FF
FFFEF8F0E0C08,,
1900 RESTORE 1740
1910 FOR I=32 TO 143 :: READ C$ :: CALL CHAR(I,C$):: NEXT I
1920 SUBEND

```

**Ein Spiel
mit toller
Grafik**

SLICKS

*SLICKS ist auf dem TI 99/4 A in
Extended BASIC geschrieben und
wird mittels Joystick 1 gesteuert.*

Autorennen gibt es viele.

Wir unterscheiden hierbei zwischen 2 Arten: Die erste Art simuliert ein Autorennen in der Draufsicht der Straße und hat eine schlichte Grafik. Solche Autorennen sind schon mehrfach als BASIC-Programme für die verschiedensten Computersysteme veröffentlicht worden.

Bei der zweiten Art handelt es sich um Autorennen, die das Gefühl, im Rennwagen zu sitzen, perfekt simulieren. Das Rennen läuft so ab, als ob man selbst am Steuer sitzt. Diese Autorennen bieten eine perfekte Grafik und einen ausgezeichneten 3D-Effekt. Lei-



der handelt es sich hierbei um Maschinenprogramme, die meist für teures Geld auf Modulen zu kaufen sind. Dies alles ändert sich jedoch schlagartig

mit diesem Programm, denn hier kommt SLICKS, das Autorennen der zweiten Art, als BASIC-Programm. Tippen Sie das Programm ein, und nehmen Sie Platz in Ihrem TI Formel 1-Rennwagen. Geben Sie den Befehl

RUN, und in ca. einer Minute wird das Rennen beginnen.

Nachdem das Titelbild erschienen ist und die Melodie erklingt, drücken Sie den roten Aktionsknopf am linken Joystick, und das Rennen beginnt. Steuern Sie Ihren gelben Rennwagen mittels des Joysticks nach rechts oder links. Weichen Sie hierbei allen möglichen Sachen, die Ihnen entgegenkommen, aus. Die Geschwindigkeit steigert sich in 11 Stufen, die an der Höhe

des Motorengeräusches zu unterscheiden sind. Haben Sie einen Gegenstand erforderlich umfahren, so werden Ihnen hierfür Punkte gutgeschrieben.

Wenn Sie jedoch einen Unfall bauen oder von der Straße abkommen, so wird Ihnen ein Auto abgezogen. Zu Spielbeginn stehen Ihnen 3 Fahrzeuge zur Verfügung, mit denen Sie das Rennen bestreiten.

Vorstehendes Bild zeigt eine Zusammenstellung der verschiedenen Situationen auf der Straße. Es können Ihnen also entgegenkommen:

Rote Rennwagen. Diese sind am einfachsten zu umfahren.

Grüne Ölflecken. Hier wird die Sache schon etwas schwieriger.

Blaue mit Wasser gefüllte Schlaglöcher, die am schwierigsten zu beherrschen sind.

Je nach Geschwindigkeit werden Ihnen an Punkten gutgeschrieben:

Rennwagen: 1 bis 11 Punkte
Ölflecken: 2 bis 22 Punkte
Schlaglöcher: 3 bis 33 Punkte

Nachdem Sie alle 3 Rennwagen in die „ewigen Jagdgründe“ befördert haben, so erscheint wieder der Titelschriftzug, und die Melodie erklingt. Jetzt können Sie durch Druck auf den roten Aktionsknopf ein neues Rennen beginnen.

Beim Spielverlauf sind keine Besonderheiten zu beachten. Sogar die „ALPHA LOCK“ Taste kann eingaset sein.

```

100 !*****
110 !*          SLICKS          *
120 !*-----*
130 !*      TI 99/4 A          *
140 !*   Extended BASIC      *
150 !*-----*
160 !* (c)1984 Tronicsoft *
170 !*      (V.B)           *
180 !*****
190 !
200 OPTION BASE 1
210 DIM B(3,7),PZ(4),FK(4),ZH(4)
220 DATA 195,261,329,164,220,261,174,220,293,195,245,349,184,261,293,207,293,329
,174,220,261
230 RESTORE 220
240 FOR I=1 TO 7 :: FOR K=1 TO 3 :: READ B(K,I):: NEXT K :: NEXT I
250 CALL CLEAR :: CALL SCREEN(9):: CALL COLOR(0,9,9):: CALL HCHAR(1,1,30,768)
260 DATA 70,90,106,121,1,3,6,10,109,121,141,169
270 FOR I=1 TO 4 :: READ PZ(I):: NEXT I
280 FOR I=1 TO 4 :: READ FK(I):: NEXT I
290 FOR I=1 TO 4 :: READ ZH(I):: NEXT I :: RANDOMIZE
300 FOR I=1 TO 14 :: CALL COLOR(I,1,1):: NEXT I
310 CALL GRAFIK :: CALL BILD
320 CS$="DA828200DA0A0A0ADB0E0404000404046EDA828200038282DA5B5090001B81415B"
330 CS$="000038100000000000103810000000000107C3800000000000387CFEFE7C"
340 FOR I=12 TO 21 :: CALL HCHAR(12,I,I+36):: NEXT I
350 CALL COLOR(0,6,6,1,4,2,2,13,6,5,16,2,6,16,2,7,16,2,8,16,2,9,16,2,10,16,2,11,
16,2,12,16,2)
360 CALL COLOR(13,13,4,14,4,4,3,16,6,4,16,6):: CALL HCHAR(1,1,32,32):: CALL C2 :
: CALL TITEL(CS$,CS$,B(,)):: CALL RECHAR :: GOSUB 1000
370 CALL SPRITE(#1,135,11,175,121)
380 SPEED=11

```

Weiter Seite 46



Impressum: Computronic erscheint alle 2 Monate im Verlag: Tronic-Verlagsgesellschaft mbH, Landstr. 29, 3444 Wehretal 1, Tel.: (0 56 51) 4 06 43 / 4 06 93. Redaktion: (Verantw.) Axel Gredé, Rainer Beck, Frank Brall. Freie Mitarbeiter: Volker Becker, Rolf Freitag, H. Grede, Gesamtherstellung: D+V Druckhaus Die-richts Kassel. Vertrieb: Inland (Groß-, Einzel- und Bahnhofsbuchhandel), sowie Österreich und Schweiz: Verlagsunion, 6200 Wiesbaden. Anzeigen-

verwaltung u. Software-Service: A. Kratzenberg. Anzeigenpreis: Es gilt die Anzeigenliste Nr. 1. Bitte Media-Unterlagen anfordern. Bezugspreise: Einzelheft (Inland) DM 6,50, Abonnement (Inland) DM 35,-, (Ausland) DM 45,-. Autoren und Manuskripte: Bei Zusendung von Manuskripten und Datenträger erteilt der Autor dem Verlag die Genehmigung zum Abdruck und Versand der veröffentlichten Programme auf Datenträger. Für die mit Namen des Verfassers gekenn-

zeichneten Beiträge übernimmt die Redaktion lediglich die presserechtliche Verantwortung. Alle in dieser Zeitschrift veröffentlichten Beiträge sind urheberrechtlich geschützt. Nachdruck (auch auszugsweise) und Vervielfältigung nur mit ausdrücklicher Genehmigung des Verlages. Für unaufgeforderte Einsendungen von Manuskripten, Tonträgern und Software wird keine Haftung übernommen.

So machen Sie mehr aus Ihrem **COMMODORE 64**:

Tips & Tricks



DER BESTSELLER – BAND 1

64 Tips & Tricks, das mit über 70.000 Exemplaren meistverkaufte DATA BECKER BUCH, ist eine hochinteressante Sammlung von Anregungen zur fortgeschrittenen Programmierung des COMMODORE 64, POKE's und andere nützliche Routinen, interessanten Programmen. Aus dem Inhalt: 3D-Graphik in BASIC – Farbige Balkengraphik – Definition eines eigenen Zeichensatzes – Tastaturbelegung – Simulation der Maus mit einem Joystick – BASIC für Fortgeschrittene – C-64 spricht deutsch – CP/M auf dem COMMODORE 64 – Druckeranschluß über den USER-Port – Datenübertragung von und zu anderen Rechnern – Synthesizer in Stereo – Retten einer nicht ordnungsgemäß geschlossenen Datei – Erzeugen einer BASIC-Zeile in BASIC – Kassettenpuffer als Datenspeicher – Multitasking auf dem COMMODORE 64-POKE's und die Zeropage.– GOTO, GOSUB und RESTORE mit berechneten Zeilennummern, INSTR und STRING-Funktion – Repeat-Funktion für alle Tasten. Alle Maschinenprogramme mit BASIC-Ladeprogrammen.

64 Tips & Tricks ist eine echte Fundgrube für jeden COMMODORE 64 Anwender. 64 TIPS & TRICKS, 1984, über 300 Seiten, DM 49.–

JETZT NOCH MEHR TIPS & TRICKS – BAND 2

Auch der zweite Band von 64 Tips & Tricks dürfte sehr schnell ein Bestseller werden. Das Buch enthält eine Fülle hochkarätiger Programme, Anregungen und Routinen: ein umfangreiches Kapitel über Softwareschutz – Befehlsweiterungen und wie man sie macht – Tips & Tricks zur Programmierung von Superspielen – Zeiger und deren Manipulation – mehr übers Interrupt-Handling mit vielen Beispielen – erweiterte Hardware-Möglichkeiten – Betriebssystem ins RAM kopieren und dort manipulieren – sowie viele weitere Programme, Befehlsweiterungen und nützliche Routinen. Wer gerne programmiert und mehr wissen will über den COMMODORE 64, der braucht dieses neue Buch. 64 TIPS & TRICKS Band 2, ca. 250 Seiten, DM 39.–



DATA BECKER'S GROSSE PROGRAMM-SAMMLUNG ZUM COMMODORE 64, 250 Seiten, DM 49.–



DER COMMODORE 64 UND DER REST DER WELT, 220 Seiten, DM 49.–



DAS TRAININGSBUCH ZU SIMON'S BASIC, 380 Seiten, DM 49.–



COMMODORE 64 FÜR TECHNIK UND WISSENSCHAFT, 300 Seiten, DM 49.–



DAS IDEENBUCH ZUM COMMODORE 64, 240 Seiten, DM 29.–

Diese und viele weitere DATA BECKER BÜCHER gibt's im Buchhandel, im Computerefachhandel und in den Warenhäusern. Dort gibt's auch den kostenlosen, großen DATA BECKER Katalog mit der großen Buch- und Softwareauswahl rund ums Thema Computer. Katalog auch kostenlos direkt von DATA BECKER.

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (02 11) 31 00 10

BESTELL-COUPON
 Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1
 Bitte senden Sie mir:
 per Nachnahme Versandkosten
 zzgl. DM 5.– Verrechnungsscheck liegt bei
 Name und Adresse
 bitte deutlich
 schreiben


```

390 V=(RND*2*SPEED-SPEED)
400 CALL C2
410 CALL SOUND(-4250,10*SPEED,0,-6,5)
420 CALL SPRITE(#2,132,9,99,121,SPEED,V)
430 FOR I=3 TO 4
440 CALL JOYST(1,X,Y):: CALL MOTION(#1,0,3*X):: CALL C1 :: CALL PAUSE(42-2*SPEED)
450 CALL POSITION(#1,Y,X):: IF (X>185)+(X<57)THEN 730
460 CALL POSITION(#2,X,Y):: IF X>=PZ(I)THEN CALL PATTERN(#2,131+I)
470 CALL JOYST(1,X,Y):: CALL MOTION(#1,0,3*X):: CALL C2 :: CALL PAUSE(42-2*SPEED)
480 CALL POSITION(#2,X,Y):: IF X>=PZ(I)THEN CALL PATTERN(#2,131+I):: GOTO 500
490 GOTO 440
500 NEXT I
510 P=P+(SPEED-10)
520 CALL PEEK(-31877,CN):: IF (CN AND 32)=32 THEN 730
530 CALL C1 :: CALL DELSPRITE(#2):: CALL JOYST(1,X,Y):: CALL MOTION(#1,0,3*X)
540 CALL PUNKTE(P)
550 SPEED=SPEED+1 :: IF SPEED>21 THEN SPEED=21
560 ON INT(RND*2.5)+1 GOTO 390,570,840
570 V=(RND*128)-64 :: SH=V/10
580 CALL C2
590 CALL SOUND(-4250,10*SPEED,0,-6,5)
600 FOR I=1 TO 3 STEP 2
610 CALL JOYST(1,X,Y):: CALL MOTION(#1,0,3*X):: CALL C1
620 CALL POSITION(#1,Y,X):: IF (X>185)+(X<57)THEN 730
630 CALL SPRITE(#2,139+I,8,ZH(I),121+FK(I)*SH)
640 CALL JOYST(1,X,Y):: CALL MOTION(#1,0,3*X):: CALL C2
650 CALL SPRITE(#2,140+I,8,ZH(I+1),121+FK(I+1)*SH)
660 NEXT I
670 P=P+3*(SPEED-10)
680 CALL PEEK(-31877,CN):: IF (CN AND 32)=32 THEN 730
690 CALL C1 :: CALL DELSPRITE(#2):: CALL JOYST(1,X,Y):: CALL MOTION(#1,0,3*X)
700 CALL PUNKTE(P)
710 SPEED=SPEED+1 :: IF SPEED>21 THEN SPEED=21
720 GOTO 390
730 CALL MOTION(#1,0,0):: CALL DELSPRITE(#2):: CALL COLOR(#1,10)
740 FOR I=0 TO 15 STEP 2 :: CALL SOUND(-100,-7,I)
750 ON I+1 GOSUB 800,800,800,800,810,810,810,810,820,820,820,820,830,830,830,830
760 NEXT I
770 CALL DELSPRITE(#1)
780 ATO=ATO-1 :: CALL AUTO(ATO):: IF ATO=0 THEN 1010
790 GOTO 370
800 CALL COLOR(#1,10):: CALL PATTERN(#1,33):: RETURN
810 CALL COLOR(#1,11):: CALL PATTERN(#1,34):: RETURN
820 CALL COLOR(#1,12):: CALL PATTERN(#1,38):: RETURN
830 CALL COLOR(#1,15):: CALL PATTERN(#1,39):: RETURN
840 V=(RND*128)-64 :: SH=V/10
850 CALL C2
860 CALL SOUND(-4250,10*SPEED,0,-6,5)
870 FOR I=1 TO 3 STEP 2
880 CALL JOYST(1,X,Y):: CALL MOTION(#1,0,3*X):: CALL C1
890 CALL POSITION(#1,Y,X):: IF (X>185)+(X<57)THEN 730
900 CALL SPRITE(#2,123+I,3,ZH(I),121+FK(I)*SH)
910 CALL JOYST(1,X,Y):: CALL MOTION(#1,0,3*X):: CALL C2
920 CALL SPRITE(#2,124+I,3,ZH(I+1),121+FK(I+1)*SH)
930 NEXT I
940 P=P+2*(SPEED-10)
950 CALL PEEK(-31877,CN):: IF (CN AND 32)=32 THEN 730
960 CALL C1 :: CALL DELSPRITE(#2):: CALL JOYST(1,X,Y):: CALL MOTION(#1,0,3*X)
970 CALL PUNKTE(P)
980 SPEED=SPEED+1 :: IF SPEED>21 THEN SPEED=21
990 GOTO 390

```


Alles über den Commodore 64



① Das sollte Ihr erstes Buch zum COMMODORE 64 sein. Eine sehr leicht verständliche Einführung in Handhabung, Einsatz, Ausbaumöglichkeiten und Programmierung des C64, die keinerlei Vorkenntnisse voraussetzt. Viele Abbildungen, Fotos und nützliche Anwendungsbeispiele ergänzen den Text. Auch als Orientierung vor dem 64er Kauf gut geeignet. **64 FÜR EINSTEIGER**, ca. 200 S., DM 29,-

② Dieses über 65.000mal verkaufte Standardwerk zum COMMODORE 64 braucht jeder ernsthafte Anwender. Alles über Technik, Betriebssystem und fortgeschrittene Programmierung des C64. Mit ausführlichem ROM-Listing, sorgfältig dokumentierten Originalschaltplänen zum Ausklappen, zahlreichen Abbildungen, Schaltbildern, Blockdiagrammen und natürlich nützlichen Programmen. Mit diesem unentbehrlichen Buch lernen Sie Ihren C64 erst richtig kennen. **64 INTERN**, ca. 350 S., DM 69,-

③ Der Bestseller zur Graphikprogrammierung des COMMODORE 64 vom Autor der berühmten Supergraphik. Für Einsteiger, Fortgeschrittene und Profis. Bringt alles von den Grundlagen der Graphikprogrammierung über Sprites, High-Res-Graphik, Multicolor, Zeichensatzprogrammierung bis hin zu dreidimensionaler Graphik und CAD. Unzählige Superprogramme und Routinen zum Abtippen. **DAS GRAFIKBUCH ZUM COMMODORE 64**, 295 S., DM 39,-

④ Das Superbuch, das Ihnen zeigt, was alles in Ihrem Rekorder steckt. Informiert detailliert und leicht verständlich über Datensette und Cassetten-Speicherung. Mit absoluten Spitzenprogrammen: Autostart, Catalog (sucht und lädt automatisch!), Backup von und auf Floppy, Save von Speicherbereichen und das Tollste: ein neues Cassetten-Betriebssystem mit dem 10-20mal schnelleren Fasttape. Außerdem weitere nützliche Hinweise (Kopfjustage, Kontroll-Lautsprecher) und Programme. **DAS CASSETTENBUCH ZUM COMMODORE 64 UND VC-20**, ca. 180 S., DM 29,-

⑤ Das über 50.000mal verkaufte Standardwerk zur Floppy VC-1541. Alles über Diskettenprogrammierung für Einsteiger, Fortgeschrittene und Profis. Neben grundlegenden Informationen zum DOS, zu den Systembefehlen und Fehlermeldungen stehen mehrere Kapitel zur praktischen Dateiverwaltung mit der Floppy. Umfangreiches, dokumentiertes DOS-Listing. Dazu eine Fundgrube verschiedenster Programme und Hilfsroutinen, die das Buch für jeden Floppy-Anwender zur Pflichtlektüre machen. **DAS GROSSE FLOPPY-BUCH**, ca. 320 S., DM 49,-

⑥ Mit diesem Buch meistern Sie jedes Drucker-Problem. Ob Sekundäradressen, Schnittstellen, Steuerzeichen, formatierte Datenausgabe oder Graphik-Hardcopy, alles wird hervorragend erklärt. Selbstverständlich wieder viele nützliche Programme zum Abtippen. Außerdem wichtige Hilfen zur Druckeranpassung, ein Betriebssystemlisting des MPS 801 und ein eigenes Kapitel zum VC-1520. Mit diesem Buch holen Sie das Optimum aus Ihrem Drucker heraus. **DAS GROSSE DRUCKERBUCH**, über 300 S., DM 49,-

DATA BECKER

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

BESTELL-COUPON

Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1

Zzgl. DM 5,- Versandkosten

per Nachnahme Verrechnungsscheck liegt bei

Bitte senden Sie mir:

Name und Adresse bitte deutlich schreiben


```

1000 P=0 :: ATO=3 :: CALL PUNKTE(P):: CALL AUTO(ATO):: RETURN
1010 CALL TITEL(CB$,CS$,B(,))
1020 GOSUB 1000
1030 GOTO 370
1040 SUB POEME(B(,),KY)
1050 DATA 2,1,3,311,30000,3,329,30000,2,2,6,440,30000,1,3,1,293,30000,1,329,3000
0,1,391,30000
1060 DATA 1,4,1,349,30000,1,329,30000,1,311,30000,2,1,1,349,30000,5,329,30000,1,
5,3,369,30000,1,1,3,391,30000
1070 DATA 2,6,5,523,30000,1,493,30000,1,2,1,415,30000,1,440,30000,1,659,523
1080 DATA 1,5,2,587,493,1,523,440,2,4,1,523,440,5,493,391
1090 DATA 1,1,3,659,523,1,2,3,659,523,1,7,1,659,523,1,587,493,1,493,415,1,2,1,52
3,440
1100 DATA 1,587,493,1,659,523,2,3,6,880,698,2,4,1,880,698,1,783,659,2,1396,880,1
,1318,783,1,1174,698
1110 DATA 1,2,3,1046,659,1,7,3,1046,587,2,1,1,1046,659,1,987,30000,1,659,523
1120 DATA 3,783,659,2,4,1,783,30000,1,587,30000,1,783,30000,1,698,30000,1,659,30
000,1,622,30000
1130 DATA 2,1,1,698,30000,5,659,30000
1140 DATA 1,1,3,659,523,1,2,3,659,523,1,7,1,659,523,1,587,493,1,493,415,1,2,1,52
3,440
1150 DATA 1,587,493,1,659,523,2,3,6,880,698,2,4,1,880,698,1,783,659,2,1396,880,1
,1318,783,1,1174,698
1160 DATA 1,2,3,1046,659,1,7,3,1046,587,2,1,1,1046,659,1,987,30000,1,659,523
1170 DATA 3,783,659,2,4,1,783,30000,1,587,30000,1,783,30000,1,698,30000,1,659,30
000,1,622,30000
1180 DATA 2,1,1,698,30000,5,659,30000,0,0
1190 RESTORE 1050
1200 AZ,STV,CL=0
1210 READ K :: IF K=0 THEN 1190
1220 READ INR
1230 FOR I=1 TO K
1240 FOR J=1 TO 3
1250 CALL C1
1260 STV=STV+1 :: IF STV>AZ THEN READ AZ,M1,M2 :: STV=1
1270 CALL SOUND(420,M1,0,M2,0,B(J,INR),6):: CALL KEY(1,T,S):: IF S<>0 THEN SUBEX
IT
1280 CALL C2
1290 NEXT J :: NEXT I
1300 GOTO 1210
1310 SUBEND
1320 SUB RECHAR
1330 DATA 00182424242418,0008180808080808,0018240408103C,00182408042418,0028282838
0808,003C2038042418,00182038242418
1340 DATA 003C040810101,00182418242418,001824241C0418
1350 RESTORE 1330
1360 CALL HCHAR(12,12,30,10):: CALL COLOR(3,11,2,4,11,2)
1370 FOR I=48 TO 57 :: READ C$ :: CALL CHAR(I,C$):: NEXT I
1380 SUBEND
1390 SUB PUNKTE(P)
1400 DISPLAY AT(1,23)SIZE(6):USING "#####":P
1410 SUBEND
1420 SUB AUTO(ATO)
1430 CALL HCHAR(1,3,32,3)
1440 IF ATO=0 THEN SUBEXIT
1450 CALL HCHAR(1,3,59,ATO)
1460 SUBEND
1470 SUB TITEL(C1$,C2$,B(,))
1480 CALL CHAR(124,C1$)
1490 PS=97 :: FOR I=1 TO 4 :: CALL SPRITE(#1,123+I,10,73,PS):: PS=PS+16 :: NEXT
I
1500 CALL POEME(B(,),1)
1510 FOR I=1 TO 4 :: CALL DELSPRITE(#I):: NEXT I

```

```

1520 CALL CHAR(124,C2#)
1530 SUBEND
1540 SUB C1 :: CALL COLOR(7,16,2,8,16,2,11,16,2,5,2,2,6,2,2,9,2,2,10,2,2,12,2,2,
13,13,4,14,4,4):: SUBEND
1550 SUB C2 :: CALL COLOR(5,16,2,6,16,2,9,16,2,10,16,2,12,16,2,7,2,2,8,2,2,11,2,
2,14,13,4,13,4,4):: SUBEND
1560 SUB PAUSE(Z):: FOR I=1 TO Z :: NEXT I :: SUBEND
1570 SUB BILD
1580 DATA (*-/) + (*(*-/) + . . . .) + (*-/) + - / (*)+
1590 DATA #####$vw^%#####,#####$axzed%#####,###
#####$bcly{waf%#####
1600 DATA #####$PCWNP$WNYX%#####,#####$RSWWqtwWNCZ%#####,###
#####$TSWWP$WNC%#####
1610 DATA #####$QAWWWWhIWWWWIH%#####,#####$BCWWWWWimWWWWWkJK%#####,###
#####$DEWWWWWhnWWWWWWML%#####
1620 DATA #####$FGWWWWWWkOWWWWWWWON%###
1630 DATA 15,13,128,15,20,129,16,11,136,16,22,136,18,9,130,18,24,130
1640 DATA 19,9,131,19,24,131,22,5,138,23,5,139,22,28,138,23,28,139
1650 DATA 7,4,63,3,7,61,5,10,60,5,12,62,6,19,61,4,23,60,4,25,62,7,29,63
1660 RESTORE 1580
1670 FOR Z=14 TO 24 :: READ C# :: FOR S=1 TO 32
1680 CALL VCHAR(Z,S,ASC(SEG$(C#,S,1)))
1690 NEXT S :: NEXT Z
1700 FOR I=1 TO 12 :: READ Z,S,C :: CALL VCHAR(Z,S,C):: NEXT I
1710 CALL MAGNIFY(2)
1720 FOR I=21 TO 28 :: READ Z,S,C :: CALL SPRITE(#I,C,16,8*Z,8*S):: NEXT I
1730 SUBEND
1740 SUB GRAFIK
1750 DATA ,422418BDF7E3C5A,422418ADF563C4A,FFFFFFFFFFFFFFFF,FFFEFCF8F0E0C08,FF
7F3F1F0F070301,022010A93D502842
1760 DATA 0220008821100042,081C3C7F7FFFFFFF,03071F1F3F7F7FFF,4070F8FCFEFFFFFF,C0
E0F8F8FCFEFEFF,,0000000C1F7F7FFF
1770 DATA ,000000000C0F8FE,00030506060503,00C4AC2424A4C4,00F79497F414F7,00D252D
25F42C2,003F080A0B0A0A
1780 DATA 0FF00E12121E,00F500B5D59595,00CF08C80F00CF,00FE013DA5A5BD,00D010B810
101,0000303000303,0010387CB8FE92
1790 DATA 010F3FFFFFF7F0C,00187C7EFF7F3E1,0000C0ECFE7F0E04,00007C3E1F0E04,0103070
F1F3E7EFC,F0E0C08,0103070F1F3F7EFC
1800 DATA F8F0E0C08,0103070F1F3F7FFE,FCF8F0E0C08,0103070F1F3F7FFF,FCF8F8F0E0C08,
80C0E0F0F87C7E3F,0F070301
1810 DATA 80C0E0F0F8FC7E3F,1F0F070301,80C0E0F0F8FCFE7F,3F1F0F070301,80C0E0F0F8FC
FEFF,3F1F1F0F070301,0103070E1C3878F
1820 DATA C08,0103070F1E3C78F,E0C08,0103070F1E3E7CF8,0044444444444438,0102040C18
3060C,
1830 DATA 80C0E070381C1E0F,0301,80C0E0F0783C1E0F,070301,80C0E0F0787C3E1F,0038080
808080838,80402030180C0603
1840 DATA 00000000000007C,0103060C183870E,8,0103070E1C3870E,C08,80C06030181C0E0
7,01
1850 DATA 80C0E070381C0E07,0301,1F1F1F1F1F1F1F1F,1F1F1F1F3F3F3F3F,3F3F3F3F7F7F7
7F,7F7F7F7FFFFFFF,F8F8F8F8F8F8F8F8
1860 DATA F8F8F8F8FCFCFCFC,FCFCFCFCFEFEFEFE,FEFEFEFEFFFFFFF,0303030307070707,07
07070707070707,0F0F0F0F0F0F0F0F,C0C0C0C0E0E0E0E
1870 DATA E0E0E0E0E0E0F0F,F0F0F0F0F0F0F0F,000000101010101,000000808080808,01010
101010101,0303030303030303,808080808080808
1880 DATA C0C0C0C0C0C0C0,001010100010101,003008080408083,0000205408,,0000103810
381,0000081C081C08
1890 DATA 00000018183C183C,7E183C181818,000018183C18,0000183C183C3C,00183C3C5A7E
5A,185A7E5A18BDF99,00183C183C7E1818
1900 DATA ,183C7E18183C7EFF,18183C7EFF181818,0000081C08,00040E1F0E04,08142E5F2E1
408,2854AE5BAE5428
1910 RESTORE 1750
1920 FOR I=32 TO 143 :: READ C# :: CALL CHAR(I,C#):: NEXT I
1930 SUBEND

```


Super Miner

Das Programm läuft auf allen Atari-Computern mit mindestens 32 Kb Ram und wird mit Joystick in Port 1 gesteuert.

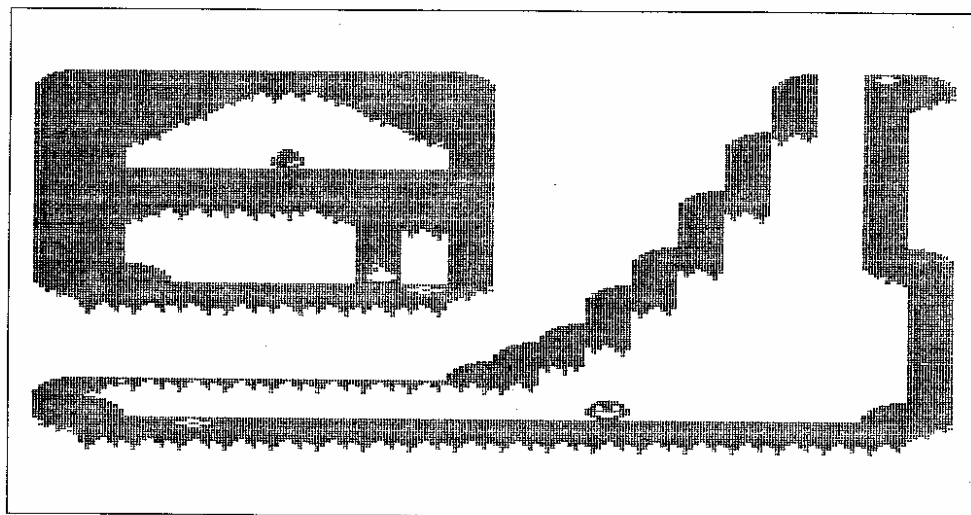
Nachdem nun der mutige Abenteurer, dem wir schon in THE BIG QUEST begegnet sind, viele Perlen gesammelt, viele Türen geöffnet und viele Monster GEZAPPT hat, begegnen wir ihm bei seinem zweiten Abenteuer, daß in den verlassenenen Minen von CLUBS spielt.

Da er sehr lange gebraucht hat, um das Schloß auszuräumen, befinden wir uns inzwischen schon im Jahre 2000! Da der technische Fortschritt auch in den Minen nicht aufzuhalten ist, wurden Fahrstühle und ähnliche Teufelsmaschinen eingebaut, um die Arbeit zu erleichtern.

Da die Minen schon seit 1994 verlassen sind, funktionieren die ganzen Erfindungen der Technik nun nicht mehr ganz so wie sie sollten, was unserem Abenteurer natürlich zum Verhängnis, wird wenn ...

An dieser Stelle sollen Sie Ihren Joystick zur Hand nehmen, es sich in Ihrem Sessel bequem machen und dem armen, ach so sturen Gesellen ein wenig zur Hand gehen.

Da er schon immer sehr geldgierig war, will er die Minen natürlich ausbeuten. Also: steuern Sie die Figur, die Ihnen schon aus THE BIG QUEST entgegenlacht, über die alten, brüchig gewordenen Ebenen und sammeln Sie



alle Gegenstände ein, die einsammelbar sind. Doch Vorsicht: Es gibt auch Gegenstände, die nur aufgestellt wurden, um uneingeweihte hinteres Licht zu führen!

Eine kleine Bemerkung

Das Programm THE BIG QUEST läuft nur vollständig, wenn die Punkte in den DATAZEILEN ab 10 000 auch als PUNKTE und nicht als KOMMAS eingegeben werden!

Eine kleine Aufzählung der Gefahren, die Sie in den Minen bedrohen:

- Falltüren
- Absackende Ebenen
- Spitzen
- Löcher
- Abbröckelnde Ebenen
- Bomben
- Zapfen
- Transmitter-Türen
- Tödliche Abgründe
- Knöpfe
- Unsichtbare Ebenen ...

Alle Bildschirme sind SCHAFFBAR!

```

0 DIM A$(400):A$(1)=" ":A$(400)=" ":A$(2)=A$:REM WEG DAMIT !!!
1 REM *****
2 REM **** SUPER MINER 2! ****
3 REM ** COPYRIGHT (C) 1984 **
4 REM * BY THE BYTE NIBBLER! *
8 REM *****
10 PAGE=PEEK(106)-8:CHS=PAGE*256:POKE 77,128:POKE 559,0:RESTORE 28000
12 GOTO 20
13 SOUND 0,10,8,L:L=L+5:RETURN
20 READ C:IF C<>-1 THEN B=CHS+C*8:FOR I=0 TO 7:READ A:POKE B+I,A:NEXT I:GOTO 20
50 REM SPIELBEGINN
60 M=9:P=0:SC=1:SCM=10
61 LVL=10
70 GRAPHICS 17:POKE 756,PAGE:POKE 709,132:POKE 710,30:POKE 711,184
80 COLOR 235:PLOT 0,0:POSITION 1,0:?" #5)":"4"
100 REM NEUER SCREEN
101 FOR I=1 TO 23:COLOR 32:PLOT 0,I:DRAWTO 19,I:NEXT I
102 RESTORE 28990+LVL*10:FOR I=0 TO 7:READ A:POKE CHS+336+I,A:NEXT I:GOSUB 1950

```



```

105 BF=0
110 POKE 559,0:POSITION 0,4:GOSUB 9750+SC*250:POKE 559,34
120 COLOR 235:PLOT X,Y:H=32
149 TRAP 5000
150 REM HAUPTSCHLEIFE
160 POKE 77,0:ST=STICK(0)
170 IF ST=11 AND X>0 THEN GOSUB 1000:SOUND 0,0,0,0
180 IF ST=7 AND X<19 THEN GOSUB 1100:SOUND 0,0,0,0
190 SOUND 0,0,0,0
200 IF STRIG(0)=0 THEN GOSUB 1300:GOSUB 1600
210 GOSUB 1300
220 IF ST=14 AND H=104 THEN GOSUB 1800
230 IF ST=13 THEN GOSUB 1900
990 GOTO 150
1000 REM LEFT MOVE
1001 L=5
1010 LOCATE X-1,Y,Z:LOCATE X-1,Y-1,Z1
1020 IF Z=81 THEN 5000
1030 IF Z=202 THEN COLOR 32:PLOT X-1,Y:GOSUB 1200
1040 IF Z=39 AND (Z1=32 OR Z1=104) THEN COLOR H:PLOT X,Y:Y=Y-1:COLOR 235:PLOT X,
Y:Z=Z1
1050 IF Z<>32 AND Z<>104 THEN RETURN
1055 X=X-1
1060 COLOR 238:PLOT X,Y:COLOR 239:PLOT X+1,Y:GOSUB 13
1070 COLOR 236:PLOT X,Y:COLOR 237:PLOT X+1,Y:GOSUB 13
1080 COLOR 235:PLOT X,Y:COLOR H:PLOT X+1,Y:GOSUB 13
1085 H=Z
1090 RETURN
1100 REM RIGHT MOVE
1101 L=5
1110 LOCATE X+1,Y,Z:LOCATE X+1,Y-1,Z1
1120 IF Z=81 THEN 5000
1130 IF Z=202 THEN COLOR 32:PLOT X+1,Y:GOSUB 1200
1140 IF Z=40 AND (Z1=32 OR Z1=104) THEN COLOR H:PLOT X,Y:Y=Y-1:COLOR 235:PLOT X,
Y:Z=Z1
1150 IF Z<>32 AND Z<>104 THEN RETURN
1160 COLOR 236:PLOT X,Y:COLOR 237:PLOT X+1,Y:GOSUB 13
1170 COLOR 238:PLOT X,Y:COLOR 239:PLOT X+1,Y:GOSUB 13
1180 COLOR H:PLOT X,Y:COLOR 235:PLOT X+1,Y:GOSUB 13
1185 H=Z:X=X+1
1190 RETURN
1200 REM KEY GENOMMEN
1210 P=P+25:POSITION 4,0: ? #6:P
1220 FOR I=1500 TO 0 STEP -64:SOUND 0,I,10,I/100:NEXT I
1230 KEYS=KEYS-1:IF KEYS>0 THEN RETURN
1240 GOTO 7000
1300 REM DOWN?
1305 CTR=0
1310 LOCATE X,Y+1,Z
1320 IF Z=32 THEN SOUND 0,Y*10,10,10:COLOR H:PLOT X,Y:Y=Y+1:CTR=CTR+1:COLOR 235:
PLOT X,Y:GOTO 1310
1330 IF CTR>5 THEN 5000
1335 IF CTR>0 THEN RETURN
1340 IF Z>42 AND Z<48 THEN SOUND 0,Z,12,10:COLOR Z+1:PLOT X,Y+1:IF Z=47 THEN COL
OR 32:PLOT X,Y+1
1350 IF Z>42 AND Z<48 THEN RETURN
1360 IF Z=65 THEN SOUND 0,21,4,15:COLOR 32:PLOT X,Y+1:GOTO 1300
1370 IF Z=66 OR Z=73 THEN 5000
1380 IF Z=71 AND BF THEN 1500
1389 C=0
1390 IF Z=67 THEN LOCATE X,Y+2,C:IF C=32 OR C=33 THEN SOUND 0,98,12,15:COLOR 32:
PLOT X,Y+1:COLOR Z:PLOT X,Y+2
1395 IF Z=67 AND (C=32 OR C=33) THEN 1300

```

```

1400 IF Z<>42 THEN RETURN
1410 REM HOCH TELEPORTIEREN
1420 MIN=1
1430 LOCATE X,Y-MIN,Z:IF Z<>33 THEN MIN=MIN+1:SOUND 0,MIN,8,10:GOTO 1430
1440 LOCATE X,Y-MIN,Z:IF Z<>32 THEN MIN=MIN+1:SOUND 0,MIN,8,10:GOTO 1440
1450 FOR I=0 TO 15 STEP 0.75:SOUND 0,100,10,I:SOUND 1,120,10,I:NEXT I
1460 COLOR H:PLOT X,Y:COLOR 235:PLOT X,Y-MIN:Y=Y-MIN:H=32
1470 FOR I=15 TO 0 STEP -0.75:SOUND 0,100,10,I:SOUND 1,120,10,I:NEXT I
1480 RETURN
1500 REM BOMBE AUSGELÖST
1510 COLOR 80:PLOT BX,BY
1520 FOR I=15 TO 0 STEP -0.5:SOUND 0,100,8,I:SOUND 1,50,8,I:SOUND 2,30,8,I:NEXT I
1530 COLOR 32:PLOT BX,BY
1540 FOR I=43 TO 47:COLOR I:PLOT BX,BY-1:PLOT BX,BY-2:SOUND 0,I,2,10:FOR W=1 TO 10:NEXT W:NEXT I
1550 COLOR 32:PLOT BX,BY-1:PLOT BX,BY-2:SOUND 0,0,0,0:BF=0:RETURN
1600 REM JUMP
1610 FOR I=1 TO 3
1620 LOCATE X,Y-1,Z:IF Z<>32 THEN 5000
1630 COLOR H:PLOT X,Y:Y=Y-1:COLOR 235:PLOT X,Y:H=32
1640 GOSUB 1700:FOR U=I*50 TO I*50+49 STEP 15:SOUND 0,U,10,10:NEXT U
1650 NEXT I
1660 I=3
1670 LOCATE X,Y+1,Z:IF Z<>32 THEN RETURN
1680 COLOR H:PLOT X,Y:Y=Y+1:COLOR 235:PLOT X,Y:H=32
1690 GOSUB 1700:FOR U=I*50 TO I*50+49 STEP 15:SOUND 0,U,10,10:NEXT U
1695 I=I-1:IF I>0 THEN 1670
1697 RETURN
1700 REM LEFT OR RIGHT AT JUMP
1710 ST=STICK(0):XR=0
1720 XR=XR-(ST=11 AND X>0)
1730 XR=XR+(ST=7 AND X<19)
1740 LOCATE X+XR,Y,Z:IF Z<>32 THEN RETURN
1750 COLOR H:PLOT X,Y:H=32:X=X+XR:COLOR 235:PLOT X,Y:RETURN
1800 REM UP MOVE
1810 LOCATE X,Y-1,Z
1820 IF Z<>32 AND Z<>104 THEN RETURN
1825 SOUND 0,100,8,15
1830 COLOR H:PLOT X,Y:Y=Y-1:H=Z:COLOR 235:PLOT X,Y
1840 RETURN
1900 REM DOWN MOVE
1910 LOCATE X,Y+1,Z
1920 IF Z<>104 THEN RETURN
1925 SOUND 0,100,8,15
1930 COLOR H:PLOT X,Y:Y=Y+1:H=Z:COLOR 235:PLOT X,Y
1940 RETURN
1950 COLOR 32:PLOT 0,0:DRAWTO 19,0:POKE 756,224
1955 POSITION 0,8:? #6;"SAMMELN SIE NUN DIE:" :SETCOLOR 0,0,15
1960 READ A$:POSITION 0,14:? #6;A$
1970 IF STRIG(0)=1 THEN SOUND 0,PEEK(53770),8,10:SOUND 0,0,0,0:GOTO 1970
1980 COLOR 32:PLOT 0,8:DRAWTO 19,8:PLOT 0,14:DRAWTO 19,14:COLOR 235:PLOT 0,0:POSITION 1,0:? #6;"":M;"":P
1990 POKE 756,PAGE:RETURN
5000 REM KAPUTT
5001 TRAP 40000
5010 POP :POP :POP :M=M-1:POSITION 2,0:? #6;M
5015 FOR I=0 TO 15:SOUND 0,42,8,I:SOUND 1,22,8,I:SOUND 2,4,8,I:NEXT I
5020 FOR I=242 TO 250
5030 COLOR I:PLOT X,Y
5040 SOUND 0,I-200,8,15:SOUND 1,I-220,8,15:SOUND 2,I-240,8,15:FOR W=1 TO 12:NEXT W:NEXT I
5044 COLOR 32:PLOT X,Y

```



```

5045 FOR I=15 TO 0 STEP -1: SOUND 0,50,8,I: SOUND 1,30,8,I: SOUND 2,10,8,I: NEXT I
5050 SOUND 0,0,0,0: SOUND 1,0,0,0: SOUND 2,0,0,0
5060 IF M>0 THEN 100
5070 POKE 77,128: IF PEEK(53279)<>6 THEN 5070
5080 GOTO 50
7000 REM GESCHAFFT
7010 POP : POP : POP
7020 P=P+SC*1000: POSITION 4,0: ? #6:P
7030 FOR I=15000 TO 0 STEP -320: SOUND 0,1,10,I/1000: NEXT I
7035 LVL=LVL+1: IF LVL>19 THEN LVL=1
7040 SC=SC+1: IF SC>SCM THEN SC=1
7050 GOTO 100
10000 ? #6:" DFE#!XEER&' " ;
10010 ? #6:" D " ;
10020 ? #6:" " ;
10030 ? #6:" " ;
10040 ? #6:"<!!!!' " ;
10050 ? #6:"!!!!!! " ;
10060 ? #6:"!!!!!! " ;
10070 ? #6:"!!!!!! DEE " ;
10080 ? #6:"DEDFED " ;
10090 ? #6:" <!!!!' " ;
10100 ? #6:"!!!!!! " ;
10110 ? #6:"!!!!!! " ;
10120 ? #6:" $!!!!!! " ;
10130 ? #6:"!!!!!! " ;
10140 ? #6:"!!!!!! " ;
10150 ? #6:"<!!!!!!*!!!!!! " ;
10160 ? #6:"DEDFDFDEDEDDDFDFDFDF " ;
10170 ? #6:" " ;
10180 ? #6:" " ;
10190 ? #6:" " ;
10200 COLOR 202:PLOT 12,18:PLOT 2,7:
PLOT 15,12:X=1:Y=18:KEYS=3
10210 SETCOLOR 0,2,2
10220 RETURN
10250 ? #6:"<!!!!!!*!!!!!! " ;
10260 ? #6:"DEDEDEEFDFFDFDFD " ;
10270 ? #6:" " ;
10280 ? #6:" (<' " ;
10290 ? #6:" <!!!! " ;
10300 ? #6:" <!!!!!! " ;
10310 ? #6:" (<+++++*!!!!!! " ;
10320 ? #6:" DAAAAAAAAAADEDEEEEE " ;
10330 ? #6:" " ;
10340 ? #6:"<' " ;
10350 ? #6:"!!+' " ;
10360 ? #6:"!!+!!' " ;
10370 ? #6:"#!+!!!' " ;
10380 ? #6:" &+!!!!!BBBBB!!' " ;
10390 ? #6:" REDEEEEDDEDDED " ;
10400 ? #6:" " ;
10410 ? #6:" (<' " ;
10420 ? #6:" <!!!! " ;
10430 ? #6:"<IB!!!!!!*!!!!!! " ;
10440 ? #6:"DEFEDDEDEDEDEDEDEDD " ;
10450 COLOR 202:PLOT 1,21:KEYS=1:X=0:Y=3
10460 SETCOLOR 0,3,4
10470 RETURN
10500 ? #6:"<G' <!!!!' " ;
10510 ? #6:"FFF " ;
10520 ? #6:" " ;
10530 ? #6:" ) |XFED&' " ;
10540 ? #6:" | " ;
10550 ? #6:" (< | " ;
10560 ? #6:" ) $!' 0 " ;
10570 ? #6:" $!!!!!!+!!h' " ;
10580 ? #6:" FEFEFEFEFD " ;
10590 ? #6:" ) h " ;
10600 ? #6:" ) h " ;
10610 ? #6:" h " ;
10620 ? #6:"<!!K!!! " ;
10630 ? #6:"!!!!K!!!!!!' " ;
10640 ? #6:"!!!!K!!!!!!*!!!!!!' " ;
10650 ? #6:"DEDFDFD$!!!!!!*!!!!!!' " ;
10660 ? #6:" h &!!!!!!*!!!!!!' " ;
10670 ? #6:" h DD&!!!!!!*!!!!!!' " ;
10680 ? #6:" h FE$!!!!# " ;
10690 ? #6:"DDFDRR FDF " ;
10700 COLOR 202:PLOT 11,10:PLOT 13,3:
PLOT 6,22:X=0:Y=22:KEYS=2
10710 SETCOLOR 0,7,6:SETCOLOR 1,1,
8:BX=14:BY=10:BF=1.
10720 RETURN
10750 ? #6:" " ;
10760 ? #6:" <!!!!!!*!!!!!!h" " ;
10770 ? #6:" $!XEER&#!D$!%h" " ;
10780 ? #6:"h!' D D F h" " ;
10790 ? #6:"h!!! " ;
10800 ? #6:"h!!!!' +++ " ;
10810 ? #6:"h!!!!' RRR " ;
10820 ? #6:"h!!!!!!' <!!!!h!" " ;
10830 ? #6:"hEDFDFDFDF $!!!!h%" " ;
10840 ? #6:"h EDFDh " ;
10850 ? #6:"h h " ;
10860 ? #6:"h h " ;
10870 ? #6:"h h " ;
10880 ? #6:"h h " ;
10890 ? #6:"!!!!' h!!!% $!#" " ;
10900 ? #6:"!!!!#DD hFEE D " ;
10910 ? #6:"!# h " ;
10920 ? #6:"!!!' h " ;
10930 ? #6:"!!!!!!*!!!!!!' <!! " ;
10940 ? #6:"EFDDEFDFFDFDDDFREEFD" " ;
10950 COLOR 202:PLOT 2,20:PLOT 8,4:P
LOT 9,8:PLOT 1,6:KEYS=4:X=19:Y=21
10960 SETCOLOR 0,4,4:SETCOLOR 1,1,6
10970 RETURN
11000 ? #6:" " ;
11010 ? #6:" (<' " ;
11020 ? #6:"<!!#DE (<#!D&%R$' (<' " ;
11030 ? #6:"! &!! !% (<# !" " ;

```



```

11040 ? #6; "! &! <|&!!IDF % |";
11050 ? #6; "! ' ! &! D&! (<|");
11060 ? #6; "!%# ! !' &' (<|");
11070 ? #6; "! &# !ID |C! |!";
11080 ? #6; "! $! (<|!| $!|");
11090 ? #6; "! &|' ! ! &|!| &|";
11100 ? #6; "! !#DD ! ! &# &' |";
11110 ? #6; "!D ! &' &' |";
11120 ? #6; "! $|' !' | |";
11130 ? #6; "! !EDF !!!!!#DE |";
11140 ? #6; "!FE &' E$!!!% |";
11150 ? #6; "! !+' D&! ' (<|");
11160 ? #6; "!|' !+|!| | | |";
11170 ? #6; "!FDE DAD&! !' ! !#";
11180 ? #6; "&' (<' (<|%ADD% |";
11190 ? #6; " DDDDDDDDDDDDD ";
11200 COLOR 202:PLOT 11,7:PLOT 13,9:PLOT
12,11:PLOT 9,9:PLOT 16,16:PLOT 17,21:PL
OT 14,20:PLOT 11,22:PLOT 6,18
11202 PLOT 4,15:PLOT 1,19:PLOT 2,22
:PLOT 7,22:X=11:Y=5:KEYS=8
11210 SETCOLOR 0,7,2
11220 RETURN
11250 ? #6; " (<' (<' (<' |";
11260 ? #6; "FFF FFFF FFF FFF FFF";
11270 ? #6; " ";
11280 ? #6; " ";
11290 ? #6; " ";
11300 ? #6; " (<' (<' (<' |";
11310 ? #6; " FFFF FF FFF FFF FF ";
11320 ? #6; " ";
11330 ? #6; " ";
11340 ? #6; " ";
11350 ? #6; " ";
11360 ? #6; "FF FF FF FF FF FF ";
11370 ? #6; " ";
11380 ? #6; " ";
11390 ? #6; " ";
11400 ? #6; " ";
11410 ? #6; " FF F F F FF ";
11420 ? #6; " ";
11430 ? #6; " ";
11440 ? #6; " ";
11450 SETCOLOR 0,11,4:COLOR 202:PLOT 18,
4:PLOT 18,9:PLOT 18,14:PLOT 18,19:KEYS=4
:X=0:Y=4
11470 RETURN
11500 ? #6; "(+++++++h+++++++";
11510 ? #6; "(+++++++h+++++++";
11520 ? #6; "AAAAAAAAAAAAAAAA&+++";
11530 ? #6; " h (+++";
11540 ? #6; " h &+++";
11550 ? #6; " h (+++";
11560 ? #6; "(+++++++h+++++++";
11570 ? #6; "(+++++++h+++++++";
11580 ? #6; "++#AAAAAAAAAAAAAAAA";
11590 ? #6; "++' h ";
11600 ? #6; "++% h ";
11610 ? #6; "++' h ";
11620 ? #6; "(+++++++h+++++++";
11630 ? #6; "(+++++++h+++++++";
11640 ? #6; "AAAAAAAAAAAAAAAA&+++";
11650 ? #6; " h (+++";
11660 ? #6; " h $+++";
11670 ? #6; " h (+++";
11680 ? #6; "(+++++++h+++++++";
11690 ? #6; "AAAAAAAAAAAAAAAA";
11700 COLOR 202:PLOT 0,3:PLOT 19,3:PL
OT 0,9:PLOT 15,9:PLOT 3,15:PLOT
19,15:PLOT
15,21:X=0:Y=21:KEYS=7
11710 SETCOLOR 0,13,2:SETCOLOR 1,12,4
11720 RETURN
11750 ? #6; "(!!!!!!|' (+G'";
11760 ? #6; "!!!!#D&!!! |!%";
11770 ? #6; "!!!# &!! |!";
11780 ? #6; "!!!# $! (D |";
11790 ? #6; "!!! | |!";
11800 ? #6; "!!!!!!| |!";
11810 ? #6; "!!!!!!| (|!";
11820 ? #6; "!!!#DEF&!!! !E |";
11830 ? #6; "!!! !E! |!";
11840 ? #6; "!!! |! (<|!";
11850 ? #6; "!!! Q! !E) &|";
11860 ? #6; "&!!!!!!% (<|!";
11870 ? #6; " DDEDEEEE !D |";
11880 ? #6; " (<|!";
11890 ? #6; " (<|E |";
11900 ? #6; " (<|F |";
11910 ? #6; "(%DDDDDDDDAD |";
11920 ? #6; "!' (<|";
11930 ? #6; "$!*!!!!!!|!%";
11940 ? #6; " DDDDDDDDDDDDDDDDD ";
11950 COLOR 202:PLOT 5,8:X=1:Y=19:BX=7:
BY=14:BF=1:KEYS=1
11960 SETCOLOR 0,0,8
11970 RETURN
12000 ? #6; "(!!+!!!!|CC!!!!+!!";
12010 ? #6; "!FEA$!!!!!!|!#REF!";
12020 ? #6; "! $!!!!!!|!";
12030 ? #6; "!A' Fh!!!!|E (<A!";
12040 ? #6; "! !' h!!!!|h (<|!";
12050 ? #6; "!+!!!!|h!!!!|h<!!!!+!";
12060 ? #6; "!A!#DFh!!!!|hDD$!A!";
12070 ? #6; "! !' h!!!!|h |!";
12080 ? #6; "! !' hK!!!!|h (<|!";
12090 ? #6; "!+!!!!|h!!!!|h<!!!!+!";
12100 ? #6; "!A!#Dh!!!!|hhd&!A!";
12110 ? #6; "! !' hhh!!!!|hhh |!";
12120 ? #6; "! !' hhh!!!!|hhh (<|!";
12130 ? #6; "!+!!!!|hhh!!!!|hhh<!!!!+!";
12140 ? #6; "!A!#FhKh!!!!|hKhF&!A!";
12150 ? #6; "! !' hhh!!!!|hhh (<|!";
12160 ? #6; "! ! !' hhh!!!!|hhh<!!!";
12170 ? #6; "! ! ! ! ! ! ! ! ! ! ! ! ! ! !";
12180 ? #6; "!*!!!!!!|!";
12190 ? #6; "FDEEDFEE FDFDFDDF";
12200 COLOR 202:PLOT 15,16:PLOT 4,16:P
LOT 4,12:PLOT 15,12:X=4:Y=19:KEYS=4
12210 SETCOLOR 0,2,2:SETCOLOR 1,0,4
12220 RETURN
12250 ? #6; " ";
12260 ? #6; " DEFDEFDE EFD DFD E"

```



```

0
110 GOSUB 950:GOSUB 950:PLOT SX,SY:NEXT
I:COLOR 36:FOR I=1 TO 4:GOSUB 1350:LOOP(
I)=0:NEXT I
120 FOR I=1 TO 4:SOUND 0,20-5*(I=4),10,1
0+2*(I=4):FOR J=1 TO 150:NEXT J
130 SOUND 0,0,0,0:FOR J=1 TO 150:NEXT J:
NEXT I:K=0
140 FOR I=1 TO 40:NEXT I
150 A=STICK(0):TIME=TIME-1
160 POSITION 4,0:PRINT #6;TIME;CHR$(129)
:POSITION 7,11:PRINT #6;SCORE
170 IF TIME<0 THEN 1000
180 IF A=7 THEN X1=1:Y1=0:GOTO 230
190 IF A=11 THEN X1=-1:Y1=0:GOTO 230
200 IF A=13 THEN Y1=1:X1=0:GOTO 230
210 IF A=14 THEN Y1=-1:X1=0:GOTO 230
220 GOTO 600
230 SOUND 0,71,2,15:TIME=TIME-1:LOCATE X
+X1,Y+Y1,J:SOUND 0,0,0,0
240 IF J=32 THEN 280
250 IF J=34 THEN SOUND 2,30,10,10:SCORE=
SCORE+50:GOTO 280
260 IF J<>36 THEN 290
270 GOTO 1000
280 COLOR 32:PLOT X,Y:X=X+X1:Y=Y+Y1
290 COLOR A:PLOT X,Y:A1=A:SOUND 2,0,0,0:
IF STRIG(0) THEN 600
300 IF J=162 OR J=29 THEN STEIN=J:FREI=2

:TIME=TIME-2:GOTO 400
310 GOTO 600
400 SOUND 2,0,0,0:LOCATE X+FREI*X1,Y+FRE
I*Y1,J:IF J=36 THEN SOUND 2,30,10,15:SCO
RE=SCORE+100:GOTO 410
405 IF J<>32 THEN 500
410 SOUND 1,20,100,70:SCORE=SCORE+5:COLO
R 32:PLOT X+FREI*X1-X1,Y+FREI*Y1-Y1
420 COLOR STEIN:PLOT X+FREI*X1,Y+FREI*Y1
430 FREI=FREI+1:GOTO 400
440 COLOR 163:GOSUB 450:COLOR 165:GOSUB
450:COLOR 32:PLOT X+X1,Y+Y1:SCORE=SCORE+
5:K=0:SOUND 2,0,0,0:GOTO 600
450 PLOT X+X1,Y+Y1:FOR I=1 TO 5:SOUND 2,
K,0,15:K=K+1:NEXT I:RETURN
460 J=4:FOR I=1 TO 20
470 POKE CH+243,103+J:SOUND 0,90+J,10,10
480 FOR K=1 TO 20:NEXT K:SOUND 0,0,0,0
490 J=-J:NEXT I:RETURN
500 SOUND 1,0,0,0:IF STEIN=162 AND FREI=
2 THEN 440
510 IF STEIN=162 THEN 600
520 SX=X+FREI*X1-X1:SY=Y+FREI*Y1-Y1
530 FOR K=1 TO 2:DIF=-DIF
540 LOCATE SX,SY+DIF,J:IF J=29 OR J=30 T
HEN S1X=SX:S1Y=SY+DIF:GOSUB 570

```

```

550 LOCATE SX+DIF,SY,J:IF J=29 OR J=30 T
HEN S1X=SX+DIF:S1Y=SY:GOSUB 570
560 SOUND 1,0,0,0:NEXT K:GOTO 600
570 SOUND 1,15,10,15:COLOR 30:PLOT SX,SY
:PLOT S1X,S1Y
580 IF BLOCK=0 THEN BLOCK=1:SCORE=SCORE+
200:RETURN
590 POP :GOTO 1380
600 WAIT=WAIT+1:IF WAIT<PAUSE THEN 140
610 WAIT=0:FOR I=1 TO 4:LOCATE FX(I),FY(
I),J:IF J<>36 THEN GOSUB 730:NEXT I:GOTO
150
620 X2=-SGN(FX(I)-X):Y2=-SGN(FY(I)-Y)
630 IF Y2<>0 AND X2<>0 THEN GOSUB 710
640 LOCATE FX(I)+X2,FY(I)+Y2,J
650 IF J=162 THEN 770
660 IF J=29 OR J=30 OR J=36 OR J=34 THEN
NEXT I:GOTO 150
670 COLOR 32:PLOT FX(I),FY(I)
680 FX(I)=FX(I)+X2:FY(I)=FY(I)+Y2
690 COLOR 36:PLOT FX(I),FY(I):IF J=A1 TH
EN POP :X1=0:Y1=0:GOTO 1000
700 NEXT I:GOTO 150
710 IF RND(0)>.5 THEN X2=0:RETURN
720 Y2=0:RETURN
730 IF J=34 THEN 760
740 LOOP(I)=LOOP(I)+1:IF LOOP(I)<3 THEN
RETURN
750 COLOR 34:GOSUB 1350:LOOP(I)=0:RETURN

760 COLOR 36:PLOT FX(I),FY(I):RETURN
770 SOUND 0,5,10,10:COLOR 32
780 PLOT FX(I)+X2,FY(I)+Y2:SOUND 0,0,0,0
:NEXT I:GOTO 150
900 COLOR 129:PLOT 2,0:DRAWTO 17,0
910 COLOR 136:PLOT 18,0:COLOR 137:DRAWTO
18,9
920 COLOR 138:PLOT 18,10:COLOR 129:DRAWT
O 2,10
930 COLOR 140:PLOT 1,10:COLOR 137:DRAWTO
1,1
940 COLOR 143:PLOT 1,0:RETURN
950 DIF=-DIF
960 LOCATE SX,SY+DIF,J:IF J=29 THEN POP
:GOTO 100
970 LOCATE SX+DIF,SY,J:IF J=29 THEN POP
:GOTO 100
980 RETURN
1000 COLOR 129:PLOT 4,0:DRAWTO 8,0
1010 COLOR 32:PLOT X,Y:COLOR 3:GOSUB 450
:COLOR 5:GOSUB 450:COLOR 32:GOSUB 450:SO
UND 2,0,0,0
1020 ROUND=ROUND-1:LEBEN=LEBEN-1:BLOCK=0
:A1=7:GOSUB 1400:K=0
1030 IF LEBEN=0 THEN 1500
1040 GOTO 40

```



```

1100 RESTORE 1140
1110 FOR I=1 TO 1023:POKE CH+I,PEEK(5734
4+I):NEXT I
1120 READ A:IF A=-1 THEN RETURN
1130 FOR I=0 TO 7:READ B:POKE CH+I+A*8,B
*NEXT I:GOTO 1120
1140 DATA 1,0,255,255,255,255,0,0
1150 DATA 2,62,127,127,127,127,127,62,0
1160 DATA 7,0,56,109,127,127,125,56,0
1170 DATA 11,0,14,87,127,127,95,14,0
1180 DATA 13,60,126,126,118,60,24,60,0
1190 DATA 14,60,24,60,118,126,126,60,0
1200 DATA 4,60,126,90,126,90,102,60,0
1210 DATA 29,62,127,119,107,119,127,62,0
1220 DATA 30,62,127,119,99,119,127,62,0
1230 DATA 3,42,85,54,73,54,85,42,0
1240 DATA 5,0,0,20,8,20,0,0,0
1250 DATA 8,0,252,252,252,252,252,60,60
1260 DATA 9,60,60,60,60,60,60,60,60
1270 DATA 10,60,252,252,252,252,252,0,0
1280 DATA 12,60,63,63,63,63,63,0,0
1290 DATA 15,0,63,63,63,63,63,60,60
1300 DATA 6,60,66,153,161,161,153,66,60
1309 DATA -1
1350 FX(I)=INT(RND(0)*12)+6:FY(I)=INT(RN
D(0)*9)+1:LOCATE FX(I),FY(I),J
1360 IF J<>32 THEN 1350
1370 PLOT FX(I),FY(I):RETURN
1380 SOUND 1,0,0,0:SCORE=SCORE+2000+TIME
+ROUND*100:A1=7:BLOCK=0:PAUSE=PAUSE-1
1390 POSITION 1,11:PRINT #6;"SCORE:";SCO
RE:COLOR 129:PLOT 4,0:DRAWTO 8,0:GOSUB 4.
60:GOSUB 1400:GOTO 40
1400 IF SCORE<25000*BONUS THEN FOR I=1 T
O 500:NEXT I:RETURN
1410 POSITION 4,5:PRINT #6;" BONUSLEBEN
"
1420 FOR I=255 TO 0 STEP -1:SOUND 0,I,10
,10:POKE 712,I:NEXT I
1430 LEBEN=LEBEN+1:BONUS=BONUS+1:A1=7:RE
TURN
1500 POSITION 4,2:PRINT #6;" game over
"
1510 IF SCORE>TOP THEN TOP=SCORE
1520 POSITION 3,4:PRINT #6;" SCORE:";SCO
RE;" "
1530 POSITION 4,6:PRINT #6;" HIGH:";TOP;
" ":POSITION 4,8:PRINT #6;" START? "
1540 COLOR 32:PLOT 1,11:DRAWTO 16,11:COL
OR 38:PLOT 5,11:PRINT #6;" 1984"
1550 IF STRIG(0) AND PEEK(53279)<>6 THEN
1550
1560 SCORE=0:ROUND=0:PAUSE=14:A1=7:LEBEN
=3:BONUS=1:GOTO 40
1600 GRAPHICS 18:POKE 708,15

```

```

1610 POSITION 6,3:PRINT #6;"DIAMONDS"
1620 ZL=PEEK(560)+PEEK(561)*256:POKE ZL+
11,6:POKE ZL+13,6:POKE ZL+16,6
1630 POSITION 9,6:PRINT #6;"by":POSITION
3,8:PRINT #6;"frank priggert":GOSUB 200
0
1640 CH=(PEEK(106)-8)*256:GOSUB 1100
1650 POSITION 2,11:PRINT #6;"spielregeln
j";CHR$(15);"n"
1660 POKE 764,255:OPEN #1,4,0,"K:":GET #
1,K:IF K=74 THEN 1680
1670 RETURN
1680 GRAPHICS 0:POKE 82,0:POKE 752,1:POK
E 710,255:POKE 712,255:POKE 709,201:ZL=P
EEK(560)+PEEK(561)*256+6
1690 POKE ZL-3,70:POKE ZL,6
1700 POSITION 6,0:PRINT #6;"diamonds"
1710 PRINT " Ziel des Spiels ist es,3 Di
amanten zu- sammenzufuegen."
1720 PRINT "Um deinen Feinden zu entkomm
en,bewirfst du sie mit den Diamanten und
Kloetzen."
1730 PRINT "Das geht aber nur,wenn dahin
ter kein an-derer steht,sonst explodiert
der Klotz."
1740 PRINT "Hiefuer drueckst du den Feue
rknopf."
1750 PRINT "Sind einmal Diamanten zusamm
engeschoeben,lassen sie sich aber nicht m
ehr trennen!";
1760 PRINT "Da deine Gegner Geister sind
,kommen sie bald wieder.Jetzt musst du n
ur ueber sie";
1770 PRINT "hinwegrennen,dann passiert d
asselbe von neuem."
1780 PRINT "Du besitzt 3 Leben und bekom
mst jede 25000 Punkte ein neues dazu.
"
1790 PRINT "Oben links steht deine Soliz
eit daneben deine uebriggebliebenen Lebe
n.Unten die"
1800 PRINT "Punktzahl und die Runde in d
er du bist."
1810 POSITION 7,22:PRINT "DRUECKE IRGRND
EINE TASTE !":GET #1,K:POKE 764,255
1820 GRAPHICS 18:POSITION 4,0:PRINT #6;"
ERLAUTERUNG":POKE 756,CH/256:POKE 710,1
96:POKE 709,27
1830 RESTORE 1840:FOR I=1 TO 8:READ A,B,
C:COLOR A:PLOT B,C:NEXT I:FOR I=1 TO 300
*NEXT I
1840 DATA 7,5,5,162,6,5,29,7,5,36,12,5,2
9,14,5,34,7,6,29,8,6,162,14,6,163,165,32
1850 FOR I=1 TO 3:READ A:COLOR A:PLOT 6,
5:FOR J=1 TO 4:NEXT J:NEXT I

```



```

1860 POSITION 6,10:PRINT #6;"5 PUNKTE":F
OR I=1 TO 350:NEXT I:COLOR 32:PLOT 6,10:
DRAWTO 14,10
1870 COLOR 32:PLOT 5,5:COLOR 7:PLOT 6,5:
FOR I=1 TO 100:NEXT I
1880 COLOR 32 PLOT 7,5:COLOR 29:PLOT 8,5
'POSITION 6,10:PRINT #6;"5 PUNKTE":FOR I
=1 TO 100:NEXT I
1890 FOR I=8 TO 12:COLOR 32:PLOT I,5:COL
OR 29:PLOT I+1,5:IF I<>11 THEN 1910
1900 POSITION 5,10:PRINT #6;"100 PUNKTE"
:FOR J=1 TO 100:NEXT J
1910 IF I<12 THEN FOR J=1 TO 5:NEXT J:NE
XT I
1920 COLOR 30:PLOT 13,5:PLOT 14,5:POSITI
ON 5,10:PRINT #6;"200":FOR I=1 TO 350:NE
XT I:COLOR 32:PLOT 5,10
1930 DRAWTO 15,10:PLOT 6,5:COLOR 13:PLOT
6,6:FOR J=1 TO 40:NEXT J:COLOR 32:PLOT.
6,6:COLOR 7:PLOT 7,6
1940 POSITION 5,10:PRINT #6;"50 PUNKTE":
FOR I=1 TO 350:NEXT I:COLOR 32:PLOT 5,10
'DRAWTO 15,10
1950 FOR I=8 TO 12:FOR J=1 TO 5:NEXT J:P

```

```

LOT I,6:COLOR 29:PLOT I+1,6:COLOR 32:NEX
T I
1960 COLOR 30:PLOT 13,6:POSITION 4,10:PR
INT #6;"2000 PUNKTE":FOR I=1 TO 350:NEXT
I
1970 POSITION 4,10:PRINT #6;" start?
"
1980 IF STRIG(0) AND PEEK(53279)<>6 THEN
1980
1990 RETURN
2000 PM=PEEK(106)-16:POKE 54279,PM:PMB=P
M*256:POKE 53277,3
2010 FOR I=PMB+512 TO PMB+639:POKE I,0:N
EXT I:RESTORE 2040
2020 FOR I=PMB+512+57 TO PMB+517+57:READ
A:POKE I,A:NEXT I
2030 POKE 53248,0:POKE 623,1:RETURN
2040 DATA 56,109,127,127,125,56
2050 POKE 559,46:FOR I=40 TO 64:SOUND 0,
135-I*2,14,10:POKE 53248,I:NEXT I:SOUND
0,0,0,0
2060 POKE 53248,0:COLOR 137:PLOT 1,5:RET
URN

```

Top- programm

Spinnen

Das Top-Programm „Spinnen“ ist ein
schnelles MC-Spiel für den ZX-81/16k.

Der eigentliche Teil des Programmes läuft in Maschinensprache ab, lediglich das Steuerprogramm, mit dem einige Parameter eingestellt werden können, ist in Basic verfaßt.

Sie überfliegen mit Ihrem Expeditionsschiff eine Wiese und gelangen so zum Eingang einer Höhle, die übrigens jedesmal neu und zufallsgesteuert erzeugt wird. Ihr Schiff läßt sich mit den Tasten >6< (nach unten) und >7< (nach oben) bewegen. Durch Drücken der Taste >0< wird die Bordkanone ausgelöst, die sich, durch Halten der

>0<-Taste, auch mit Dauerfeuer betreiben läßt. Allerdings nimmt pro Betätigung dieser Waffe auch Ihr Energiestand ab. Ist dieser bei >0< angeht, versagt Ihnen die Kanone ihren Dienst. Deshalb empfiehlt es sich, den ständig lauernnden Gefahren auch einmal durch Flugmanöver auszuweichen, denn diese verbrauchen keine Energie. Das Gestein der Höhle läßt sich übrigens zerschießen. Es ist also möglich, soweit der Energiebestand das zuläßt, sich seinen eigenen Weg zu bahnen. Im Gegensatz zu den Gesteinsbrocken, erhöhen die Köpfe der Spinnen, die überall herumhängen, Ihre Energiereserven. Die Spinne wird bei einem Abschub außerdem vollständig eliminiert, d. h. auch der Fa-

den, an dem der Spinnenkopf hängt, verschwindet. Die andere Möglichkeit, eine Spinne zu beseitigen, ist die, den Gesteinsbrocken, an dem sie hängt, abzuschießen. Allerdings gibt diese Alternative keine Punkte. Eine weitere mögliche Gefahr, die Sie zum Scheitern verurteilen könnte, sind die „giftigen Blicke“, die die Spinnen von Zeit zu Zeit aussenden. Diese Blicke durchdringen alles und haben es auf Ihr Schiff abgesehen, das bei einer Kollision sofort den Geist aufgibt. Durch ein Zusammenprallen mit einem Ihrer Geschosse wird jedoch auch der „Blick“ ausgelöscht. Sollten Sie diesen 1. Teil überstanden und nach ca. 1200 km Flug noch am Leben sein, geht es dann automatisch in den 2. Teil:

Ein „Sulp“ taucht auf, die Höhle wird zu einem Tunnel. Der „Sulp“ bewegt sich zufällig zwischen den Begrenzungen des Tunnels rauf und runter und dabei regelmäßig vorwärts. Ist er Ihrem Schiff nahegekommen, fängt er an zu blinken und wird langsamer. Sie müssen auf jeden Fall eine Kollision mit ihm vermeiden, was sehr von Ihrem Geschick abhängt. Um Ihnen das Überleben so schwer wie möglich zu machen, schießt der „Sulp“. Diese Schüsse sind aber gleichfalls abschießbar. Nicht vernichtbar und deshalb eine große Gefahr sind die Hindernisse, die der „Sulp“ aufbaut. Es

muß daher auf jeden Fall verhindert werden, daß dieser alles zumauert und ein Weiterkommen ausgeschlossen wird. Trifft man einen „Sulp“, gibt dies 40 Punkte auf das Energiekonto. Ein anderer Weg am „Sulp“ vorbeizukommen ist durch die Möglichkeit gegeben, ihm einfach auszuweichen. Ist er verschwunden, erscheint in der 2. Situation der „Morg“:

Dieser gibt bei Abschluß 90 Punkte. Es gilt, in ihn einzudringen und sein Hirn zu zerstören. Vorher muß man aber die Stellen ausmachen, an denen er empfindlich ist und die sich folglich durchschießen lassen. Der „Morg“

taucht wie gesagt nur auf, wenn der „Sulp“ nicht zerstört wurde.

3. Teil: Er entspricht weitgehend dem 1. Teil und unterscheidet sich lediglich durch seine individuelle Geschwindigkeit und Beschleunigung. Nach 6000 km Flug ist man dann am Ziel angelangt, der „Gruff der Spinne“. Verliert man ein Schiff, wird die bis dahin erreichte Energie auf das „Punkte-Konto“ übertragen. Sind am Schluß noch Schiffe vorhanden, erhält man für jedes Schiff die jeweilige Energiepauschale. In die High-Score-Wertung kommt nur derjenige, der bis zur Gruff durchgehalten hat.

Jedem Spieler ist es selbst überlassen, sich sein eigenes Spiel mit seinen speziellen Werten zurechtzulegen. Es empfiehlt sich aber fürs erste, einmal die vorgeschlagenen Werte einzugeben, um Erfahrungen zu sammeln. Im „Standard“-Modus ist es dann möglich, die Anzahl der zur Verfügung stehenden Schiffe zu bestimmen sowie den Schwierigkeitsgrad zu wählen. Dann werden die dem Schwierigkeitsgrad entsprechenden Werte eingepoket, und das Spiel kann beginnen. Existiert ein High-Score, wird dieser in der untersten Zeile ausgegeben.

*** LISTING 1 ***

```

0 REM          1820 ZEICHEN
10 FOR N=16514 TO 18265
20 INPUT A
30 POKE N,A
40 SCROLL
50 PRINT N,A
60 NEXT N
    
```

*** MC-LISTING I ***

16514	237	105	00	107	174	115
16515						115
16516	107	100				115
16517			11	14		115
16518			11	14		115
16519			11	14		115
16520			11	14		115
16521			11	14		115
16522			11	14		115
16523			11	14		115
16524	15	00	00	00	00	115
16525						115
16526						115
16527						115
16528						115
16529						115
16530						115
16531						115
16532						115
16533						115
16534						115
16535						115
16536						115
16537						115
16538						115
16539						115
16540						115
16541						115
16542						115
16543						115
16544						115
16545						115
16546						115
16547						115
16548						115
16549						115
16550						115
16551						115
16552						115
16553						115
16554						115
16555						115
16556						115
16557						115
16558						115
16559						115
16560						115
16561						115
16562						115
16563						115
16564						115
16565						115
16566						115
16567						115
16568						115
16569						115
16570						115
16571						115
16572						115
16573						115
16574						115
16575						115
16576						115
16577						115
16578						115
16579						115
16580						115
16581						115
16582						115
16583						115
16584						115
16585						115
16586						115
16587						115
16588						115
16589						115
16590						115
16591						115
16592						115
16593						115
16594						115
16595						115
16596						115
16597						115
16598						115
16599						115
16600						115

Zuerst muß eine Rem-Zeile der Länge 1820 Zeichen erzeugt werden. Die erste und bequemste Möglichkeit, eine Rem-Zeile dieser Länge zu erzeugen, besteht in der Verwendung des Rem-Zeilen-Generators aus Heft 5/84. Sollten Sie auf diese Möglichkeit nicht zurückgreifen können, gehen Sie bitte folgendermaßen vor:

- Geben Sie die Rem-Zeile 1 mit 160 Zeichen ein.
- EDITieren Sie diese Zeile und erzeugen Sie dann noch 10 weitere gleichartige Rem-Zeilen (Zeile 2-11)
- POKE 16511,30 NEWLINE
- POKE 16512,7 NEWLINE
- POKE 16510,0 NEWLINE

Achtung: Führen Sie diese Anweisungen direkt hintereinander aus, da sonst das System abstürzen könnte.

- LIST NEWLINE

Als nächster Schritt geben Sie das Listing 1 ein, das zur Eingabe der Maschinensprache-Zahlen dient. Nach der Eingabe starten Sie das Programm mit RUN; die Eingabe der Zahlen aus MC-Listing I kann beginnen (von links nach rechts). Da es in Anbetracht der Menge der Zahlen wahrscheinlich ist, daß sich der eine oder andere Fehler einschleicht, wurden die Programme 2 und 3 beigelegt. Listing 2 berechnet die Prüfsummen, Listing 3 schreibt die Speicherinhalte auf den Bildschirm.

Wenn Sie ein fehlerfreies Maschinenprogramm vorliegen haben, löschen Sie die Zeilen einzeln und geben das Basic-Hauptlisting ein. Nachdem Sie auch dies erledigt haben, starten Sie das Programm mit RUN und wählen „NEU EINSTELLEN“.

Das Eingeben neuer Werte:

- Geschwindigkeit Teil 1-3: Je größer die Zahl, desto langsamer. Die Gesamtgeschwindigkeit hängt jedoch auch noch von Punkt 4 und 5 ab.
- Beschleunigung Teil 1-3: Je größer die Beschleunigung, desto schneller nimmt die Geschwindigkeit des jeweiligen Teils zu.
- Länge Morg: Der Wert gibt an, wie weit die Pfosten des Morg auseinander sind. Dabei gilt: Je weiter, desto leichter.
- Beweglichkeit: Sie gibt an, um wieviel das Schiff nach unten bzw. nach oben bewegt werden kann, bevor es sich einmal vorwärtsbewegt.
- Schußweite pro Bewegung: Der Wert gibt an, um wieviel der Schuß vorankommt, bevor eine Bewegung des Schiffs möglich ist.


```

210 PRINT AT 7,0; "* SCHWIERIGKEITSGRAD"
220 GOTO 130
230 PRINT SG
240 FOR N=1 TO 11
250 LET C(N)=A(SG,N)
260 NEXT N
270 GOTO 3000
400 REM ANZAHL SCHIFFE
410 CLS
415 PRINT B#
420 PRINT AT 5,0; "* WIEVIELE SCHIFFE (1-9) : "
430 LET AS=CODE INKEY$-28
440 IF AS>0 AND AS<10 THEN GOTO 470
450 PRINT AT 5,0; "* ANZEHLE GEWICHTSSTUFE (1-9) : "
460 GOTO 420
470 PRINT AS
480 LET S=AS-2
490 POKE 16539,S
500 IF AS=1 THEN POKE 16539,255
510 RETURN
520 REM SPEICHERN ?
530 CLS
540 PRINT B#
550 PRINT AT 5,0; "WOLLEN SIE : "
560 PRINT "A) DIE WERTE UNTER EINEM BEWERTEN ? "
570 PRINT "B) NACH EINGABE ZUERST SPIELEN ? "
580 IF INKEY$<>"A" THEN GOTO 58
590 GOSUB 2000
600 GOTO 600
610 IF INKEY$<>"B" THEN GOTO 57
620 LET C=0
630 REM NEU EINSTELLEN
640 CLS
650 PRINT B#
660 PRINT " * NEU EINSTELLEN : "
670 PRINT
680 FOR N=1 TO 3
690 PRINT " * GESCHW. TEIL (N) : " ; (" (" AND C=1); ((STR$ A(SG,N)) AND C=1); (" (" AND C=1);
700 INPUT C(N)
710 LET C(N)=INT C(N)
720 IF C(N)<0 OR C(N)>255 THEN GOTO 670
730 PRINT C(N)
740 NEXT N
750 PRINT
760 FOR N=1 TO 3
770 PRINT " * BESCHLEUNIG. TEIL (N) : " ; (" (" AND C=1); ((STR$ A(SG,N+3)) AND C=1); (" (" AND C=1);
780 INPUT C(3+N)
790 LET C(3+N)=INT C(3+N)
800 IF C(3+N)<0 OR C(3+N)>255 THEN GOTO 730
810 PRINT C(3+N)
820 NEXT N
830 PRINT
840 "* LAENGE MORG (5-9) : " ; (" (" AND C=1); ((STR$ A(5+7)) AND C=1); (" (" AND C=1);
850 INPUT C(7)
860 LET C(7)=INT C(7)
870 IF C(7)<5 OR C(7)>9 THEN GOTO 780
880 PRINT C(7)
890 PRINT " * BEWEGLICHKEIT (2-5) : " ; (" (" AND C=1); ((STR$ A(5+9)) AND C=1); (" (" AND C=1);
900 INPUT C(8)
910 LET C(8)=INT C(8)
920 IF C(8)<2 OR C(8)>5 THEN GOTO 830
930 PRINT C(8)
940 PRINT " * SCHUSSWEITE PRO BEWEGUNG (2-5) : " ; (" (" AND C=1); ((STR$ A(5+9)) AND C=1); (" (" AND C=1);
950 INPUT C(9)
960 LET C(9)=INT C(9)

```

```

910 IF C(9)<2 OR C(9)>5 THEN GOTO 830
920 PRINT C(9)
930 PRINT " * SCHUSSLAENGE (2-5) : " ; (" (" AND C=1); ((STR$ A(5+10)) AND C=1); (" (" AND C=1);
940 INPUT C(10)
950 LET C(10)=INT C(10)
960 IF C(10)<2 OR C(10)>25 THEN GOTO 830
970 LET C(10)=(INT (C(10)/C(9)+.8))*C(9)
980 PRINT C(10)
990 PRINT " * ENERGIE/SCHIFF (10) : " ; (" (" AND C=1); ((STR$ A(5+11)) AND C=1); (" (" AND C=1);
1000 INPUT C(11)
1010 LET C(11)=INT C(11)
1020 IF C(11)<0 OR C(11)>100 THEN GOTO 1000
1030 PRINT C(11)
1040 GOSUB 400
1050 IF C=1 THEN GOTO 2140
1070 GOTO 3000
2000 REM WERTE SPEICHERN
2010 CLS
2020 PRINT B#
2030 PRINT AT 5,0; "* WERTE SPEICHERN"
2040 PRINT AT 7,0; "* UNTER SCHWIERIGKEITSGRAD : "
2045 LET SG=CODE INKEY$-28
2047 IF SG>0 AND SG<6 THEN GOTO 2120
2050 PRINT "1) SEHR LEICHT"
2060 PRINT "2) LEICHT"
2070 PRINT "3) MITTEL"
2080 PRINT "4) SCHWER"
2090 PRINT "5) SEHR SCHWER"
2100 PRINT AT 7,0; "* UNTER SCHWIERIGKEITSGRAD : "
2110 GOTO 2040
2120 PRINT SG
2125 IF C=0 THEN GOTO 2140
2130 LET C=1
2135 RETURN
2140 FOR N=1 TO 11
2150 LET A(SG,N)=C(N)
2160 NEXT N
2170 CLS
2180 PRINT B#
2190 PRINT AT 5,0; "* SAVEN ? (J/N) : "
2200 IF INKEY$="J" THEN GOTO 2240
2210 IF INKEY$="N" AND C=0 THEN GOTO 3710
2215 IF INKEY$="N" AND C=1 THEN GOTO 3000
2220 PRINT AT 5,0; "* SAVEN ? (J/N) : "
2230 GOTO 2190
2240 SAVE "SPINNE"
2250 GOTO 35
3000 REM SPIELSTART
3005 FOR N=1 TO 10
3010 POKE B(N),C(N)
3015 NEXT N
3020 FOR N=11 TO 15
3025 POKE B(N),0
3030 NEXT N
3031 POKE 16539,S
3032 IF AS=2 THEN POKE 16539,255
3035 POKE 16418,0
3036 POKE 16611,C(11)
3037 POKE 17006,C(7)
3038 LET E#="STR$ C(11)"
3040 CLS
3050 PRINT "
3060 PRINT AT 12,0; "
3070 PRINT AT 15,0; "
3080 PRINT "
3090 PRINT

```



```

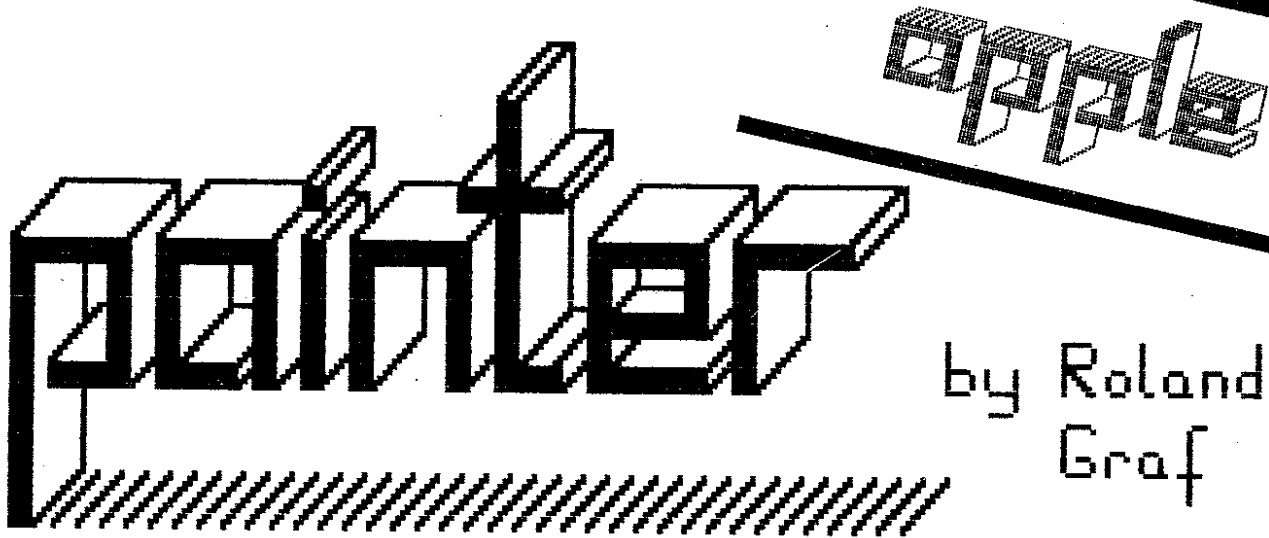
3100 PRINT "ENERGIE 00"; ("0" AND
LEN E#=1); E#; "0 KM 00000"
3110 PRINT "REST "" "" "; S+1; "
SCORE 00000"
3120 IF H#="00000" THEN GOTO 314
0
3130 PRINT ", * HIGH SCORE VON "
N#; " : "; H#
3140 PRINT AT 10,0; "TASTE BITTE"
3150 IF INKEY#="" THEN GOTO 3150
3160 PRINT AT 10,0; "
3170 LET L=USR 16540
3180 PRINT AT 21,11; ("0 " AND P
PEEK 16539-255)
3190 LET SC=PEEK 16559+255*PEEK
16560+PEEK 16569+255*PEEK 16530
3192 LET SC=SC+(11)*(PEEK 1653
9+1) AND PEEK 16539<>255)
3195 LET S#=(("000" AND SC<10)+(
"0" AND SC<100)+("0" AND SC<1000
)+5)*S# SC+0
3200 PRINT AT 21,10; S#
3205 IF PEEK 16537+255*PEEK 1653
0<>800 THEN GOTO 3500
3204 PRINT AT 20,10; "XNF 00000"
3205 POKE 18079,20
3207 POKE 18104,201
3208 LET L=USR 18078
3209 POKE 18079,10
3210 POKE 18104,50
3214 GOSUB 4000
3215 REM HIGH-SCORE ?
3220 IF VAL S#<VAL H# THEN GOTO
3500
3230 LET H#=S#
3240 CLS
3250 PRINT B#; "
3260 PRINT " * HERZLICHEN GLUECKW
UNTSCH " "; H#; " PUN
KTE " "; " * DAS IST DER NEUE HIGHS
CORE "
3270 FOR N=1 TO 20
3280 PRINT AT 6,19; "HIGHSCORE"
3290 PRINT AT 6,19; "HIGHSCORE"
3300 NEXT N
3310 PRINT AT 10,0; " * BITTE GEBE
N SIE IHR KUERZEL EIN (1-4) "
3335 LET N#=""
3336 PRINT AT 20,0; "
3337 PRINT AT 15,20; "
3338 PRINT AT 15,0; " * NEUER HI-S
CORE
VON "
3339 FOR N=1 TO 4
3340 IF INKEY#<>" " THEN GOTO 334
3343
3343 IF INKEY#="" THEN GOTO 3343
3345 LET A#=INKEY#
3346 PRINT A#;
3350 LET N#=N#+A#
3370 NEXT N
3380 PRINT AT 20,0; " * O.K. ? (J/
GZ)
3390 IF INKEY#="J" THEN GOTO 350
3400 IF INKEY#="N" THEN GOTO 333
3410 PRINT AT 20,0; " * O.K. ? (J/
GZ) "
3420 GOTO 3380
3500 REM NEUES SPIEL ?
3501 PRINT AT 10,0; "TASTE BITTE"
3502 IF INKEY#="" THEN GOTO 3502
3503 POKE 16410,2
3510 CLS
3520 PRINT B#
3530 PRINT AT 5,0; " * MENUE ."
3540 PRINT ", "1) DASSELBE NOCH E
INMAL "
3550 PRINT ", "2) SPIELART AEWDER
N "
3560 PRINT ", "3) AUFHOEREN "
3570 PRINT ", "4) WERTE SPEICHERN
" AND C=0
3580 IF INKEY#="1" THEN GOTO 300
0
3590 IF INKEY#="2" THEN GOTO 35
3600 IF INKEY#="3" THEN GOTO 999
0

```

```

3610 IF INKEY#="4" AND C=0 THEN
GOTO 3700
3620 PRINT AT 5,0; " * MENUE "
3630 GOTO 3530
3700 GOTO 2000
3710 LET C=2
3720 GOTO 3510
4000 REM ZIEL ERREICHT
4001 PRINT AT 10,0; "TASTE BITTE"
4002 IF INKEY#="" THEN GOTO 4002
4010 LET A#=""
4020 LET A#=A#+ " "
4030 LET A#=A#+ " "
4040 LET A#=A#+ " "
4050 LET A#=A#+ " "
4060 LET A#=A#+ " "
4070 LET A#=A#+ " " *
4080 LET A#=A#+ " "
4090 LET A#=A#+ " "
4100 LET A#=A#+ " "
4110 LET A#=A#+ " "
4120 LET A#=A#+ " "
4130 LET A#=A#+ " " *
4140 LET A#=A#+ " " 0...
4150 LET A#=A#+ " " / 0...
4160 LET A#=A#+ " " / 0..
4170 LET A#=A#+ " " / 0..
4180 LET A#=A#+ " " 00 / 0.
4190 LET A#=A#+ " " 0000 0.
4200 LET A#=A#+ " " 00000 / 0
4210 LET A#=A#+ " " 0000 0
4220 LET A#=A#+ " " 0
0.....0
4230 CLS
4240 PRINT A#
4250 REM LAUFTEXT
4260 LET A#="SUPER...SIE HABEN E
S ALSO GESCHAFFT, "
4270 LET A#=A#+ "UND SIND IN DER
"
4280 LET A#=A#+ "DEN SCHATZ, DEN
SIE LINKS SEHEN, QUERFEN SIE MITN
EHMEN. "
4290 LET A#=A#+ "...ALLERDINGS NUR
SOVIEL, WIE SIE MIT IHRER GEWONN
ENEN ENERGIE "
4300 LET A#=A#+ "TRANSPORTIEREN K
OENNEN "
4310 LET A#=A#+ "...DAS SIND IMME
RHIN "+S#+ " PUNKTE FUER IHR ABEN
TEURERKONTO. "
4320 LET A#=A#+ "****HERZLICHEN G
LUECKWUNSCH****"
4330 LET A#=A#+ "
TASTE BITTE "
4340 FOR N=1 TO LEN A#
4350 LET L=USR 17274
4360 PRINT AT 23,31; A#(N)
4365 FOR X=1 TO 1
4366 NEXT X
4370 NEXT N
4380 FOR N=1 TO 19
4390 LET L=USR 17274
4400 FOR X=1 TO 1
4410 NEXT X
4420 NEXT N
4430 IF INKEY#="" THEN GOTO 4430
4440 RETURN

```



by Roland Graf

PAINTER ist ein Zeichenprogramm zum Erstellen von hochaufgelösten grafischen Darstellungen auf dem APPLE II-Computer.

Es unterstützt den Benutzer durch viele Spezialfunktionen. Wenn man sich mit diesem Programm eingearbeitet hat, kann man die tollsten Grafiken erstellen. Perspektivisches Zeichnen ist auch möglich. Man muß eben alles mal ausprobieren, um zu sehen, was das Programm leistet.

Eine in den Programminstruktionen nicht erwähnte Funktion ist die „HELP“-Funktion. Sie wird durch Drücken der Taste „H“ angesprochen und setzt dann Orientierungspunkte auf den Bildschirm, die durch nochmaliges Drücken der Taste „H“ wieder gelöscht werden.

Vor Eingabe des Basic-Hauptprogramms muß noch folgender Hexdump eingegeben werden:

CALL-151

* 300: 01 00 04 00 04 00 00 00

Dann wird er auf Disk gespeichert:

* BSAVE PAINTER OBJ., A\$300, L\$07

Danach wird wieder mit CTRL-C/RETURN ins Basic zurückgeschaltet und das Basic-Hauptprogramm eingegeben.

```

1  REM *****
2  REM *
3  REM * - PAINTER - *
4  REM *
5  REM * (C) 1984 *
6  REM *
7  REM * WRITTEN BY *
8  REM * ROLAND GRAF *
9  REM *
10 REM *****
11 POKE - 16302,0: HGR : HGR2 : HOME :
TEXT : POKE 232,0: POKE 233,3

```



```
12 PRINT CHR$ (4);"BLOAD PAINTER OBJ."
15 GOTO 500
20 ROT= 0: SCALE= 1: HGR2
30 X = 140:Y = 96:F = 1:R = 0:S = 1
40 ROT= R: SCALE= S
50 XDRAW 1 AT X,Y:XX = X:YY = Y
60 IF PDL (0) > 200 AND X < 279 - F THEN X = X + F
70 IF PDL (0) < 50 AND X > F THEN X = X - F
80 IF PDL (1) > 200 AND Y < 191 - F THEN Y = Y + F
90 IF PDL (1) < 50 AND Y > F THEN Y = Y - F
100 IF PDL (6) = 255 THEN HCOLOR= 3: DRAW 1 AT XX,YY
110 IF PDL (5) = 255 THEN HCOLOR= 0: DRAW 1 AT XX,YY
120 IF PEEK ( - 16384) > 127 THEN GET IN$: GOSUB 200
130 XDRAW 1 AT XX,YY
140 GOTO 40
200 REM EINGABE AUSWERTEN
210 IF IN$ = "-" AND R > 0 THEN R = R - 1: IF R = 0 THEN R = 64: RETURN
220 IF IN$ = "+" AND R < 64 THEN R = R + 1: IF R = 64 THEN R = 0: RETURN

230 IF IN$ = CHR$ (8) AND S > 0 THEN S = S - 1: RETURN
240 IF IN$ = CHR$ (21) AND S < 255 THEN S = S + 1: RETURN
250 IF IN$ = "," AND F > 1 THEN F = F - 1: RETURN
260 IF IN$ = "." AND F < 100 THEN F = F + 1: RETURN
270 IF IN$ = "C" THEN 20
275 IF IN$ = "Q" THEN XDRAW 1 AT X,Y: TEXT : GOTO 500
280 IF IN$ = "H" THEN SCALE= 1: ROT= 0: XDRAW 1 AT 0,0: XDRAW 1 AT 140,0
: XDRAW 1 AT 279,0: XDRAW 1 AT 0,191: XDRAW 1 AT 140,191: XDRAW 1 AT
279,191: XDRAW 1 AT 0,96: XDRAW 1 AT 279,96: XDRAW 1 AT 140,96
290 RETURN
500 REM MENUE
510 HOME
520 INVERSE : PRINT "                ";; PRINT "
                PAINTER                ";; PRINT "
                "
530 NORMAL : PRINT
540 PRINT "WOLLEN SIE...": PRINT
550 PRINT "<1> EIN BILD LADEN"
560 PRINT "<2> EIN BILD SAVEN"
570 PRINT "<3> EIN NEUES BILD ANFANGEN"
580 PRINT "<4> DAS BILD INVERTIEREN": PRINT "<5> INSTRUKTIONEN": PRINT "<
6> DAS PROGRAMM BEENDEN"
585 PRINT "<7> WEITERMALEN"
590 VTAB 15: HTAB 1: PRINT "< >": VTAB 15: HTAB 2: GET M$:M = VAL (M$): IF
M < 1 OR M > 7 THEN 590
600 IF M = 1 THEN 1000
610 IF M = 2 THEN 1100
620 IF M = 3 THEN 20
630 IF M = 4 THEN 1200
640 IF M = 6 THEN HOME : PRINT "ENDE": END
645 IF M = 7 THEN POKE - 16304,0: POKE - 16297,0: POKE - 16299,0: GOTO
30
650 HOME : PRINT "WIE MALT MAN MIT PAINTER ?"
660 PRINT : PRINT "DER 'STIFT' WIRD MIT DEM JOYSTICK GE-": PRINT "FUEHRT"

670 PRINT : PRINT "SOLANGE TASTE 1 GEDRUECKT IST, MALT DER": PRINT "'STIF
T'"
680 PRINT "MIT TASTE 2 WIRD GELDESCHT"
690 PRINT : PRINT "AUF DER TASTATUR GIBT ES FOLGENDE FUNK-": PRINT "TIONE
N:": PRINT
```




```

700 PRINT "TASTE --> = STIFT VERGROESSERN"
710 PRINT "TASTE <-- = STIFT VERKLEINERN"
720 PRINT "TASTE + = STIFT DREHEN"
730 PRINT "TASTE - = STIFT ANDERSRUM DREHEN"
740 PRINT "TASTE > = GROESSERE SCHRITTWEITE"
750 PRINT "TASTE < = KLEINERE SCHRITTWEITE"
755 PRINT "TASTE Q = ZURUECK ZUM MENUE"; PRINT "TASTE C = BILDSCHIRM
LOESCHEN"
760 PRINT : PRINT "BITTE EINE TASTE DRUECKEN..."; GET A$: GOTO 500
1000 REM LADEN
1005 HOME
1010 PRINT "NAME DES ZU LADENDENBILDES ODER RETURN"; PRINT "FUER CATALOG"
: PRINT : INPUT ">";NA$
1020 IF NA$ = "" THEN PRINT CHR$(4);"CATALOG": GOTO 1010
1030 PRINT CHR$(4);"BLOAD";NA$; ",A$4000"
1040 GOTO 500
1100 REM SAVEN
1105 HOME
1110 INPUT "NAME DES ZU SAVENDEN BILDES:";SA$
1120 PRINT CHR$(4);"BSAVE";SA$; ",A$4000,L$1FF8"
1130 GOTO 500
1200 REM INVERTIEREN
1210 POKE - 16304,0: POKE - 16297,0: POKE - 16299,0
1220 SCALE= 192: ROT= 0
1230 FOR I = 0 TO 279
1240 XDRAW 1 AT I,0
1250 NEXT : GET WE$
1260 TEXT : GOTO 500

```

Computronic **Abonnement**

Bitte ausschneiden und senden an:
Tronic-Verlag, Postfach 41, 3444 Wehretal 1.

Hiermit bestelle ich die Zeitschrift »Computronic« ab Heft Nr. _____ zum Jahresabonnementspreis (6 Ausgaben) von DM 35,- incl. Versand, Inland und DM 45,- incl. Versand, Ausland.

Name/Vorname: _____

Straße, Nr. _____

Plz, Ort: _____

Ich wünsche folgende Zahlungsweise

Bargeldlos durch Bankeinzug:

Bankleitzahl: _____

Geldinstitut: _____

Konto Nr.: _____

gegen Rechnung

gegen Vorkasse
(Betreffendes bitte ankreuzen)

Datum, Unterschrift _____

Abonnements-Kündigungen:

6 Wochen vor Ablauf des Jahres-Abonnements.

Außerdem kann ich diese Bestellung innerhalb von 10 Tagen beim Verlag widerrufen!

Datum, Unterschrift _____

Haben Sie ein gutes Programm geschrieben?

Wenn ja – wir würden uns freuen, in Ihnen einen Ansprechpartner für eine künftige Zusammenarbeit zu finden. Der TRONIC-Verlag vereinbart mit seinen Software-Autoren pro veröffentlichte volle Seite **ein Honorar von DM 120,-**. Die Auszahlung erfolgt nach Veröffentlichung der Programme in einer unserer Ausgaben. Einzusenden sind:

- Programmbeschreibung
- bespielte Cassette oder Diskette
- Listing (mit Copyright)
- Freiumsschlag

Der Verlag wird vom Autor berechtigt, seine Manuskripte (Programm) zur Darstellung im Heft heranzuziehen und abzudrucken.

Der Autor erklärt sich mit der Lieferung seines Programmes oder seiner Beiträge ausdrücklich bereit, die Verwertung durch den Verlag freizugeben, d. h., er überträgt nicht nur die Nutzung, sondern auch die Übereignung des Computerprogrammes und der Beiträge.

Der Autor verpflichtet sich nur solche Programme anzubieten, die eigene Entwicklungen des Autors sind.

Mit der Veröffentlichung oder dem Anlauf des Programmes und der Beiträge ist dem Verlag gestattet, auch eine anderweitige bzw. weitergehende Verwertung vorzunehmen, da der Autor dem Verlag das Copyright gegen Honorar gestattet hat. Die Verwertung durch den Verlag ist unbeschränkt und unwiderruflich, wenn nicht 10 Tage nach Zusendung der Unterlagen durch den Autor widersprochen wird.

Haben Sie Interesse? Haben Sie noch Fragen? Setzen Sie sich telefonisch mit unserer Redaktion in Verbindung!

TRONIC-VERLAG
DIE REDAKTION



Bowling

Wie in der realistischen Sportart „Bowling“ vorgegeben, beschäftigen wir uns in dieser Spielversion mit dem beliebten Freizeitsport.

Im Spiel stehen Ihnen 10 Durchgänge zur Verfügung. Pro Spieldurchgang hat jeder Spieler 2 Kugeln. Allerdings mit zwei Ausnahmen: hat man mit der ersten Kugel abgeräumt, so ist der nächste Spieler an der Reihe. Wurde beim 10. Durchgang mit der 2. Kugel abgeräumt, bekommt man noch eine 3. Kugel.

Wird ein Spieler zum Wurf aufgefordert, blinkt sein Anfangsbuchstabe auf, der zuvor eingegeben wurde. Nun starten und lenken Sie die Kugel mit folgenden Tasten: >A< die Kugel wird nach unten abgelenkt; >Q< die Kugel wird nach oben abgelenkt.

Ist die Kugel am Rollen, kann sie mit einem zweiten Tastendruck abgelenkt werden. Jeder umgefallene Kegel ergibt einen Punkt. Hat man mit der zweiten Kugel abgeräumt, zählen die mit der nächsten Kugel umgefallenen Kegel doppelt. Wenn mit der ersten Kugel abgeräumt wurde, zählen die mit den nächsten beiden Kugeln umgeworfenen Kegel doppelt.

Geben Sie bitte zu Anfang die Anzahl der Spieler und den ersten Buchstaben des Namens ein. Als letzte Eingabe erfolgt die Geschwindigkeit; sie kann Werte zwischen 1 und 100 annehmen. 1 = sehr schnell; 100 = sehr langsam.

```

1 DIM K(10)
10 GOSUB 1830: GOSUB 1700
15 GR
20 N$(2) = " ": F = - 2: HOME : G1 =
   0: G2 = 0
30 VTAB 22: INPUT "Wieviel Spiele
   r max.2"; Z: IF Z < = 0 OR Z >
   = 3 THEN 60
40 FOR I = 1 TO Z

```

```

50 PRINT "1. Buchstabe des "; I; ".
   Spielers ";: INPUT N$(I): NEXT
   I
60 COLOR= 10: HOME
70 INPUT "Geschwindigkeit Profi 1
   -> Anfaenger 100 "; UU%: IF UU
   % < 1 OR UU% > 100 THEN 70
80 HOME
90 FOR I = 0 TO 39: PLOT I, 20: PLOT
   I, 4: NEXT
110 COLOR= 15
120 VTAB 20: PRINT " 4 4 4
   4 4 4 4 4 4"
130 PRINT N$(1)
140 PRINT "-----"
   "
150 PRINT N$(2)
160 FOR V = 1 TO 10
170 Q = 10: P1 = 0: P2 = 0: F = F + 4
   : P = 0: NN = 2
180 SP = 0
190 Q = 10: SP = SP + 1
200 N = N + 1
210 IF SP = 1 THEN FLASH : VTAB
   21: HTAB 1: PRINT N$(1): NORMAL
   : VTAB 23: HTAB 1: PRINT N$(2
   )
220 IF SP = 2 THEN FLASH : VTAB
   23: HTAB 1: PRINT N$(2): NORMAL
   : VTAB 21: HTAB 1:: PRINT N$(
   1)
230 IF Q < 10 THEN 270
240 FOR I = 1 TO 4: FOR J = 1 TO
   I: PLOT 29 + 2 * I, 10 - 2 * I
   + 4 * J: NEXT J: NEXT I
250 FOR I = 1 TO 10: K(I) = 1: NEXT

260 Q = 0
270 REM KUGEL AM START
280 S = 1: Y = 4
290 POKE - 16368, 0
300 FOR Y = Y + S TO Y + S * 14 STEP
   S
310 PLOT 2, Y
320 FOR I = 1 TO UU%: NEXT
330 X = PEEK ( - 16384): AA = X
340 IF X > 127 THEN POKE - 1636
   8, 0: GOTO 390
350 COLOR= 0: PLOT 2, Y: COLOR= 15

360 NEXT
370 S = - S
380 GOTO 300
390 REM KUGEL ROLLT
400 A = 0
410 IF AA = 193 THEN A = .12
420 IF AA = 209 THEN A = - .12
430 COLOR= 0: PLOT 2, Y: COLOR= 15

440 FF = 0
450 FOR X = 4 TO 32 STEP 2

```

```

460 IF FF = 0 THEN XX = PEEK ( -
16384)
470 IF XX > 127 THEN AA = XX: POKE
- 16368,0:FF = 1
480 IF AA = 193 THEN A = .12
490 IF AA = 209 THEN A = -.12
500 PLOT X,Y + 2 * A
510 FOR I = 1 TO UU%: NEXT
520 Y = Y + 2 * A
530 IF Y > = 20 OR Y < 5 THEN COLOR=
10: FOR I = 0 TO 39: PLOT I,2
0: PLOT I,4: NEXT : COLOR= 15
: GOTO 800
540 COLOR= 0: PLOT X,Y: COLOR= 15

550 NEXT X
560 REM 1.REIHE
570 IF Y > 13 OR Y < 11 THEN 590
580 D1 = SGN (12 - Y): GOSUB 1060

590 PLOT 32,Y + A:Y = Y + 2 * A: COLOR=
0: PLOT 32,Y - A: COLOR= 15: PLOT 33,Y
600 REM 2.REIHE
610 IF Y < 13 OR Y > 15 THEN 630
620 D1 = SGN (14 - Y): GOSUB 1150

630 IF Y < 9 OR Y > 11 THEN 650
640 D1 = SGN (10 - Y): GOSUB 1240

650 COLOR= 0: PLOT 33,Y: COLOR= 1
5: PLOT 34,Y + A:Y = Y + 2 *
A: COLOR= 0: PLOT 34,Y - A: COLOR=
15: PLOT 35,Y
660 REM 3.REIHE
670 IF Y < 15 OR Y > 17 THEN 690
680 D1 = SGN (16 - Y): GOSUB 1330

690 IF Y < 11 OR Y > 13 THEN 710
700 D1 = SGN (12 - Y): GOSUB 1400

710 IF Y < 7 OR Y > 9 THEN 730
720 D1 = SGN (8 - Y): GOSUB 1470
730 COLOR= 0: PLOT 35,Y: COLOR= 1
5:: PLOT 36,Y + A:Y = Y + 2 *
A: COLOR= 0: PLOT 36,Y - A: COLOR=
15: PLOT 37,Y
740 REM 4.REIHE
750 IF Y > = 17 AND Y < = 19 THEN
GOSUB 1530
760 IF Y > = 13 AND Y < = 15 THEN
GOSUB 1570
770 IF Y > = 9 AND Y < = 11 THEN
GOSUB 1610
780 IF Y > = 5 AND Y < = 7 THEN
GOSUB 1660
790 COLOR= 0: PLOT 37,Y: COLOR= 1
5
800 REM UMGEFALLENE ZAEHLEN
810 Q1 = Q:Q = 0
820 FOR I = 1 TO 10:Q = Q + (K(I)
= 0): NEXT I

830 P = 0:P = P + Q - Q1
840 IF SP = 2 AND K2 > 0 THEN HTAB
F: VTAB 23:P2 = P2 + 2 * P: PRINT
P2::K2 = K2 - 1: GOTO 890
850 IF SP = 1 AND K1 > 0 THEN HTAB
F: VTAB 21:P1 = P1 + 2 * P: PRINT
P1::K1 = K1 - 1: GOTO 870
860 IF SP = 1 THEN P1 = P1 + P: VTAB
21: HTAB F: PRINT P1;
870 IF N = 2 AND Q = 10 AND SP =
1 THEN PRINT "8":K1 = K1 + 1

880 IF SP = 2 THEN P2 = P2 + P: VTAB
23: HTAB F: PRINT P2;
890 IF N = 2 AND Q = 10 AND SP =
2 THEN PRINT "6":K2 = K2 + 1

900 IF N = 1 AND P = 10 AND SP =
1 THEN N = 2: PRINT "X":K1 =
K1 + 2: GOSUB 1810
910 IF N = 1 AND P = 10 AND SP =
2 THEN N = 2: PRINT "X":K2 =
K2 + 2: GOSUB 1810
920 IF N = 2 AND Q = 10 AND SP =
1 AND V = 10 THEN NN = 3:K1 =
1
930 IF N = 2 AND Q = 10 AND SP =
2 AND V = 10 THEN NN = 3:K2 =
1
940 IF N < NN THEN 200
950 IF NN = 3 THEN NN = 2
960 N = 0
970 IF SP < = Z - 1 THEN 190
980 G1 = P1 + G1:G2 = P2 + G2
990 NEXT V
1000 GET AA$
1010 PRINT "TOTAL:": PRINT N*(1);
": ";G1: PRINT N*(2);": ";G
2
1015 IF G1 > HI THEN HI = G1:H2 =
1
1016 IF G2 > HI THEN HI = G2:H2 =
1
1017 IF H2 = 1 THEN H2 = 0: GOSUB
2000
1020 PRINT "NOCHMAL ?"
1030 GET AA$: IF AA$ = "J" THEN 2
0
1040 IF AA$ < > "N" THEN 1030
1050 END
1060 REM 1. REIHE 1. KEGEL
1070 IF K(1) = 0 THEN RETURN
1080 K(1) = 0
1090 COLOR= 0: PLOT 31,12: COLOR=
15
1100 D = D1
1110 D1 = INT ( RND (1) * 20 - 10
)
1120 IF D = 1 THEN 1150
1130 IF D = - 1 THEN 1240
1140 GOTO 1400
1150 REM 2. REIHE 2. KEGEL

```




```
1160 IF K(2) = 0 THEN RETURN
1170 K(2) = 0
1180 COLOR= 0: PLOT 33,14: COLOR=
15
1190 D = D1
1200 D1 = INT ( RND (1) * 20 - 10
)
1210 IF D = 1 THEN GOTO 1330
1220 IF D = - 1 THEN GOTO 1400
1230 GOTO 1570
1240 REM 2.REIHE 3.KEGEL
1250 IF K(3) = 0 THEN RETURN
1260 K(3) = 0
1270 COLOR= 0: PLOT 33,10: COLOR=
15: VTAB 21: PRINT ""
1280 D = D1
1290 D1 = INT ( RND (1) * 20 - 10
)
1300 IF D = 1 THEN GOTO 1400
1310 IF D = - 1 THEN GOTO 1470
1320 GOTO 1610
1330 REM 3.REIHE 4.KEGEL
1340 IF K(4) = 0 THEN RETURN
1350 K(4) = 0
1360 COLOR= 0: PLOT 35,16: COLOR=
15: VTAB 21: PRINT ""
1370 IF D1 = 1 THEN GOTO 1530
1380 IF D1 = - 1 THEN GOTO 1570

1390 GOTO 1690
1400 REM 3.REIHE 5.KEGEL
1410 IF K(5) = 0 THEN RETURN
1420 K(5) = 0
1430 COLOR= 0: PLOT 35,12: COLOR=
15: VTAB 21: PRINT ""
1440 IF D1 = 1 THEN GOTO 1570
1450 IF D1 = - 1 THEN GOTO 1610

1460 GOTO 1690
1470 REM 3.REIHE 6.KEGEL
1480 IF K(6) = 0 THEN RETURN
1490 K(6) = 0
1500 COLOR= 0: PLOT 35,8: COLOR=
15: VTAB 21: PRINT ""
1510 IF D1 = 1 THEN GOTO 1610
1520 GOTO 1690
1530 REM 4.REIHE 7.KEGEL
1540 K(7) = 0
1550 COLOR= 0: PLOT 37,18: COLOR=
15: VTAB 21: PRINT ""
1560 GOTO 1690
1570 REM 5.REIHE 8.KEGEL
1580 K(8) = 0
1590 COLOR= 0: PLOT 37,14: COLOR=
15: VTAB 21: PRINT ""
1600 GOTO 1690
1610 REM 6.REIHE 9.KEGEL
1620 K(9) = 0
1630 COLOR= 0: PLOT 37,10: COLOR=
15: VTAB 21: PRINT ""
1640 GOTO 1690
1650 IF N = 1 AND P = 10 AND SP =
```

```
2 THEN N = 2: PRINT ;"X":K2 =
K2 + 2:H2 = P2
1660 REM LETZTER KEGEL
1670 K(10) = 0

1680 COLOR= 0: PLOT 37,6: COLOR=
15: VTAB 21: PRINT ""
1690 RETURN
1700 HOME : VTAB 23: SPEED= 100
1710 FLASH : PRINT "
B O W L I N G "
NORMAL : PRINT
1720 GOSUB 1810
1725 PRINT : PRINT "*****
*****":
PRINT : PRINT "
PRODUCED": PRINT : PRINT "
BY": PRINT :
PRINT " HANS-PE
TER": PRINT : PRINT "
VOGT": PRINT : PRINT
: PRINT "*****
*****"
1730 PRINT "DIE REGELN": PRINT :
PRINT "JEDER SPIELER BEKOMMT
PRO DURCHGANG": PRINT : PRINT
"ZUNAECHST 2 KUGELN. IM GESAM
TEN SIND 10": PRINT : PRINT "
DURCHGAENGE ZU SPIELEN.": PRINT
: PRINT "DER SPIELER, DER AN
DER REIHE IST, WIRD": PRINT :
PRINT "DADURCH ANGEZEIGT, DA
SS SEIN BUCHSTABE": PRINT : PRINT
"BLINKT"
1740 PRINT
1750 PRINT "STEUERUNG DER KUGEL:"
: PRINT : PRINT " A = KUGEL W
IRD NACH UNTEN": PRINT "
ABGELENK
T": PRINT " Q = KUGEL WIRD NA
CH OBEN": PRINT : PRINT "JEDE
ANDERE TASTE = KUGEL ROLLT G
ERADE"
1760 PRINT
1770 PRINT "WENN DIE KUGEL ROLLT,
SO KANN MAN SIE": PRINT : PRINT
"EINEM WEITEREN TASTENDRUCK A
BLENKEN": PRINT : PRINT : PRINT
"WERDEN ALLE 10 KEGEL MIT DER
ERSTEN": PRINT : PRINT "KUGE
L ABGERAEUMT, SO IST DER NAECH
STE": PRINT : PRINT "SPIELER
AN DER REIHE. BEI DEN NAECHS
TEN"
1780 PRINT : PRINT "BEIDEN KUGELN
DES VORHERIGEN SPIELERS": PRINT
: PRINT "WERDEN DIE ABGERAEUM
TEN KEGEL DOPPELT": PRINT : PRINT
"GEZAEHLT": PRINT : PRINT "RA
EUMT DER SPIELER ERST MIT DER
ZWEITEN": PRINT : PRINT "KUG
EL ALLE 10 KEGEL AB, SO ZAEHL
```



```
EN NUR": PRINT : PRINT "DIE B
EI SEINER NAECHSTEN KUGEL ABG
E-"
1790 PRINT : PRINT "RAEUMTEN KEGE
L DOPPELT. WIRD BEIM 10.": PRINT
: PRINT "DURCHGANG MIT DER 2.
KUGEL ABGERAEUMT,": PRINT : PRINT
"SO BEKOMMT DER SPIELER NOCH
EINE KUGEL."
1800 SPEED= 160: FOR OP = 1 TO 13
: PRINT "
": PRINT :
NEXT : SPEED= 255: RETURN
1810 FOR Y = 1 TO 2: POKE 768,2: POKE
769,241: CALL 770: POKE 768,2
: POKE 769,231: CALL 770: POKE
768,2: POKE 769,231: CALL 770
: POKE 768,2: POKE 769,228: CALL
770: POKE 768,2: POKE 769,231
: CALL 770: POKE 768,2: POKE
769,222: CALL 770: POKE 768,2
: POKE 769,214: CALL 770
1820 POKE 768,2: POKE 769,206: CALL
770: POKE 768,6: POKE 769,192
: CALL 770: NEXT : RETURN
1830 FOR Y = 770 TO 807: READ AS:
POKE Y,AS: NEXT : RETURN
1840 DATA 172,1,3,174,1,3,169,4,3
2,168,252,173,48,192,232,208,
253,136,208,239,206,0,3,208,2
31,96,32,32,112,8,24,216,136,
8,160,160,16,56
2000 HI$ = STR$ (HI):LE = LEN (H
I$)
2005 GR : COLOR= 15:AB = 37
2006 VLIN 29,36 AT 3: VLIN 29,36 AT
7: HLIN 4,6 AT 32: HLIN 10,14
AT 29: HLIN 10,14 AT 36: VLIN
30,35 AT 12
2010 IF LE = 3 THEN L$(1) = LEFT$
(HI$,1):L$(2) = MID$ (HI$,2,
1):L$(3) = RIGHT$ (HI$,1)
2020 IF LE = 2 THEN L$(2) = LEFT$
(HI$,1):L$(3) = RIGHT$ (HI$,
1)
2030 IF LE = 1 THEN L$(3) = HI$
2040 FOR I = 3 TO 0 STEP - 1
2045 AB = AB - 7
2050 FOR ZZ = 0 TO 9:ZZ$ = STR$
(ZZ): IF L$(I) = ZZ$ THEN ON
(ZZ + 1) GOTO 3000,3100,3200,
3300,3400,3500,3600,3700,3800
,3900
2060 NEXT ZZ
2070 NEXT I
2080 RETURN
3000 REM 0
3010 HLIN AB + 1,AB + 3 AT 29: HLIN
AB + 1,AB + 3 AT 36: VLIN 30,
35 AT AB: VLIN 30,35 AT AB +
4
```

```
3020 NEXT I
3100 REM 1
3110 VLIN 29,36 AT AB + 4: PLOT A
B + 3,29: PLOT AB + 2,30: PLOT
AB + 1,31: PLOT AB,32
3120 NEXT I
3200 REM 2
3210 PLOT AB,30: HLIN AB + 1,AB +
3 AT 29: VLIN 30,31 AT AB + 4
: PLOT AB + 3,32: PLOT AB + 2
,33: PLOT AB + 1,34: PLOT AB,
35: HLIN AB,AB + 4 AT 36
3220 NEXT I
3300 REM 3
3310 PLOT AB,30: HLIN AB + 1,AB +
3 AT 29: VLIN 30,31 AT AB + 4
: HLIN AB + 2,AB + 3 AT 32: VLIN
33,35 AT AB + 4: HLIN AB + 1,
AB + 3 AT 36: PLOT AB,35
3320 NEXT I
3400 REM 4
3410 VLIN 29,32 AT AB: HLIN AB +
1,AB + 4 AT 32: VLIN 33,36 AT
AB + 2
3420 NEXT I
3500 REM 5
3510 HLIN AB,AB + 4 AT 29: VLIN 3
0,32 AT AB: HLIN AB + 1,AB +
3 AT 32: VLIN 33,35 AT AB + 4
: HLIN AB + 1,AB + 3 AT 36: PLOT
AB,35
3520 NEXT I
3600 REM 6
3610 HLIN AB + 1,AB + 3 AT 29: PLOT
AB + 4,30: VLIN 30,35 AT AB: HLIN
AB + 1,AB + 3 AT 36: VLIN 33,
35 AT AB + 4: HLIN AB + 1,AB +
3 AT 32
3620 NEXT I
3700 REM 7
3710 HLIN AB,AB + 4 AT 29: VLIN 3
0,31 AT AB + 4: PLOT AB + 3,3
2: PLOT AB + 2,33: PLOT AB +
1,34: VLIN 35,36 AT AB
3720 NEXT I
3800 REM 8
3810 HLIN AB + 1,AB + 3 AT 29: VLIN
30,31 AT AB: VLIN 30,31 AT AB
+ 4: HLIN AB + 1,AB + 3 AT 3
2: VLIN 33,35 AT AB: VLIN 33,
35 AT AB + 4: HLIN AB + 1,AB +
3 AT 36
3820 NEXT I
3900 REM 9
3910 HLIN AB + 1,AB + 3 AT 29: VLIN
30,31 AT AB: VLIN 30,35 AT AB
+ 4: HLIN AB + 1,AB + 3 AT 3
2: HLIN AB + 1,AB + 3 AT 36: PLOT
AB,35
3920 NEXT I
```



Programm für 10 Funktionstasten

Jeder Apple II-Besitzer wird sich schon einmal darüber geärgert haben, daß sein Rechner im Gegensatz zu fast allen anderen Rechnern keine Funktionstasten hat (von CNTL Y mal abgesehen).

Aber man kann Abhilfe schaffen. Mit diesem Programm lassen sich die Tasten „0-9“ zu Funktionstasten umgestalten. Durch die Eingabe von „ESC“, gefolgt von „0-9“ im Monitor, Apple-soft-Basic oder einem Programm, wird eine Funktion abgerufen. Die Tasten „1-9“ sind mit häufig benutzten Basic-Befehlen belegt, die Taste 0 kann man nach eigenem Wunsch mit einem Text belegen. Hierfür ist das „DEF“-Programm zuständig. Die „&“-Funktion kann benutzt werden, um dieses Programm auszuschalten, da es gegen „RESET“ geschützt ist.

Doch nun zur Beschreibung und Funktion des Textfiles: Zuerst wird das ROM in die Language-Karte dubliziert und dort verändert.

Der Delimiter (der Bindestrich hinter Adresse) des Monitors wird zu einem Doppelpunkt umdefiniert. Nun kann man Zeilen im Monitor direkt editieren, ohne den Doppelpunkt mit der Hand jedesmal einfügen zu müssen.

Das Listformat wird auf volle Bildschirmgröße ausgedehnt. Das heißt, man kann nun Basic-Zeilen mit PRINT-Texten editieren, die über eine Zeile hinausgehen, ohne auf den ESC-Cursor zurückgreifen zu müssen.

Die Routine ESCNEW des ROMs wird um eine Verzweigung erweitert, so daß nun bei Eingaben im Monitor oder Basic getestet wird, ob eine der Tasten „ESC 1 – ESC 9“ oder „ESC 0“ gedrückt wurde. Ist dieses der Fall, so

verzweigt der Rechner von „GETLN“ zu „RDCHAR“ zu „ESG“ zu „ESCNEW“ zu der Funktionsroutine bei 9500. Hier werden nun Zeichen aus der ROM-Datei in den Eingabepuffer kopiert, am Bildschirm ausgegeben und getestet, so daß der Text nicht über den Puffer hinausgehen kann.

Die „&“-Funktion wird zum Hin- und Herschalten zwischen ROM und Language-Karte definiert, da es mitunter beim Listen störend ist, wenn Basic-Befehle wahllos getrennt werden. Das Programm wird hierbei nicht gelöscht. Um dieses zu erreichen, tippt man im Monitor das Maschinenprogramm ein (HEX-DUMP) und vergleicht es danach mit dem Assemblerlisting. Eventuelle Fehler müssen vor dem Start korrigiert werden, da sich der Rechner sonst „aufhängt“. Danach muß das Basic-Programm eingegeben werden. Dieses Programm zerstört sich selber, nachdem es ein Textfile namens „FBASIC“ erstellt hat.

Um die Funktionstasten immer zur Verfügung zu haben, empfiehlt es sich, im „HELLO“-Programm ein „EXEC FBASIC“ einzufügen.

Achtung:

Dieses Programm funktioniert nur bei Maxfiles 3 (Normalfall). Falls man aber Maxfiles erhöht, zerstört der DOS-Filepuffer das Maschinenprogramm und man landet bei Betätigung einer der Funktionstasten im Monitor.

```

10 REM *****
15 REM *
20 REM * PROGRAMM FUER *
25 REM *
30 REM * 10 FUNKTIONSTASTEN *
35 REM *
40 REM * -- COPYRIGHT BY -- *
45 REM *
50 REM * & HANS - GEORG & *
55 REM *
60 REM * &&& WITTKEMPER &&& *
65 REM *
70 REM * &&& 4358 HALTERN &&& *
75 REM *
80 REM * & ZUM DACHSBERG 16 & *
85 REM *
90 REM *****
95 REM
99 REM INITIALISIERUNG
100 HIMEM: 38144: CLEAR : SPEED= 255:D$ = CHR$(4): HOME
105 CALL 1002:DO$ = D$ + "OPEN":DC$ = D$ + "CLOSE":DW$ = D$ + "WRITE":N$ = "FBASIC"
110 VTAB 5: HTAB 5: PRINT "NUN WIRD DAS TEXTFILE ERSTELLT "
200 PRINT : PRINT DO$;N$
210 PRINT DW$;N$
215 REM BILDSCHIRM AUSSCHALTEN
220 PRINT "POKE37,27"
230 PRINT "NOMONC,I,0"
235 REM LANGUAGE-CARD LADEN
240 PRINT "CALL-151"
250 PRINT "C081 C081"
260 PRINT "D000<D000.FFFFFM"

```




```

270 PRINT "C083 C083"
275 REM MONITOR '-' ZU ':'
280 PRINT "FD9F;BA"
285 REM '&' ZUM UMSCHALTEN
290 PRINT "3F5:4C 8B 95"
295 REM NEUES LIST FORMAT
300 PRINT "D705:4C 12 D7"
305 REM 10 FUNKTIONEN INS ROM
310 PRINT "FBA5:C9 BA B0 07 C9 B0 90 3 4C 0 95"
320 PRINT "FBB0:C9 CE B0 E3 C9 C9 90 DF"
330 PRINT "FBB8:C9 CC F0 DB D0 DD EA EA"
335 REM START FBASIC
340 PRINT "E0036"
345 REM LADE DIE FUNKTIONEN
350 PRINT "BLOADESC1"
355 PRINT "HIMEM:38144"
360 PRINT "HOME"
370 PRINT "?"; CHR$ (34); "ES STEHEN NUN 10 FUNKTIONSTASTEN BEREIT"; CHR$ (34)
380 PRINT DC$;N$
390 PRINT D$;"EXEC";N$
400 PRINT D$;"FP"
410 END

```

Teil 2

```

10 HOME : PRINT CHR$ (12)
15 INPUT " DEFINITION FUER TASTE 'ESC 0'
    ==> ";A$
20 FOR I = 1 TO LEN (A$): POKE 38308 + I, ASC ( MID$ (A$,I))
30 IF I < > LEN (A$) THEN NEXT
40 POKE 38308 + I, ASC ( MID$ (A$,I)) + 128
50 HOME : PRINT "DIE NEU TASTEN DEFINITION FUER ESC 0": PRINT : PRINT "=="> IST
NUN ";A$: END

```

Teil 3

*9500.95A3

```

9500: 48 A9 D0 BD 56 95 A9 D0
9508: BD 57 95 68 C9 B0 D0 OF
9510: A9 A5 BD 56 95 A9 95 BD
9518: 57 95 A0 00 4C 55 95 A0
9520: C3 C9 B1 F0 30 A0 D4 C9
9528: B2 F0 2A A0 D9 C9 B3 F0
9530: 24 A0 03 C9 B4 F0 1E A0
9538: 06 C9 B5 F0 18 A0 0E C9
9540: B6 F0 12 A0 55 C9 B7 F0
9548: 0C A0 94 C9 B8 F0 06 A0
9550: 29 EA EA EA EA B9 D0 D0
9558: C9 80 B0 0D 9D 00 02 EB
9560: C8 20 75 95 4C 55 95 EA
9568: EA 38 E9 80 9D 00 02 EB
9570: 20 75 95 60 00 48 A9 FF
9578: B5 32 68 18 69 B0 20 ED
9580: FD E0 FE 90 05 68 68 4C
9588: 62 FD 60 AD 74 95 C9 B0
9590: F0 09 A9 80 8D 74 95 BD
9598: 81 C0 60 A9 00 8D 74 95
95A0: 8D B3 C0 60

```

BSAVE ESC 1, A 38144, L 181

Manfred S. Schmidt
**DER MIKRO-
 COMPUTER
 IM BERUF**

HEYNE COMPUTER-BÜCHER
 Manfred S. Schmidt
 DER MIKROCOMPUTER
 IM BERUF
 Heyne-Buch Nr. 15/4, DM 9,80
 Originalausgabe

Jump



**Ein schnelles Spiel
für den ZX-Spectrum
16/48 K.
Geschrieben in
Maschinensprache.**

Fred arbeitet mit seinen Kollegen auf einer Großbaustelle. Bei der Fertigstellung eines neuen Hochhauses werden Gerüste angebracht und Leitern aufgestellt.

Fred hat die Aufgabe, die obere Fassade des Hauses fertigzustellen. Auf seinem Weg nach oben, über Leitern und Gerüste, gibt es Schwierigkeiten. Bauschutt, Fässer und Arbeitsmaterial, alles, was seine Kollegen für die Arbeit brauchen, kann für Fred gefährlich werden, wenn es herunterfällt. Um diesen Gefahren rechtzeitig auszuweichen, muß Fred immer häufiger Pausen einlegen und seine mitgebrachten Hamburger essen, um bei Kräften zu bleiben.

Das Spiel enthält neun Bilder. Um ins nächste Bild zu kommen, muß geschickt vorgegangen werden. Mit den Bildern verändern sich auch die Schwierigkeiten, mit denen Fred zu kämpfen hat. Zusätzlich muß darauf geachtet werden, das nicht zuviel Zeit für ein Bild benötigt wird. Denn die Uhr, zu sehen in den letzten beiden Bildschirmzeilen, ist unbestechlich. Auch darf Fred, um das Spiel nicht vorzeitig zu beenden, nicht noch vom Gerüst fallen. Wenn Sie es geschafft haben, alle neun Bilder zu bewältigen, beginnt das Spiel von vorn. Allerdings – ein wenig schneller.

Steuerung:

Q bis T: Springen / Leiter hoch

Caps Shift bis V: Leiter runter

O: links

P: rechts

Sie können das Spiel auch nach Ihren Wünschen gestalten, dazu bestehen folgende Möglichkeiten: mit >S< läßt

sich der Schwierigkeitsgrad einstellen; mit >A< läßt sich die Spielseite einstellen, an der das Spiel beginnen soll; mit >L< läßt sich die Anzahl der Spielfiguren im Bereich von 1 bis 5 wählen.

Zum Laden: geben Sie zuerst das Listing Teil 1 ein und speichern Sie es auf Kassette. Danach geben Sie den Maschinencode nach Listing Teil 3 ein. Hierfür sollten Sie entweder den bevorzugten Hex-Monitor laden oder das Programm nach Listing 4 verwenden.

Die Version des HEX-LOADERS beinhaltet ebenfalls eine Routine zur Prüfsummen-Bildung, wie sie in Listing 3 verwendet wird. Nachdem Sie den Maschinensprache-Teil noch einmal überprüft haben, können Sie auch diesen Teil auf Kassette speichern. Bei Verwendung des HEX-LOADERS geht das ganz einfach durch GOTO 1000.

Jetzt geben Sie einfach NEW und das Programm nach Listing 2 ein, welches Sie anschließend mit GOTO 5000 auf Kassette speichern.

Jetzt nehmen Sie eine andere Kassette und speichern darauf das Programm aus Listing 2 mit GOTO 5000 ab.

Dann laden Sie das Programm aus Listing 1. Das Programm wird sofort zur Ausführung gelangen, und nun legen Sie die Kassette ein, auf der nur das Programm nach Listing 2 enthalten ist. Schalten Sie den Rekorder auf Aufnahme, und das Programm aus Listing 1 wird nun die Titelgrafik auf die Kassette speichern. Zum Schluß laden Sie den Maschinencode Teil mit LOAD ""CODE 29260 und speichern ihn hinter die Titelgrafik mit SAVE „JUMP-Code“ CODE 29260,2900.

Nun haben Sie eine vollständige Kassette mit dem Programm JUMP. Spulen Sie zurück und laden Sie das Programm mit LOAD"".

Listing 1 : Titelfgrafik

```

1 REM © by Michael Havermeester
2 INK 2: PAPER 0: BORDER 1: 0
L6 : PRINT FLASH 1;TAB 31;TAB 0;
"STOP THE TAPE";TAB 0
: DATA 02,00,01,00,00,01,00
3 DATA 07,11,10,00,00,00,00
4 DATA 10,04,10,00,004,100,10
5 DATA 100,100,004,000,000,10
6 DATA 100,00,02,00,00,00,07
7 DATA 00,00,01,01,07,11,10
8 DATA 00,00,00,00,44,00,10
9 DATA 00,100,100,100,100,100
10 DATA 000,000,100,100,100,00
11 DATA 04,40,00,00,00,00,04
12 DATA 00,100,100,100,100,00,
13 DATA 04,04,04,00,00,100,00
14 DATA 00,04,00,100,100,100,1
15 DATA 00,00,04,04,04,00,00
16 DATA 00,101,140,107,100,00,
17 DATA 00,00,101,140,107,100,
18 DATA 00,000,00,001,00,00,00
19 DATA 110,00,00,00,00,100,00
20 DATA 00,00,00,000,100,171,1
21 DATA 000,000,000,000,00,00,
22 DATA 00,100,00,00,00,00,00
23 DATA 100,107,140,101,00,00,
24 DATA 140,04,00,10,014,00,10
25 DATA 100,00,00,04,04,00,00
26 DATA 100,00,00,00,00,00,00
27 FOR i=000000 TO 00474: READ
28 POKE i,a: NEXT i
29 POKE 20675,117: POKE 20675,
114
30 BRIGHT 1: PAPER 0: INK 4: B
ORDER 1: CLS
40 PRINT "SCORE:01000 HI:00000
SHT:"; FLASH 1; INK 5;CHR# 164;
CHR# 164;CHR# 164; FLASH 0;"
50 PRINT
60 PRINT INK 5;TAB 30;CHR# 160
70 PRINT
80 PRINT TAB 12;CHR# 162;TAB 2
0;CHR# 150
90 PRINT " "; INK 3;CHR# 150;
INK 4;CHR# 150;
100 FOR i=1 TO 20: PRINT INK 3;
CHR# 150;: NEXT i: PRINT
110 PRINT TAB 3;CHR# 150
120 PRINT TAB 3;CHR# 150
130 PRINT TAB 10;CHR# 162
140 PRINT " "; INK 3;
150 FOR i=1 TO 20: PRINT INK 3;
CHR# 150;: NEXT i: PRINT INK 4;C
HR# 150; INK 3;CHR# 150
160 PRINT TAB 20;CHR# 150
170 PRINT TAB 20;CHR# 150
180 PRINT TAB 5;CHR# 162;TAB 0;
CHR# 150;TAB 20;CHR# 150
190 PRINT " "; INK 3;CHR# 150;

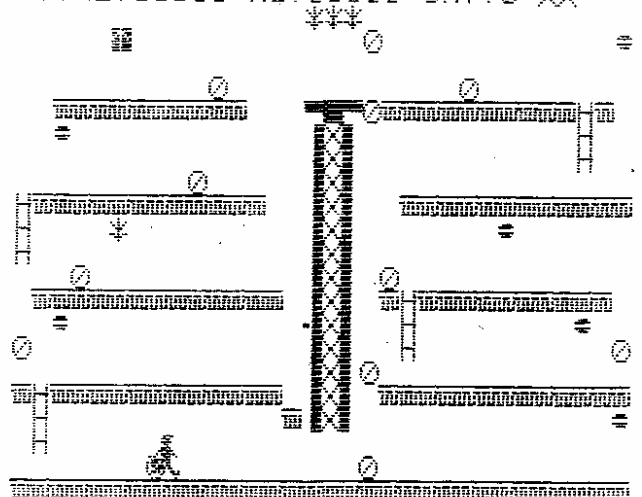
```

```

INK 4;CHR# 150;
200 FOR i=1 TO 20: PRINT INK 3;
CHR# 150;: NEXT i: PRINT
310 PRINT TAB 3;CHR# 150
220 PRINT TAB 3;CHR# 150
230 PRINT TAB 10;CHR# 162;TAB 1
5;CHR# 150;TAB 17;CHR# 162;TAB 2
0;CHR# 150
240 PRINT " ";
250 FOR i=1 TO 20: PRINT INK 3;
CHR# 150;: NEXT i: PRINT INK 4;C
HR# 150; INK 3;CHR# 150; INK 4;C
HR# 150
260 PRINT TAB 15; INK 5;CHR# 16
1; INK 4;TAB 20;CHR# 150
270 PRINT INK 7;TAB 10;CHR# 144
;CHR# 140; INK 4;TAB 20;CHR# 150
280 PRINT TAB 3;CHR# 160; INK 7
;TAB 10;CHR# 145;CHR# 147; INK 4
;TAB 20;CHR# 150
290 PRINT INK 3;CHR# 150; INK 4
;CHR# 150;: FOR i=1 TO 20: PRINT
INK 3;CHR# 150;: NEXT i: PRINT
300 PRINT 00;AT 0,0;" TIME:";
INK 2;" PAPER 7;"
310 PAPER 0: INK 0
320 PRINT AT 7,0; FLASH 1; INK
0; PAPER 2;"START THE TAPE": PRI
NT AT 0,0
400 SAVE "JUMP"SCREEN#
430 STOP
440 SAVE "JUMP-SCF" LINE 0

```

SCORE:00000 HI:00000 SHT:5 XX



Listing 2 :

BASIC-Steuerprogramm

```

1 REM © by Michael Havemeste
  and HUGS Deteli
2 CLEAR 23000: PAPER 0: BORDER
R 0: INK 2: CLS: BRIGHT 1: FLAG
H 1: PRINT AT 10,0:
                                JUMP W
IRD GELADEN !!
                                PAPER 0:
INK 0: FLASH 0: BRIGHT 0: LOAD "
JUMP"SCREEN$
3 PRINT AT 0,0
4 LOAD "JUMP-Code"CODE 20260,
2000
5 DEF FN a$(X)="0000"+STR$ X
) (LEN STR$ X TO )
6 POKE 23675,117: POKE 23676,
114
30 LET seite=1: LET hi=0: LET
le=3: LET schw=0: LET s=1
40 LET b#=CHR$ 13
50 IF INKEY#="" THEN GO TO 50
51 LET a#=b#+TAB$ 2: PRINT "
Steuerung: - "+b#+b#+"-T...
.....Leiter hoch/Sprung "+b#+C
APS SHIFT-U,Leiter runter
+b#+"-O.....Links
"+b#+"-P.....rechts
55 BORDER RND*7: INK 7: PAPER
1: CLS
60 LET q=.05
70 FOR f=1 TO LEN a$
80 PRINT a$(f): IF a$(f)=CHR$
13 THEN PAPER 1+RND*6: INK 0
90 IF INKEY#="" THEN GO TO 20
100 LET q=-q*1.04: BEEP .05,q
110 NEXT f
120 PAUSE 60
130 FOR f=1 TO 15
140 FOR q=0 TO 7
150 BEEP f/500,q
160 PAUSE 1
170 BORDER q:
180 IF INKEY#="" THEN GO TO 20
190 NEXT q
200 NEXT f
210 PAPER 1+RND*6: INK 0
220 PRINT AT 9,0:"< Fuer die
Wahl des Schwierig- keitsgrad
es 0-9"
230 PAPER 1+RND*6: INK 0
240 PRINT AT 12,0:"< Fuer die
Wahl der Spielseite 1-9"
250 PAPER 1+RND*6: INK 0
260 PRINT AT 15,0:"< Fuer die
Wahl der Leben 1-5"
270 PAPER 1+RND*6: INK 0
280 PRINT AT 18,0:"Eine andere
Taste< Wenn keine Rendere
ng gewünscht wird."
290 PRINT @6: FLASH 1: INVERSE
1: AT 0,0:"Druecke die Taste dei
ner Wahl!"; FLASH 0:

```

```

140 FOR f=0 TO 13 STEP 3: BEEP
.005,f: NEXT f
150 IF INKEY#="" THEN GO TO 140
200 LET a#=INKEY#: IF a#<>"6" A
ND a#<>"3" THEN GO TO 220
210 INPUT "Schwierigkeitsgrad (
1-9) : ";schw: IF schw>9 THEN LE
T schw=0
220 IF schw<1 THEN LET schw=1
230 LET schw=10-schw
240 GO TO 50
250 IF a#<>"R" AND a#<>"a" THEN
GO TO 240
260 INPUT "Spielseite (1-9) : "
seite: IF seite>9 THEN LET seite
=0
270 IF seite<1 THEN LET seite=1
280 GO TO 50
290 IF a#<>"L" AND a#<>"l" THEN
GO TO 200
300 INPUT "Anzahl der Leben (1-
5) : ";le: IF le>5 THEN LET le=5
310 IF le<1 THEN LET le=1
320 GO TO 50
330 LET s=seite-1: POKE 30655,s
340 POKE 31079,0
350 LET sc=0: LET f=0
360 LET li=le
370 INK 7: PAPER 0: BORDER 0: C
LS
380 GO SUB 4000
390 POKE 23659,0: PRINT AT 22,2
:"TIME:" INK 2: PAPER 7:
POKE 23659,2
345 BORDER 1
355 POKE 23636,s
360 PRINT ;; RANDOMIZE USR 2650
370 FOR f=1 TO 3: BEEP .15,1: B
EEP .15,5: BEEP .4,1: BEEP .5,-4
380 NEXT f
390 PRINT ;; GO TO USR 29503
400 BEEP .9,1: BEEP .9,-0: BEEP
.9,-1
410 LET li=li-1
420 LET a#="G A M E O V E R"
430 PRINT AT 10,7:
440 IF li<0 THEN GO TO 310
450 FOR f=1 TO LEN a$
460 PRINT a$(f):
470 BEEP .15,15-f*2
480 FOR q=1 TO 40: NEXT q:
490 NEXT f: BEEP .6,-22
500 FOR q=0 TO 56 STEP 2
510 POKE 23536,q: PRINT ;; RAND
OMIZE USR 26506
520 NEXT q
530 FOR f=1 TO 600: NEXT f
540 IF sc>hi THEN LET hi=sc
550 GO TO 40
1000 FOR f=13 TO 0 STEP -3: BEEP
.005,f: NEXT f
1010 LET sc=sc+20: GO SUB 4000
1020 GO TO 390
2000 LET f=42: LET q=15
2005 POKE 30794,120
2010 POKE 30631,201
2020 POKE 23536,f: PRINT ;; RAND
OMIZE USR 29506
2030 LET f=(f=42)*21+(f=21)*42
2040 PRINT ;; LET l=USR 30774: P
OKE 23659,2
2050 POKE 30631,237
2060 IF l=500 THEN GO TO 2500

```

```

2000 LET s=c#s+c+50
20040 PRINT AT 0,0;FN a#(s)
20050 BEEP .004;.9: LET a=a-1.4
20060 GO TO 20020
20070 LET s=s+1
20080 IF s>8 THEN POKE 30855,PEEK
30855-8: POKE 31079,PEEK 31079+
1: LET s=0
2010 GO TO 310
4000 INK 4: PRINT AT 0,0;"SCORE:
";FN a#(s); "HI:";FN a#(H1);
4010 PRINT "GHT:";s+1;" ";
4020 FOR f=1 TO 11: PRINT INK 5;
FLPBH 1;CHR# 164; NEXT f
4030 INK 7: RETURN
5000 SAVE "JUMP" LINE 0
  
```

HEX-LOADER

```

0 REM 619946 Maschenen-Code
0 CLEAR 20000
10 LET a=10: LET b=11: LET c=1
2: LET d=13: LET e=14: LET f=15
15 INPUT "Start Adresse : ";a
7
17 LET pruef=0
20 PRINT a;TAB 8;
30 FOR h=1 TO 7
30 INPUT LINE i#: POKE a;VAL
i#(1)*16+VAL i#(2)
30 PRINT i#;" ";
30 LET pruef=pruef+PEEK a; L
ET a=a+1
40 NEXT h
45 PRINT "=";pruef
50 GO TO 17
55 REM Code ueberpruefen
60 INPUT "Start Adresse ";a
7
65 PRINT a;" ";
70 LET pruef=0
75 FOR h=1 TO 7
80 LET i=PEEK a;
80 LET j=INT (i/16)
80 PRINT CHR# (j+48+(j>9)*7);C
HR# (i-j*16+48+(i-j*16>9)*7);" ";
90 LET pruef=pruef+PEEK a; L
ET a=a+1
100 NEXT h
110 PRINT "=";pruef
120 GO TO 65
4000 SAVE "HEX-LOADER": SAVE "co
de"CODE 20000,20000
  
```

Listing 3 : Hexdump

00000000	4E	6A	04	00	00	00	00	00	00
00000001	21	40	02	77	00	00	00	00	00
00000002	74	75	00	73	01	01	00	00	00
00000003	01	40	00	00	00	70	70	00	00
00000004	00	00	01	00	00	10	00	00	00
00000005	00	00	00	00	00	00	00	00	00
00000006	07	00	10	00	00	00	00	00	00
00000007	00	10	00	00	00	00	00	00	00
00000008	00	00	00	00	00	00	00	00	00
00000009	00	00	04	10	00	00	07	00	00
0000000A	00	00	00	00	00	00	10	00	00
0000000B	00	00	00	00	00	00	10	00	00
0000000C	00	00	00	00	00	00	00	00	00
0000000D	00	00	00	00	00	00	00	00	00
0000000E	00	00	00	00	00	00	00	00	00
0000000F	00	00	00	00	00	00	00	00	00
00000010	00	00	00	00	00	00	00	00	00
00000011	00	00	00	00	00	00	00	00	00
00000012	00	00	00	00	00	00	00	00	00
00000013	00	00	00	00	00	00	00	00	00
00000014	00	00	00	00	00	00	00	00	00
00000015	00	00	00	00	00	00	00	00	00
00000016	00	00	00	00	00	00	00	00	00
00000017	00	00	00	00	00	00	00	00	00
00000018	00	00	00	00	00	00	00	00	00
00000019	00	00	00	00	00	00	00	00	00
0000001A	00	00	00	00	00	00	00	00	00
0000001B	00	00	00	00	00	00	00	00	00
0000001C	00	00	00	00	00	00	00	00	00
0000001D	00	00	00	00	00	00	00	00	00
0000001E	00	00	00	00	00	00	00	00	00
0000001F	00	00	00	00	00	00	00	00	00

00000000	00	40	00	70	00	00	00	00	00
00000001	00	00	00	00	00	00	00	00	00
00000002	00	00	00	00	00	00	00	00	00
00000003	00	00	00	00	00	00	00	00	00
00000004	00	00	00	00	00	00	00	00	00
00000005	00	00	00	00	00	00	00	00	00
00000006	00	00	00	00	00	00	00	00	00
00000007	00	00	00	00	00	00	00	00	00
00000008	00	00	00	00	00	00	00	00	00
00000009	00	00	00	00	00	00	00	00	00
0000000A	00	00	00	00	00	00	00	00	00
0000000B	00	00	00	00	00	00	00	00	00
0000000C	00	00	00	00	00	00	00	00	00
0000000D	00	00	00	00	00	00	00	00	00
0000000E	00	00	00	00	00	00	00	00	00
0000000F	00	00	00	00	00	00	00	00	00
00000010	00	00	00	00	00	00	00	00	00
00000011	00	00	00	00	00	00	00	00	00
00000012	00	00	00	00	00	00	00	00	00
00000013	00	00	00	00	00	00	00	00	00
00000014	00	00	00	00	00	00	00	00	00
00000015	00	00	00	00	00	00	00	00	00
00000016	00	00	00	00	00	00	00	00	00
00000017	00	00	00	00	00	00	00	00	00
00000018	00	00	00	00	00	00	00	00	00
00000019	00	00	00	00	00	00	00	00	00
0000001A	00	00	00	00	00	00	00	00	00
0000001B	00	00	00	00	00	00	00	00	00
0000001C	00	00	00	00	00	00	00	00	00
0000001D	00	00	00	00	00	00	00	00	00
0000001E	00	00	00	00	00	00	00	00	00
0000001F	00	00	00	00	00	00	00	00	00

Unser **Software-Service**

Das ist
phantastisch!!!

Hier können
Sie telefonisch
bestellen:

Posteingang – Postausgang

Wir haben uns bemüht, unseren Software-Service attraktiver und übersichtlicher zu gestalten, damit Sie bei der Fülle des Angebots das Richtige auf Anhieb finden. Für Ihren speziellen Wunsch benutzen Sie bitte unseren Bestellschein oder rufen Sie uns an.



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

Bitte beachten Sie:

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

Um Ihren Auftrag möglichst schnell erledigen zu können, tragen Sie bitte alle notwendigen Angaben in den Bestellschein ein und senden ihn an:

Tronic-Verlag, Postfach 41, 3444 Wehretal 1

Computronic **Bestellkarte-Software-Service**

Alle im Heft abgedruckten Programme können als zusätzlicher Service über den Verlag bezogen werden.
(Ausland nur gegen Vorkasse)

Die Zustellung erfolgt: gegen **Vorkasse**

oder Inland per **Nachnahme**
+ Versandkosten

innerhalb von 1 Woche

Entnehmen Sie bitte aus unseren Preislisten die notwendigen Angaben für Ihre Bestellung:

Bitte liefern Sie mir:

Cassette für

Bestell-Nr. _____

Anzahl

Diskette für

Bestell-Nr. _____

Anzahl

zum Preis von gesamt _____

DM

Name/Vorname: _____

Straße, Nr.: _____

PLZ/Ort: _____

Datum, Unterschrift _____

Atari

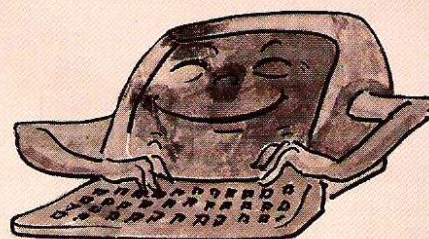
Software-Service



Programm	Preis/Kassette	Preis/Diskette	Bestell-Nr.
Mastermind Schlangenkrieg	10,50	19,50	I 41
Tank-Battle Oil Panic	12,50	19,50	I 51
Startup Zeilen-Split Tomstone-City	12,50	19,50	I 71
Painter Hardcopy	14,—	19,50	I 81
The Big Quest Fünf gewinnt	14,—	19,50	I 91
Splitt	11,—	17,50	I 101
Ski Mutation	14,50	19,50	I 121
Super Miner Diamonds	14,50	19,50	I 22

TI-99

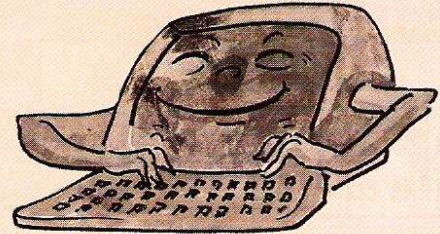
Software-Service



Programm	Preis/Kassette	Preis/Diskette	Bestell-Nr.
Drei Kronen-Spiel Zahlenputzen	8,50	-	T 41
Karl der Käfer Alien-Landing	14,50	-	T 51
Jack the Digger Noah 2099	14,50	-	T 61
Lift Bär ASC II DEF Teil 1	14,—	-	T 71
Maya ASC II DEF Teil 2	14,50	-	T 81
ASC II DEF Teil 1+2	-	19,50	T 881
Transfer Silverspar	14,50	19,50	T 91
Cave Man Alkoholverbot	14,50	19,50	T 101
Moon Race Frogger Slicks	19,50	25,50	T 22

Commodore 64

Software-Service



Programm	Preis/Kassette	Preis/Diskette	Bestell-Nr.
Mauern Widerstand	8,—	15,—	C 41
Space-Comets Erdspalte Sprite-Data	15,—	23,50	C 51
Autostart Bestellschein Roadpainter	16,50	23,50	C 61
Hardcopy Space-Fighter Data-Generator	15,50	19,50	C 71
Monster-Attack Block-Painter Epson-Drucker	16,50	23,50	C 81
Projekt Datenbank	16,—	23,50	C 91
Spiders The Basic	16,50	23,50	C 101
High Noon Skeet Grafik-Designer	17,50	23,50	C 121
Painter Star-Baddle Editor	17,50	23,50	C 22

VC-20

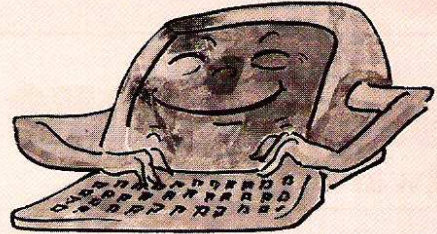
Software-Service



Programm	Preis/Kassette	Preis/Diskette	Bestell-Nr.
Bestellschein Glücky	8,50	15,—	V 61
Multigraph All Rammer	11,—	15,50	V 71
Zyklus Meteorit	11,—	19,50	V 81
Garten Schloß Gruselstein	14,—	19,50	V 91
Fressman Outlaw	14,—	19,50	V 101
Prost Buffalo Bill	14,—	19,50	V 121
Joy Man Powerpack	14,—	19,50	V 22

Apple II

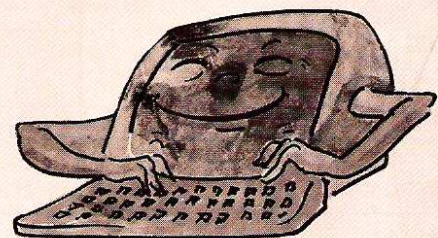
Software-Service



Programm	Preis/Kassette	Preis/Diskette	Bestell-Nr.
Wilder Westen Karambolage Maskengenerator	14,50	19,50	A 41
Music-Maker Mission: Adler Disk-Katalog	9,50	19,50	A 51
Snake Super Datei Shape-tables	12,50	19,50	A 61
Library Fight	12,50	19,50	A 71
Reversal Disk-Menue-Generator	14,—	19,50	A 81
Diamonds Hilfsprogramm	—	19,50	A 91
Tic-Tac-Toe Jumper	—	19,50	A 101
Donovan Basic-Konverter	14,50	19,50	A 121
Funktionstasten Painter Bowling	14,50	19,50	A 22

Dragon 32

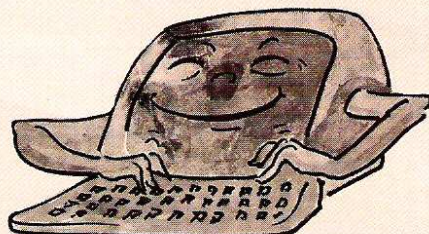
Software-Service



Programm	Preis/Kassette	Bestell-Nr.
Blizzard	8,—	D 41
Space-Flight Geosoft	10,—	D 51
Waregames	8,50	D 61
Laser-Attack	8,50	D 71
Hardcopy	13,—	D 81
Anwenderprogramm	10,—	D 91
Dragon Paint	14,—	D 101

ZX-Spectrum

Software-Service



Programm	Preis/Kassette	Bestell-Nr.
Inventur	12,—	S 41
Missile-Comment	8,50	S 51
Defender Lui der Wurm Alternativer Zeichensatz	13,50	S 61
Matheprogramm Bongo-Beeatcher	12,50	S 71
Solitaire Superstat Kleinstes gem. Vielfache	14,50	S 81
Jump about	14,50	S 91
Pac-Man Oil Panic	14,50	S 101
Frogger	16,—	S 121
Jump	14,50	S 22

ZX-81

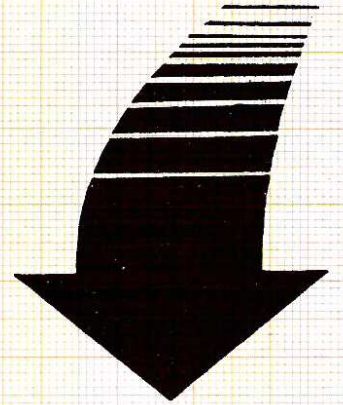
Software-Service



Programm	Preis/Kassette	Bestell-Nr.
Go-Ball Grand-Prix	10,—	Z 51
Moon-Crash ZX-Draw	10,—	Z 61
Tonprogramm Aldebaran	10,—	Z 71
Irrgarten 3D	10,—	Z 81
Reversi	10,—	Z 91
Panik Labyrinth	10,—	Z 101
Expedition	10,—	Z 121
Spinnen	14,50	Z 22

Insgesamt

500,- DM Belohnung!



Jedes Programm in Computronic wird einer Endkontrolle in unserem Hause unterzogen und genauestens geprüft. Alle im Heft abgedruckten Programme sind nach der richtigen Eingabe der Listings auch funktionsfähig. Viele Leser verlieren jedoch schnell die Geduld am Programmieren, sollte die Eingabe einmal nicht klappen. Die häufigste Ursache von Störungen im Programm sind unterlaufene Fehler bei der Eingabe.

Wurde ein fehlerhaftes Listing entdeckt, werden wir in unserer nächsten Ausgabe die Gewinner namentlich erwähnen.
Also, testen Sie unsere Listings. Testen Sie unsere Programme. Vielleicht können wir Sie überzeugen, nicht zu viel versprochen zu haben.
(Die Ermittlung der Gewinner erfolgt unter Ausschluß des Rechtsweges)

5 x 100,- DM zahlen wir Ihnen für den Nachweis eines fehlerhaften Listings in dieser Ausgabe!

Voraussetzung:

- Sie müssen unter den ersten fünf eingegangenen Hinweisen sein,
- der Nachweis des fehlerhaften Listings ist auf dem abgedruckten Nachweis-Coupon zu erbringen,
- es können nur die ausgefüllten Nachweisscheine berücksichtigt werden!

Coupon ausfüllen und einsenden an: Tronic-Verlag, Postfach, **Kennwort: Listing**, 3444 Wehretal 1.

Nachweis-Coupon

Kennwort-Listing

Name/Vorname:

Straße, Nr.:

PLZ/Ort:

Datum, Unterschrift

Ich habe folgenden Fehler in einem Listing entdeckt:

Programmname

Seite

Listing-Zeile

richtig ist:

Unser Paketservice

Kaum zu glauben

Für unsere Leser, die Überraschungen lieben, hat sich unsere Redaktion etwas einfallen lassen. Überzeugen Sie sich selbst. Wir haben Ihnen einige **Super-Angebote** zusammengestellt:

Spitzen-Programme zum Spitzen-Preis:

Bestell-Nr. 100 **kleines Programm-Paket**

3 bespielte Kassetten DM 27,50

Bestell-Nr. 110 **großes Programm-Paket**

3 bespielte Disketten DM 48,-

Bestell-Nr. 200 **exklusive Disketten-Box**

8 bespielte Kassetten DM 64,50

Bestell-Nr. 210 **exklusive Disketten-Box**

8 bespielte Disketten DM 99,-

inkl. 8 bespielte Disketten
zum Preis von DM 148,-

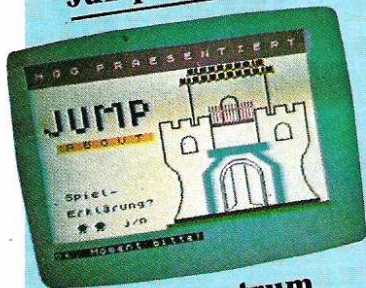
inkl. 10 Leerdisketten
zum Preis von DM 99,-

Alle bespielten Kassetten und Disketten wurden unserem Kassettenservice Seite 84 und 85 entnommen. Angebot gilt nur für Commodore 64, Atari, TI-99 und Apple!

Bestell-Nr. 300 **Computronic-Angebot**

unsere Ausgaben Heft 4-10
zum Sparpreis von DM 26,-!

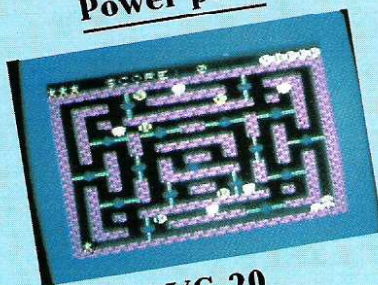
Jump about:



ZX-Spectrum

Aus unserem Angebot:

Power pack



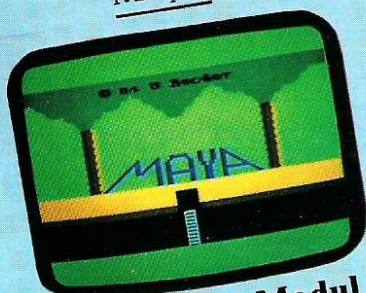
VC-20

Painter:



Commodore 64

Maya:



TI-99/Basic Modul

Star-Baddle:



Commodore 64

Slicks

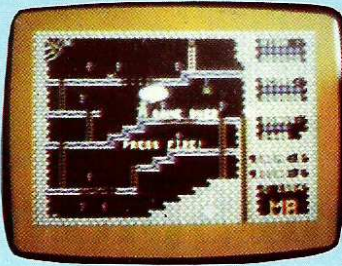


TI-99/Basic Modul

Software-Service aus unserem Angebot

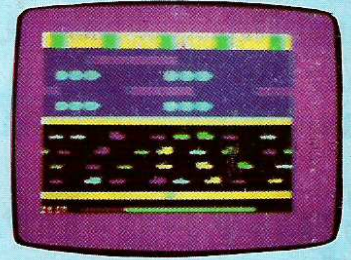
Spiders:

Ziel des Spieles ist es, zwei Spinnen, die auf ihren Netzen herumkrabbeln, zu beseitigen. Vernichten kann man die Tierchen allerdings nur mit Insektengift. Davon liegt noch genug im Keller, aber wo ist der Schlüssel? – Unser Topprogramm aus der Ausgabe Oktober. Spiders gefällt durch eine gute Grafik und guten Sound. Gespielt wird mit Joystick an Port 2. **Für Commodore 64.**



Frogger:

Ein Spiel für den **ZX-Spectrum 48K**. Die beliebte Spielversion jetzt auch für den ZX-Spectrum. Ein Frosch hockt am Straßenrand einer viel befahrenen Straße und versucht verzeifelt, sie zu überqueren. Ist ihm das gelungen, wartet eine zweite schwere Aufgabe auf ihn. Ein Fluß, verseucht mit Krokodilen, muß überquert werden. Das Spiel verfügt über einen guten Sound.

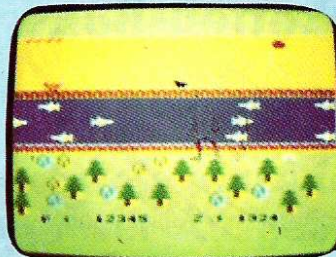


Mother Duck:

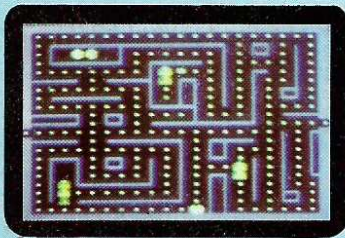
Ähnlich der Spielversion „Frogger“. Eine Entenmutter muß einen Fluß überqueren, um Futter für ihre Jungen zu bekommen. Allerlei Untier

hält sich jedoch im Wasser auf

und macht jede Flußüberquerung zu einer gefährvollen Angelegenheit. **Für TI 99.** Benötigt werden Joystick und das **TI-Ext.-Basic-Modul.**



Fressmann:



Fressmann ist eine Pacman-Version. Das Programm, welches **ohne Erweiterung** lauffähig ist, steht der Originalversion in nichts nach. Fressmann läuft auf dem **VC 20** und wird mit Joystick gespielt.

Projekt:

Als Topprogramm auserwählt von der Redaktion. Sie sollen eine gefährliche Mission ausführen. Um für die nächste Zeit genügend Uran zur Verfügung zu haben, sollen Sie eine Reise zum Mond unternehmen und nach dem edlen Metall suchen. Ihre Reise ist in fünf Phasen aufgliedert, die nacheinander bewältigt werden müssen. Ein schönes Spiel für den **Commodore 64.**



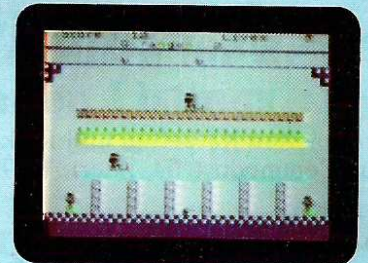
Skeet: (Tontaubenschießen)

Für **Commodore 64**. Dem realistischen Tontaubenschießen nachempfunden. Von einem Katapult geschleuderte Tontauben

müssen reaktionsschnell getroffen werden. In diesem Spiel kann jeder seine eigene Meisterschaft austragen. Eine Supergrafik zeichnet besonders aus.



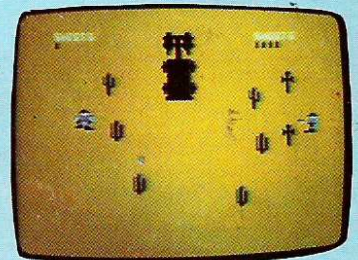
Oil-Panic:



Sie müssen versuchen, möglichst viele Tropfen aufzufangen, die aus einer defekten Ölleitung her austreten. Aber aufgepaßt, denn der Eimer, der zum Auffangen da ist, ist mit nur vier Tropfen gefüllt. **Für ZX-Spectrum 48K.**

High Noon:

Ein tolles Spiel für den **Commodore 64**. High Noon besitzt eine schöne Grafik und eine hohe Spielgeschwindigkeit. Gewählt werden kann zwischen 255! verschiedenen Spielstufen. – Auf einer vielbefahrenen Postkutschenstraße stehen sich zwei Cowboys gegenüber und duellieren sich. Ein Spiel für alle Western-Freunde.



Computronic erscheint jetzt alle 2 Monate im Tronic-Verlag. Der Bezugspreis beträgt im Inland DM 6,50. Im Preis ist die gesetzliche Mehrwertsteuer enthalten.

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